

Nginx-Exporter Installation

- Install docker - [Docker Installation](#)
- Disable SELinux
Modify this file
`vi /etc/selinux/config`

```
# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#     enforcing - SELinux security policy is enforced.
#     permissive - SELinux prints warnings instead of enforcing.
#     disabled - No SELinux policy is loaded.
SELINUX=disabled
# SELINUXTYPE= can take one of these two values:
#     targeted - Targeted processes are protected,
#     mls - Multi Level Security protection.
SELINUXTYPE=targeted
```

Execute

```
sudo setenforce 0
```

- add server module at end of `/etc/nginx/nginx.conf` file

```
server {
    listen      7121 default_server;
    listen      [::]:7121 default_server;
    root        /usr/share/nginx/html;
    location /metrics {
        stub_status on;
    }
}
```

Whole `/etc/nginx/nginx.conf` file should look this

```
user nginx;
worker_processes auto;
error_log /var/log/nginx/error.log;
pid /run/nginx.pid;

# Load dynamic modules. See /usr/share/doc/nginx/README.dynamic.
include /usr/share/nginx/modules/*.conf;

events {
    worker_connections 1024;
}

http {
    log_format main '$remote_addr - $remote_user [$time_local]
```

```

"$request" '
        '$status $body_bytes_sent "$http_referer" '
        '"$http_user_agent" "$http_x_forwarded_for"';

access_log /var/log/nginx/access.log main;

sendfile            on;
tcp_nopush          on;
tcp_nodelay         on;
keepalive_timeout   65;
types_hash_max_size 2048;

include             /etc/nginx/mime.types;
default_type        application/octet-stream;

# Load modular configuration files from the /etc/nginx/conf.d
directory.
# See http://nginx.org/en/docs/nginx_core_module.html#include
# for more information.
include /etc/nginx/conf.d/*.conf;

server {
    listen      80 default_server;
    listen      [::]:80 default_server;
    server_name www.shopify.argo.id.com shopify.argo.id.com;
    root        /usr/share/nginx/html;

    # Load configuration files for the default server block.
    include /etc/nginx/default.d/*.conf;

    location / {

    }

    location /metrics {
        stub_status on;
    }

    error_page 404 /404.html;
    location = /404.html {
    }

    error_page 500 502 503 504 /50x.html;
    location = /50x.html {
    }

    listen [::]:443 ssl ipv6only=on; # managed by Certbot
    listen 443 ssl; # managed by Certbot
    ssl_certificate /etc/letsencrypt/live/shopify.argo.id.com
/fullchain.pem;

```

```

        ssl_certificate_key /etc/letsencrypt/live/shopify.argo.id.com
        /privkey.pem;
    }

    server {
        listen      7121 default_server;
        listen      [::]:7121 default_server;
        root         /usr/share/nginx/html;
        location /metrics {
            stub_status on;
        }
    }
}

```

- `sudo nginx -t`
- `sudo systemctl restart nginx`
- Run Nginx-exporter docker container

```

sudo docker run --name nginx-exporter -d -p 9113:9113 nginx/nginx-
prometheus-exporter:0.9.0 -nginx.scrape-uri=http://<nginx_vm_ip>:7121
/metrics -nginx.ssl-verify=false

```

- Execute this command to auto restart the container if it stops
`sudo docker update --restart=always nginx-exporter`