

# Installer le sous-domaine "awstats"

- **Notes :**

- Ce domaine hébergera l'outil Awstats permettant d'analyser et de visualiser les logs web.
- Cet outil est hébergé dans un container Docker.
- Il est nécessaire de réaliser cet installation sur le serveur hébergeant les log du serveur web que nous souhaitons analyser. Dans notre cas, l'instance "web-srv".

- **Ressources :**

- Dépôt justb4/awstats: Dockerfile et exemple Docker Compose

## Installer le domaine

- Créer un fichier de configuration : vi /etc/nginx/sites-available/awstats.conf
  - Y placer le contenu suivant :

```
server {
    listen 80;
    listen [::]:80;

    server_name awstats.<domaine-sinp>;

    auth_basic "Zone restreinte";
    auth_basic_user_file /etc/nginx/.htpasswd;

    location / {
        proxy_set_header X-Forwarded-Host $host:$server_port;
        proxy_set_header X-Forwarded-Server $host;
        proxy_set_header X-Forwarded-For
$proxy_add_x_forwarded_for;
        proxy_set_header X-Forwarded-Proto $scheme;

        proxy_pass http://127.0.0.1:50083/; # ATTENTION : bien
mettre un slash final ! Sinon => erreur 404
    }
}
```

- Voir la section [Configurer Nginx](#) pour la création et la gestion du fichier .htpasswd.
- Créer un lien depuis les sites actifs : nginx\_ensite awstats.conf ou cd /etc/nginx/sites-enabled/ ; ln -s ../sites-available/web-log-analyser.conf web-log-analyser.conf
  - Tester la config et relancer Nginx si OK: nginx-reload ou nginx -t && nginx -s reload
  - Tester l'URL http://awstats.<domaine-sinp>/ qui doit afficher une erreur 502 car nous n'avons pas encore lancé le container Docker.
- En local, sur votre machine, se placer dans le dépôt Github "sinp-<region>-srv" récupéré précédemment et si nécessaire resynchroniser le dossier web-srv avec le serveur de destination en exécutant la commande Rsync indiquée dans le fichier README.md : rsync -av ./web-srv/home/admin/docker/ admin@web-<region>-sinp:/home/admin/docker/

- Sur le serveur "web-srv" dans le dossier *docker* de l'utilisateur *admin* :
  - vérifier la présence du réseau Docker spécifique à notre utilisation de type *bridge* nommé *nginx-proxy* (voir fichier *.env*) : `docker network ls`
  - se placer dans le dossier *awstats* : `cd ~/docker/awstats/`
  - exécuter la commande : `docker-compose up`
  - vérifier que tout fonctionne à l'adresse : `http://awstats.<domaine-sinp>`. Il se peut que les stats soient vides si le script de mise à jour n'a pas encore été lancé dans le container.
  - arrêter le container : `CTRL+C`
  - relancer le container en tant que service : `docker-compose up -d`
    - si besoin de l'arrêter utiliser : `docker compose down`

## Activer le SSL et HTTP2 sur le domaine

- Installer un certificat SSL via Certbot (Let's Encrypt) :
  - Pour SINP PACA : `certbot --nginx -d awstats.silene.eu`
  - Pour SINP AURA : `certbot --nginx -d awstats.biodiversite-aura.net`
  - Répondre : 1
  - Modifier le fichier de configuration de Nginx comme ci-dessous afin d'activer le support de SSL, HTTP2 et la redirection de HTTP vers HTTPS.
    - Recharger les configs Nginx : `nginx-reload` ou `nginx -t && nginx -s reload`
  - Tester ensuite la redirection auto de HTTP vers HTTPS :  
`http://web-log-analyser.<domaine-sinp>/` → doit rediriger vers HTTPS automatiquement
- Tester la configuration SSL :  
<https://www.ssllabs.com/ssltest/analyze.html?d=awstats.<domaine-sinp>>
- Tester l'URL `https://awstats.<domaine-sinp>/`
- La config finale :

```
server {  
    listen 443 ssl http2;  
    listen [::]:443 ssl http2;  
  
    server_name awstats.<domaine-sinp> ;  
  
    auth_basic "Zone restreinte";  
    auth_basic_user_file /etc/nginx/.htpasswd;  
  
    location / {  
        proxy_set_header X-Forwarded-Host $host:$server_port;  
        proxy_set_header X-Forwarded-Server $host;  
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;  
        proxy_set_header X-Forwarded-Proto $scheme;  
  
        proxy_pass http://127.0.0.1:50083/; # ATTENTION : bien mettre un  
slash final ! Sinon => erreur 404  
    }  
  
    ssl_certificate /etc/letsencrypt/live/awstats.<domaine-
```

```

    sinp>/fullchain.pem; # managed by Certbot
        ssl_certificate_key /etc/letsencrypt/live/awstats.<domaine-
    sinp>/privkey.pem; # managed by Certbot
            include /etc/letsencrypt/options-ssl-nginx.conf; # managed by
Certbot
            ssl_dhparam /etc/letsencrypt/ssl-dhparams.pem; # managed by Certbot
}

server {
    listen 80;
    listen [::]:80;

    server_name awstats.<domaine-sinp> ;

    return 302 https://awstats.<domaine-sinp>$request_uri;
}

```

## Configurer la génération des rapports GoAccess

- **Notes** : Logrotate pour Nginx doit avoir été configuré pour utiliser un format de date %Y-%m-%d
- Le dépôt Git du projet SINP-PACA contient les scripts de génération de rapport journalier, hebdomadaire et mensuel pour GoAccess dans : /web-srv/opt/goaccess-reports
  - Les transférer sur le serveur :
    - Tester : rsync -av ./web-srv/opt/ admin@web-<region>-sinp:/home/admin/dwl/ --dry-run
    - Lancer le transfert (supprimer l'option --dry-run : rsync -av ./web-srv/opt/ admin@web-<region>-sinp:/home/admin/dwl/
  - Copier le contenu de ce dossier dans /opt/goaccess-reports/ : cp -r /home/admin/dwl/goaccess-reports /opt/
  - Donner les droits d'exécution aux scriptes Bash : chmod +x \*-report
  - Copier le fichier goaccess-reports.cron dans /etc/cron.d/ en supprimant son extension .cron : cp /otp/goaccess-reports/goaccess\_reports.cron /etc/cron.d/goaccess\_reports
  - Recharger le service Cron pour prendre en compte les changements : service cron reload
- S'il est nécessaire de générer manuellement un rapport utiliser les commandes :
  - Rapport journalier : /opt/goaccess-reports/daily-report <date-au-format-YYYY-MM-DD> . Ex. : /opt/goaccess-reports/daily-report 2020-01-01
    - Le rapport est accessible ensuite à une adresse de la forme : /<année>/<mois>/<jour>
    - Ex. : <https://web-log-analyser.silene.eu/2020/01/01>
  - Rapport hebdomadaire : /opt/goaccess-reports/weekly-report <date-d-un-lundi-au-format-YYYY-MM-DD> . Ex. : /opt/goaccess-reports/daily-report 2020-01-13
    - Le rapport est accessible ensuite à une adresse de la forme : /<année>/weeks/<numéro-semaine>
    - Ex. : <https://web-log-analyser.silene.eu/2020/weeks/02>

Last update: 2021/04/22 08:58  
serveurs:installation:web-srv:docker-awstats https://sinp-wiki.cbn-alpin.fr/serveurs/installation/web-srv/docker-awstats?rev=1619081881

- Rapport mensuel : /opt/goaccess-reports/monthly-report <date-d-un-jour-du-mois-au-format-YYYY-MM-DD> . Ex. : /opt/goaccess-reports/monthly-report 2020-01-01
  - Le rapport est accessible ensuite à une adresse de la forme : /<année>/<mois>
  - Ex. : <https://web-log-analyser.silene.eu/2020/01>

From:  
<https://sinp-wiki.cbn-alpin.fr/> - CBNA SINP



Permanent link:  
<https://sinp-wiki.cbn-alpin.fr/serveurs/installation/web-srv/docker-awstats?rev=1619081881>

Last update: **2021/04/22 08:58**