nginx proxy manager behind cloudflare

here's the situation: you have a webpage running on a couple of docker containers which are all behind a nginx proxy manager reverse proxy and now you want to use cloudflare to protect your site.

here's what you have to do to achieve that:

- create a cloudflare account
- add your domain to it
  - by default, this means making cloudflare your dns server, if you don't want that you can configure a partial cname forward but this is only possible with a business or enterprise subscription
  - import your current dns config to get all your dns entries over to cloudflare
  - change your ns records with your registrar (this can take several hours)
- choose which host entries to forward through cloudflare
- set up ssl
  - on the cloudflare admin page under “SSL/TLS” choose the “Full (strict)” encryption mode
  - then go to Edge certificates and make sure you use automatic certificates from cloudflare
  - then go to Origin certificates and create a long lasting certificate for your server
  - now go to nginx proxy manager to SSL Certificates and add a new custom certificate, you will only need the key and certificate, no chain
  - in your host config on nginx proxy manager switch the ssl certificate to the newly created one
- you should now be able to see your site
- now fix the client ip address, for this we need to tell to nginx proxy manager who can send X-Forwarded-For headers to provide real client ips.

I have created a script to automatically write a include file, which can be included in the host configuration on nginx proxy manager under Advanced –> Custom Nginx Configuration. add

```
include /data/nginx/custom/cloudflare[.]conf;
```

in the text field and make sure to adjust the path in the following script to point to the same location (note that the path in the custom config is what the path is inside your nginx proxy manager container, and in the update script it will be the path of the host server).

here is the script that generates the include:

update_cloudflare.sh

```
#!/bin/bash
newconf=/tmp/cloudflare.conf
liveconf=/opt/proxy/data/nginx/custom/cloudflare.conf
echo "#Cloudflare" > $newconf;
for i in $(curl https://www.cloudflare.com/ips-v4 2>/dev/null); do
  echo "set_real_ip_from $i;" >> $newconf;
done
for i in $(curl https://www.cloudflare.com/ips-v6 2>/dev/null); do
  echo "set_real_ip_from $i;" >> $newconf;
```

```
done
echo "real_ip_header X-Forwarded-For;" >> $newconf;
echo "real_ip_recursive on;" >> $newconf;

if ! diff -q $liveconf $newconf ; then
  echo "cloudflare ip list has changed, reloading nginx proxy manager";
  cp $newconf $liveconf
  cd /opt/proxy
  docker-compose exec nxapp nginx -s reload
fi
```

adjust the $liveconf path and the name of the nginx proxy manager app for docker-compose to reload “nxapp” in my example. if you are not using docker-compose, use some other method to run the reload command in your docker container here.

- setup authenticated origin pulls

add custom config to nginx proxy manager host config

```
ssl_verify_client on;
ssl_client_certificate /data/nginx/cloudflare.pem;
```

download cloudflare certificate from https://developers.cloudflare.com/ssl/static/authenticated_origin_pull_ca.pem and save it as /data/nginx/cloudflare.pem

finally enable authenticated origin pulls in cloudflare admin console under “SSL/TLS –> Origin Server”