Installer, configurer et gérer le sous-domaine "analytics"

Notes : ce domaine hébergera l'outil *Matomo* permettant d'analyser l'utilisation des sites web du SINP (*Wordpress*, *GeoNature*...). Il est embarqué dans un container *Docker* et fait partie de la *stack analytics.silene.eu*.

Installer le domaine

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

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

    server_name analytics.silene.eu;

    location / {
        proxy_set_header Host $http_host;
        proxy_set_header X-Real-IP $realip_remote_addr;
        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
    }
}
```

- Créer un lien depuis les sites actifs : `cd /etc/nginx/sites-enabled/ ; ln -s ../sites-available/analytics.conf analytics.conf`
  - Tester la config et relancer Nginx si OK : `nginx-reload` ou `nginx -t && nginx -s reload`
  - Tester l'URL `http://analytics.<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-paca-srv*" récupéré précédemment et si nécessaire resynchroniser le dossier *web-srv/docker/* avec le serveur de destination en exécutant la commande *Rsync* indiquée dans le fichier README.md.
- Sur le serveur 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 *analytics.silene.eu* : `cd ~/docker/analytics.<domaine-sinp>`

    - exécuter la commande : `docker-compose up`
    - vérifier que tout fonctionne à l'adresse : `https://analytics.<domaine-sinp>/` (se loguer avec le compte root Mysql)
    - 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 SSL et HTTP2 sur le domaine

- Installer un certificat SSL via Certbot (Letsencrypt) : `certbot --nginx -d analytics.<domaine-sinp>`
    - Pour SINP PACA : `certbot --nginx -d analytics.silene.eu`
    - Pour SINP AURA : `certbot --nginx -d analytics.biodiversite-aura.net`
    - Répondre : 2
    - Tester ensuite la redirection auto de HTTP vers HTTPS : http://analytics.silene.eu/ → doit redirigé vers HTTPS automatiquement
- Tester la configuration SSL : https://www.ssllabs.com/ssltest/analyze.html?d=analytics.silene.eu
- Tester l'URL https://analytics.silene.eu/
- La config finale :

```
server {
    listen 443 ssl http2; # managed by Certbot
    listen [::]:443 ssl http2; # managed by Certbot

    server_name analytics.silene.eu;

    location / {
        proxy_set_header Host $http_host;
        proxy_set_header X-Real-IP $realip_remote_addr;
        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/analytics.silene.eu/fullchain.pem; # managed by Certbot
    ssl_certificate_key /etc/letsencrypt/live/analytics.silene.eu/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 analytics.silene.eu;
    return 302 https://analytics.silene.eu$request_uri;
}
```

Matomo

Configurer Matomo

- Se connecter sur https://analytics.silene.eu avec un compte de super-administrateur
- Cliquer sur l'icône de roue en haut à droite pour accéder aux paramètres
- Accéder au menu "Système" > "Paramètres généraux" > "Paramètres du serveur mail" et compléter comme suit :
  - Adresse du serveur SMTP : ssl0.ovh.net
  - Port SMTP : 587
  - Méthode d'authentification pour le serveur SMTP : login
  - Nom d'utilisateur SMTP : mailer@silene.eu
  - Mot de passe SMTP : utiliser le mot de passe du compte mailer@silene.eu.
  - Adresse d'origine SMTP : mailer@silene.eu
  - Nom d'origine SMTP : No Reply
  - Chiffrement SMTP : TLS
- Modifier le fichier de config une fois le volume créé :
  - Copier le fichier de config de Matomo sur l'hôte : `docker cp analytics-matomo:/var/www/html/config/config.ini.php ~/docker/analytics.silene.eu/matomo/config/`
  - Modifier le fichier de config en ajoutant les paramètres nécessaire :

    ```
    [General]
    force_ssl = 0
    assume_secure_protocol = 1
    proxy_client_headers[] = "HTTP_X_FORWARDED_FOR"
    proxy_host_headers[] = "HTTP_X_FORWARDED_HOST"
    ```

    - Le service Nginx communique en interne en HTTP avec Matomo (`force_ssl=0`) par contre on indique à Matomo qu'il doit renvoyer ces liens (images, css, js...) en considérant que l'on utilise du HTTPS (`assume_secure_protocol = 1`) et comme Matomo est proxifié par Nginx nous l'indiquons (`proxy_client_headers[] = "HTTP_X_FORWARDED_FOR"` et `proxy_host_headers[] = "HTTP_X_FORWARDED_HOST"`)
  - Retourner le fichier de config dans le volume : `docker cp ~/docker/analytics.silene.eu/matomo/config/config.ini.php analytics-matomo:/var/www/html/config/config.ini.php`
  - Se connecter au container et corriger les droits du fichier copié : `docker exec -it analytics-matomo /bin/bash`

        - `cd /var/www/html/config`
        - `chown www-data:www-data config.ini.php`
- Test l'envoie d'email par Matomo :
    - Se connecter au container : `docker exec -it analytics-matomo /bin/bash`
    - Se placer dans le dossier */var/www/html/* avec : `cd /var/www/html/`
    - Lancer la commande : '' ./console core:test-email test@mon-domaine.ext ''

Mise à jour de Matomo

- **Notes** : le volume Docker incluant l'ensemble de Matomo, il est donc nécessaire de procéder en 2 étapes :

1. Mise à jour du container :
    - Mettez à jour le fichier *docker-compose.yml* en indiquant la nouvelle version de l'image
    - Téléverser le fichier sur le serveur
    - Relancer le build du container : `docker-compose up -d --no-deps --build analytics-matomo`
2. Mise à du code de Matomo présent dans le volume Docker :
    - Se loguer sur Matomo avec un super-utilisateur
    - Via l'interface "Paramètres" lancer la mise à jour de Matomo.
3. Mise à jour de la base de données via le terminal dans Docker (si problème via l'interface web):
    - Accéder au container : `docker exec -it analytics-matomo /bin/bash`
    - Se placer dans le dossier principal : `cd /var/www/html`
    - Lancer la commande de mise à jour : `./console core:update`
        - Pour voir la liste des commandes dispo : `./console list`
    - Éventuellement, redonner les droits à l'utilisateur Nginx : `chown -R www-data:www-data /var/www/html`

Déboguer un graphique affichant "Oups..."

- Essayer de redémarrer le service `analytics-nginx`
- Ouvrir l'outil de Dev de votre navigateur, se rendre dans l'onglet "Réseau" et afficher la réponse de la requête Ajax affichant le graphique en erreur.
    - Des informations sur comment résoudre le problème devrait être affichée.
- Vérifier que le contenu du dossier `/var/www/html/tmp/templates_c/` appartient bien à l'utilisateur `www-data`

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

Permanent link:
https://sinp-wiki.cbn-alpin.fr/serveurs/installation/web-srv/docker-matomo?rev=1790244348

Last update: **2026/09/24 10:05**