Fail2ban Primer

Contents

1 Overview

Fail2Ban is a firewall adjunct, a spiritual successor to denyhosts[1], which is used to block ip addresses that try to break into your internet server.

Here are some of the traps, and configurations I've needed to setup fail2ban correctly. It's not a complex program, but unless you sit down and understand it, you might get caught. To be honest, it took me some time to figure out fail2ban, as I initially didn't have the patience to sit down and configure it.

2 Instructions for Setup

Quick setup for Devuan / Debian 9: First install fail2ban using apt-get. (apt-get install fail2ban).

Fail2ban is a service that will appear in /etc/init.d/ in Devuan. So it can be managed with service fail2ban {start,stop,restart}. Second, navigate to /etc/fail2ban/jail.d/ Add the following to a sshd.conf file (or name it anything you like)

```
# this is used in devuan. no other changes are made to other files, except
# that the default ssh filter is disabled in jail.conf if it enabled

[sshd]
```

[1] denyhosts was used for ssh, but eventually was abandoned. It was quite a bit simpler to configure than fail2ban, and this was its strength, but it is also more limited, and now has vulnerabilities.

```
ignoreip = 127.0.0.1/8
action   = iptables-allports
maxretry = 6
enabled = true
filter = sshd
logpath = /var/log/auth.log
bantime = 360000
findtime = 3600
```

Now, a few notes on this file.

First, action can be iptables for a single port, or iptables-multiport for more than one, but we are using iptables-allports, as we want to block everything. These actions are listed in /etc/fail2ban/actions.d/

Second, logpath, should point to your ssh log. In devuan ascii / debian stretch (9) it should be /var/log/auth.log. Other distributions may vary. The format of the ssh log can vary as well. In this guide, it's assumed to be auth.log. Gentoo users see below to enable auth.log in syslog-ng.

Third, be careful of different ssh ports. I routinely change ssh ports to be a non standard port, which although it's somewhat pointless, it still seems to block random ssh port scans for port 22. If you use a different port, you must specify it in iptables-multiport above. A potential trap is to use a nonstandard port, then wonder why fail2ban blocks port 22, but your ssh is on port 123 or something. An agressive and easier approach is to just block everything.

Fourth, the default action in iptables-common [2] is to REJECT packets. However, I have changed it to DROP (blocktype=DROP). For those unfamiliar with the difference between REJECT and DROP, from my understanding, it is that REJECT will alert the outside host that the post is unreachable, while DROP simply goes silent, leaving the other host to figure it out on their own.

To configure it, see /etc/fail2ban/actions.d/iptables-common.conf and search for blocktype.

As I consider the offending ip addresses to be attackers, I have set it to DROP. If they try to break into the server, then block all ports from

[2] this parent file in actions.d applies to all child iptables of course, being named "common"

them, and don't tell them anything. The DROP timeout is more work on their end. With REJECT, my server responds. No need to play nice, with people/robots that have no morals.

On fail2ban issues git tracker, there is some discussion about this, and it is not really definitive. It ends up being that, REJECT is default, and if you want you can change it to DROP. As I have. As long as the option is there, I think that is acceptable.

Fifth, review jail.conf, and fail2ban.conf. Usually nothing needs to be changed, but occasionally jail.conf will enable the default sshd jail (which you can disable, and use instead the new one). This will be distribution dependent.

2.1 Getting auth.log to appear in Gentoo

This guide will only cover those working with syslog-ng in Gentoo. You can add a config to syslog-ng to get auth.log to appear in Gentoo. [3] Notice in the below config, that a destination has been defined for authlog. You need not copy all the syslog-ng below, only what you need.

```
/etc/syslog-ng/syslog-ng.confSyslog-ng

@version: 3.17        #mandatory since Version 3, specify
 the version number of the used syslog-ng

options {
        chain_hostnames(no);

        # The default action of syslog-ng is to log a STATS line
        # to the file every 10 minutes.  That's pretty ugly after a while.
        # Change it to every 12 hours so you get a nice daily update of
        # how many messages syslog-ng missed (0).
        stats_freq(43200);
};

source src {
    unix-stream("/dev/log" max-connections(256));
    internal();
};
```

[3] Reference: https://wiki.gentoo.org/wiki/Security_Handbook/Logging#Syslog-ng

```
source kernsrc { file("/proc/kmsg"); };

# define destinations
destination authlog { file("/var/log/auth.log"); };
destination syslog { file("/var/log/syslog"); };
destination cron { file("/var/log/cron.log"); };
destination daemon { file("/var/log/daemon.log"); };
destination kern { file("/var/log/kern.log"); };
destination lpr { file("/var/log/lpr.log"); };
destination user { file("/var/log/user.log"); };
destination mail { file("/var/log/mail.log"); };

destination mailinfo { file("/var/log/mail.info"); };
destination mailwarn { file("/var/log/mail.warn"); };
destination mailerr { file("/var/log/mail.err"); };

destination newscrit { file("/var/log/news/news.crit"); };
destination newserr { file("/var/log/news/news.err"); };
destination newsnotice { file("/var/log/news/news.notice"); };

destination debug { file("/var/log/debug"); };
destination messages { file("/var/log/messages"); };
destination console { usertty("root"); };

# By default messages are logged to tty12...
destination console_all { file("/dev/tty12"); };

# ...if you intend to use /dev/console for programs like xconsole
# you can comment out the destination line above that references /dev/tty12
# and uncomment the line below.
#destination console_all { file("/dev/console"); };

# create filters
filter f_authpriv { facility(auth, authpriv); };
filter f_syslog { not facility(authpriv, mail); };
filter f_cron { facility(cron); };
filter f_daemon { facility(daemon); };
filter f_kern { facility(kern); };
filter f_lpr { facility(lpr); };
```

```
filter f_mail { facility(mail); };
filter f_user { facility(user); };
filter f_debug { not facility(auth, authpriv, news, mail); };
filter f_messages { level(info..warn)
        and not facility(auth, authpriv, mail, news); };
filter f_emergency { level(emerg); };

filter f_info { level(info); };
filter f_notice { level(notice); };
filter f_warn { level(warn); };
filter f_crit { level(crit); };
filter f_err { level(err); };
filter f_failed { message("failed"); };
filter f_denied { message("denied"); };

# connect filter and destination
log { source(src); filter(f_authpriv); destination(authlog); };
log { source(src); filter(f_syslog); destination(syslog); };
log { source(src); filter(f_cron); destination(cron); };
log { source(src); filter(f_daemon); destination(daemon); };
log { source(kernsrc); filter(f_kern); destination(kern); };
log { source(src); filter(f_lpr); destination(lpr); };
log { source(src); filter(f_mail); destination(mail); };
log { source(src); filter(f_user); destination(user); };
log { source(src); filter(f_mail); filter(f_info); destination(mailinfo); };
log { source(src); filter(f_mail); filter(f_warn); destination(mailwarn); };
log { source(src); filter(f_mail); filter(f_err); destination(mailerr); };

log { source(src); filter(f_debug); destination(debug); };
log { source(src); filter(f_messages); destination(messages); };
log { source(src); filter(f_emergency); destination(console); };

# default log
log { source(src); destination(console_all); };
```

3 Future Advancements

What is next for fail2ban after the above? You will want to watch apache logs, and ban any hosts from your IP that search for things they should not

be looking for (wordpress logins, phpmyadmin, etc).

Gentoo has a use flag to use a DB to do persistent blocking over time. This way you can block offending IPs through restarts.

4 Further Reading

https://github.com/fail2ban/fail2ban/issues/2217

https://www.jwz.org/blog/2019/03/apache-2-4-1-killed-fail2ban-so-thats-awesome/

https://www.fail2ban.org/wiki/index.php/Apache

https://www.fail2ban.org/wiki/index.php/