Registration and pre-moderation

All newly created accounts and domains are pre-moderated in our system. We verify company and website information that you provide during registration. You need to fill in the full name of the organization, legal and actual addresses (detailed information about the company you can provide later) and create a domain in our system indicating the website that you are going to protect.

New Qrator domain has the user-configured upstream configuration, SSL configuration (if certificates have been installed), an identification number of the form dXXXX and waits for the enabling in the "Moderation" status.

> We enable domains according to the internal policy of our company. We provide services only to legal entities by contract.

An enabled domain gets Qrator IP, that you need to apply to protect user's traffic of your website. Setup instructions sends to the contact's e-mail, that you specified at the registration and to all contacts that have techincal role, if you have such contacts in the account

The basic settings of Qrator domain for the proper operation of the service. The redirection incoming website traffic to the Qrator network

Here is the short step-by-step guide for setting up Qrator domain for the common cases:

> **Caution:** Before the modification of DNS A-record on Qrator IP, we strongly suggest reviewing the info section HTTPS filtering because this feature is custom pre-configured for each Qrator domain. If you do not use HTTPS, you can skip this step

1. Review the section HTTPS filtering and choose the type of service. Then you need to download certificate chain and set up it to the domain in the dashboard or follow the instructions of technical support in the opened ticket if you choose "HTTPS filtering without decryption".
2. Make sure that you specified the valid IP address of your server in the domain settings that is called "upstream" in our terms. If you have a few upstreams, you can set up the flexible traffic load balancing among them on Qrator side. To get more information review upstreams info section.
3. Review the section connection to Qrator. Here you get to know how to test the current configuration of Qrator domain even before you redirect users traffic to the Qrator network. After that, you can proceed with the setup instructions to redirect incoming website traffic to the Qrator network.
4. Configure firewall of your upstream using the section firewall configuration so that to prevent direct upstream IP targeted attacks (bypassing Qrator).

Other features of dashboard and recommendations

Finally, after the completing of basic configuration steps, we also recommend:

5. Review the section Real-IP to set up support and parsing of `X-Forwarded-For` HTTP header field on your web server.
6. Setting up two-factor authentication and personalize the contacts of your Qrator personal account using the following sections

Prevent check of Qrator service configuration

Before you start forwarding users' traffic to our network we suggest making several test requests to protected service using Qrator IP. It allows you to check that the Qrator service configuration is ready to handle user requests. You can do it using command line tool curl or any web browser Example:

```
# An example of using curl to resolve the domain name in Qrator IP
    curl -I --resolve example.com:80:qrator_ip http://example.com/
    curl -I --resolve example.com:443:qrator_ip https://example.com/

or

# Add an entry with your Qrator IP and domain name into the file /etc/hosts
    178.248.2XX.XX example.com

# Use curl with option -I in the command-line to check which HTTP status you
will get after
# the following provided instructions
    curl -I example.com
```

> **Please note:**
> If you receive an HTTP 502 Error we recommend reviewing info sections HTTP status codes and firewall configuration to identify possible causes of this issue

Connection to Qrator

> **TTL of DNS A-Record** First of all, we suggest checking the TTL value of the current A-record and changing it to the recommended value. The recommended TTL value for A-record for the domain is 300 or lower.

To forward users' traffic of your website to our filtration network you have to change the current IP address in DNS A record of the domain name to Qrator IP. Usually, the example of those records looks like the following ones:

```
@ IN A Qrator_IP
WWW IN A Qrator_IP
```

After DNS is updated globally, requests for your domain should arrive only from IP addresses that belong to QRATOR filtering network:

- 87.245.197.192
- 87.245.197.193
- 87.245.197.194
- 87.245.197.195
- 83.234.15.112
- 83.234.15.113

- 83.234.15.114
- 83.234.15.115
- 66.110.32.128
- 66.110.32.129
- 66.110.32.130
- 66.110.32.131
- 185.94.108.0/24

The setup instructions sends to the registration e-mail and the other contact with technical roles if you have them. To get more information about the roles of contacts in the dashboard read the section contacts

The connection to Qrator under a DDoS attack

In case of the service connection when you are under an attack, the IP addresses of the victim's website can be compromised by attackers, and we recommend obtaining new ones at the hosting where to we will forward filtered traffic. At your server's firewall level you have to restrict access to the direct IP for all IP addresses except the trusted list of IPs and Qrator nodes IP list which you can find in the section firewall configuration with the examples of configurations.

Identifying the originating IP addresses of users

In the Reverse Proxy protection mode all requests come to you with changed source IP to Qrator nodes IPs. The originating IP addresses of users we write to`X-Forwarded-For` HTTP header field and we recommend setting up its support and parsing on your web server

XFF parsing how-to (Nginx)

If your web application uses Nginx as a front-end, please consider adding the following snippet to the Nginx configuration file. This way Nginx will be able to parse the `X-Forwarded-For` header securely and send the end user´s IP address to the application.

```
#  Add Qrator network addresses to the trusted list:
set_real_ip_from 66.110.32.128/30;
set_real_ip_from 83.234.15.112/30;
set_real_ip_from 87.245.197.192/30;
set_real_ip_from 185.94.108.0/24;

# Use "X-Forwarded-For" header as a source:
real_ip_header X-Forwarded-For;

# Send the extracted user address to the application in the X-Real-IP header:
proxy_set_header X-Real-IP $remote_addr;
```

Upstream Definition

Upstreams are real IP addresses issued by your ISP for the servers hosting your domains. Qrator filtering network utilizes these addresses to pass legitimate users' traffic to your protected resources. Each domain must be assigned at least 1 upstream.

> **Caution:** we strongly recommend that you obtain new IP addresses from your ISP to use them with Qrator filtering service. This is crucial because the attackers may already have the information on your current addresses which can be used to direct a DDoS attack bypassing the protection scheme.

Using the Upstream Configuration widget

The Upstream Configuration widget allows adding, deleting, editing and moving upstreams between Primary and Backup lists. Each upstream record consists of the following fields:

- **Name**: this is an optional field that you can specify to remember and distinguish upstreams on the list. It can be left blank.
- **IP**: your server IP address which is to receive clean traffic from the Qrator network.
- **Enabled**: tick this to start using the given upstream, untick to disable it. Disabled by default
- **Weight**: in case the Upstream weights option is enabled this field replaced the Enabled checkbox. Sets the weight value determining the share of total user traffic directed to the given upstream. The value ranges from 0 to 64. the default value is 1; if set to 0, it forbids sending any traffic to this upstream.

For a newly added domain, there is a single Primary upstream record on the list which cannot be deleted or disabled. You can then add new upstream records by pressing Add upstream button. Creating one or several upstreams in additional to the original one opens a list of editable parameters:

- **Balancing mode**: this block allows choosing either of two predefined methods used for dividing the load among multiple upstreams.
    - **Round-robin**: the default method. Each incoming user request is sent to the top upstream on the list, after that the list is rotated moving this upstream to the bottom and the next one to the top to handle the next request. Hence, the requests are distributed evenly in a cycle-like fashion, making the load equal for all upstreams.
    - **IP Hash**: for each incoming request the hash is calculated from the user's source IP address. This request then goes to one of the upstreams on the list depending on the hash value. This allows sending the next requests from the same users to the same upstreams which received their original requests.
- **Additional parameters**:
    - **Upstream weights**: enable alternative balancing method which is set manually for each upstream. Use the Weight field in each upstream record to specify the share of the total load handled by it. More details in the Weights field manual above.
    - **Use backups**: this option enables a second upstream list under Backup headline. This upstream list will be used only in case all Primary upstreams become unreachable. As soon as any of the Primary upstreams is working again, the Backup list will not be

used. Each Backup upstream record follows the same rules as Primary ones do, and the chosen balancing method applies both to Primary and Backup lists.

Removing an upstream record is done by pressing the red minus button on the right side of it. If Use Backups box is checked, you can drag and drop upstream records to move them between Primary and Backup lists.

Firewall configuration: restricting access to the protected server

When using proxy-based protection (default Qrator domains without any HTTPS technologies and domains with "HTTPS filtering with decryption") Qrator forwards user requests to your servers using these IP addresses as a source:

- 66.110.32.128/30
- 83.234.15.112/30
- 87.245.197.192/30
- 185.94.108.0/24

To prevent attacks to your server targeted at its IP address (instead of its domain name), you need to set up the firewall to deny direct HTTP/HTTPS access to anyone except the aforementioned list of source addresses. This list can be expanded with your own trusted addresses (i.e. office networks, developer workstations and automated tools). It will nullify the probability of false-positive bans for your company´s staff.

Linux Netfilter configuration guide (Proxy-based protection)

> **Caution:**
> Making a "one size fits all" firewall how-to is virtually impossible because there are thousands of possible ways of configuring a system´s firewall (including all iptables extensions and even other userspace Netfilter implementations) and most of the time the firewall is already configured in some way. Please remember that the scripts provided below are just examples and cannot be applied to your system "as-is" unless you´re sure that it won´t break your firewall (i.e. when its config is empty).

To prevent a possibility of DDoS attacks to the direct IP of the protected server you should drop all incoming connections to HTTP/HTTPS ports (TCP/80 and TCP/443) for all remote IPs except some trusted addresses. This list must include all Qrator Network source IPs and may be expanded with your own trusted addresses, i.e. company workstations.

Depending on your system, you can either use "plain" iptables rules, or combine them with rules for iptables that support Netfilter´s conntrack and ipset modules. We recommend using both conntrack and ipset, because it makes iptables ruleset smaller (thus making its maintenance easier) and faster (less rules means fewer requests on the incoming packet).

Plain iptables configuration example

```
#!/bin/sh
```

```
ADMIN_IPS="127.0.0.1" # Add your trusted IPs/subnets (staff, admins, tools and
etc.) here

QRATOR_NODES="66.110.32.128/30
83.234.15.112/30
87.245.197.192/30
185.94.108.0/24
"

iptables -N qrator_ips
for IP in $ADMIN_IPS $QRATOR_NODES; do
    iptables -A qrator_ips -s $IP -j RETURN
done
iptables -A qrator_ips -j DROP

iptables -A INPUT -p tcp -m multiport --dports 80,443 -j qrator_ips
```

Iptables with conntrack and ipset support

```
#!/usr/bin/env bash

ADMIN_IPS="127.0.0.1" # Add your trusted IPs/subnets (staff, admins, tools and
etc.) here

QRATOR_NODES="66.110.32.128/30
83.234.15.112/30
87.245.197.192/30
185.94.108.0/24
"

# Creating the trusted IP set:
ipset -N trusted_nodes hash:net
for ip in $ADMIN_IPS $QRATOR_NODES; do
    ipset -A trusted_nodes ${IP}
done

# Creating the iptables rules:
iptables -N qrator
iptables -A qrator -m set --match-set trusted_ips src -j ACCEPT
iptables -A qrator -j DROP

iptables -I INPUT --state ESTABLISHED,RELATED -j ACCEPT
iptables -A INPUT -p tcp -m multiport --dports 80,443 --state NEW -j qrator
```