

Fail2Ban - securing services and web applications

Determine, which firewall is running

```
systemctl is-active ufw
ufw status

systemctl is-active nftables
nft list ruleset
```

iptables outputs, when active

```
iptables -L
```

Install, enable and start

```
apt install fail2ban
systemctl enable fail2ban
systemctl start fail2ban
systemctl status fail2ban
```

```
[10:42:24 Sat Jul 26] @gcp1mx1 /home/anton# systemctl status fail2ban
● fail2ban.service - Fail2Ban Service
   Loaded: loaded (/lib/systemd/system/fail2ban.service; enabled; preset: enabled)
   Active: active (running) since Sat 2025-07-26 10:41:40 EEST; 48s ago
     Docs: man:fail2ban(1)
  Main PID: 33622 (fail2ban-server)
    Tasks: 5 (limit: 2344)
   Memory: 16.3M
      CPU: 280ms
   CGroup: /system.slice/fail2ban.service
           └─33622 /usr/bin/python3 /usr/bin/fail2ban-server -xf start

Jul 26 10:41:40 gcp1mx1 systemd[1]: Started fail2ban.service - Fail2Ban Service.
Jul 26 10:41:40 gcp1mx1 fail2ban-server[33622]: 2025-07-26 10:41:40,254 fail2ban.configreader
Jul 26 10:41:40 gcp1mx1 fail2ban-server[33622]: Server ready
```

Observe, take a coffee and understand the config. But do not make changes, as it will be overwritten on the update.

```
vi /etc/fail2ban/jail.conf
```

Specially, which method will be applied for the action. By default, iptables will be invoked to perform an action. To be sure, it will be configured in the local config file as well.

```
201 #
202 # Action shortcuts. To be used to define action parameter
203
204 # Default banning action (e.g. iptables, iptables-new,
205 # iptables-multiport, shorewall, etc) It is used to define
206 # action_* variables. Can be overridden globally or per
207 # section within jail.local file
208 banaction = iptables-multiport
209 banaction_allports = iptables-allports
210
```

Instead, create a local copy of config (which will dominate on default config)

```
cp /etc/fail2ban/jail.conf /etc/fail2ban/jail.local
```

Edit the local config

```
vi /etc/fail2ban/jail.local
```

```
[DEFAULT]
bantime = 1h
findtime = 10m
maxretry = 3
backend = systemd
destemail = to-xx@host
sender = from-fail2ban@host
chain = INPUT
banaction = iptables-multiport
action = %(action_mwl)s
```

```
[nginx-http-auth]
enabled = true
```

```
[nginx-botsearch]
enabled = true
```

```
[nginx-limit-req]
enabled = true
```

```
[nginx-bad-request]
```

```
enabled = true
```

```
[php-url-fopen]
```

```
enabled = true
```

```
[courier-smtp]
```

```
enabled = true
```

```
[postfix]
```

```
enabled = true
```

```
[postfix-rbl]
```

```
enabled = true
```

```
[dovecot]
```

```
enabled = true
```

```
[postfix-sasl]
```

```
enabled = true
```

Restart the service

```
systemctl restart fail2ban
```

```
systemctl status fail2ban
```

```
fail2ban-client status
```

```
[11:10:39 Sat Jul 26] @gcp1mx1 /etc/fail2ban# systemctl status fail2ban
● fail2ban.service - Fail2Ban Service
   Loaded: loaded (/lib/systemd/system/fail2ban.service; enabled; preset: enabled)
   Active: active (running) since Sat 2025-07-26 11:07:23 EEST; 3min 19s ago
     Docs: man:fail2ban(1)
  Main PID: 37463 (fail2ban-server)
    Tasks: 25 (limit: 2344)
   Memory: 230.1M
      CPU: 6.991s
   CGroup: /system.slice/fail2ban.service
           └─37463 /usr/bin/python3 /usr/bin/fail2ban-server -xf start

Jul 26 11:07:23 gcp1mx1 systemd[1]: Started fail2ban.service - Fail2Ban Service.
Jul 26 11:07:23 gcp1mx1 fail2ban-server[37463]: 2025-07-26 11:07:23,712 fail2ban.configreader [37463]: WARNING 'allowipv6' not defined in 'Definition'. Using default one: 'auto'
Jul 26 11:07:28 gcp1mx1 fail2ban-server[37463]: Server ready
[11:10:42 Sat Jul 26] @gcp1mx1 /etc/fail2ban# fail2ban-client status
Status
|- Number of jail:      11
  `-- Jail list:        courier-smtp, dovecot, nginx-bad-request, nginx-botsearch, nginx-http-auth, nginx-limit-req, php-url-fopen, postfix, postfix-rbl, postfix-sasl, sshd
[11:10:44 Sat Jul 26] @gcp1mx1 /etc/fail2ban#
```

To enable jail for nginx webserver

Requires modification to webserver configuration as described in

```
https://nginx.org/en/docs/http/nginx_http_limit_req_module.html
```

Create addition config file, which will be included by main configuration.

```
vi /etc/nginx/conf.d/nginx_http_limit_req_module.conf
```

```
limit_req_zone $binary_remote_addr zone=one:10m rate=50r/s;

# for troubleshooting, enable limit_log logformat for access_log

limit_req_log_level info;
log_format limit_log '[$time_iso8601, $msec]: limit:$limit_req_status code:$status from:$remote_addr
to:$host$request_uri';
```

In addition to this, limitation need to be applied in every server config (per host):

```
vi /etc/nginx/sites-enabled/(host).conf
```

```
location / {
    limit_req zone=one burst=50 nodelay;
}
```

```
location / {
    limit_req zone=one burst=5;
    try_files $uri $uri/ /index.php?$query_string;
}
```

After webserver config changes, test and reload new config

```
nginx -t
nginx -s reload
```

To trigger 'nginx-botsearch'

```
for i in {1..60}; do curl -A "Trigger" http://dox.2dz.fi; done
```

Revision #4

Created 26 July 2025 07:38:11 by Anton

Updated 20 August 2025 12:59:28 by Anton