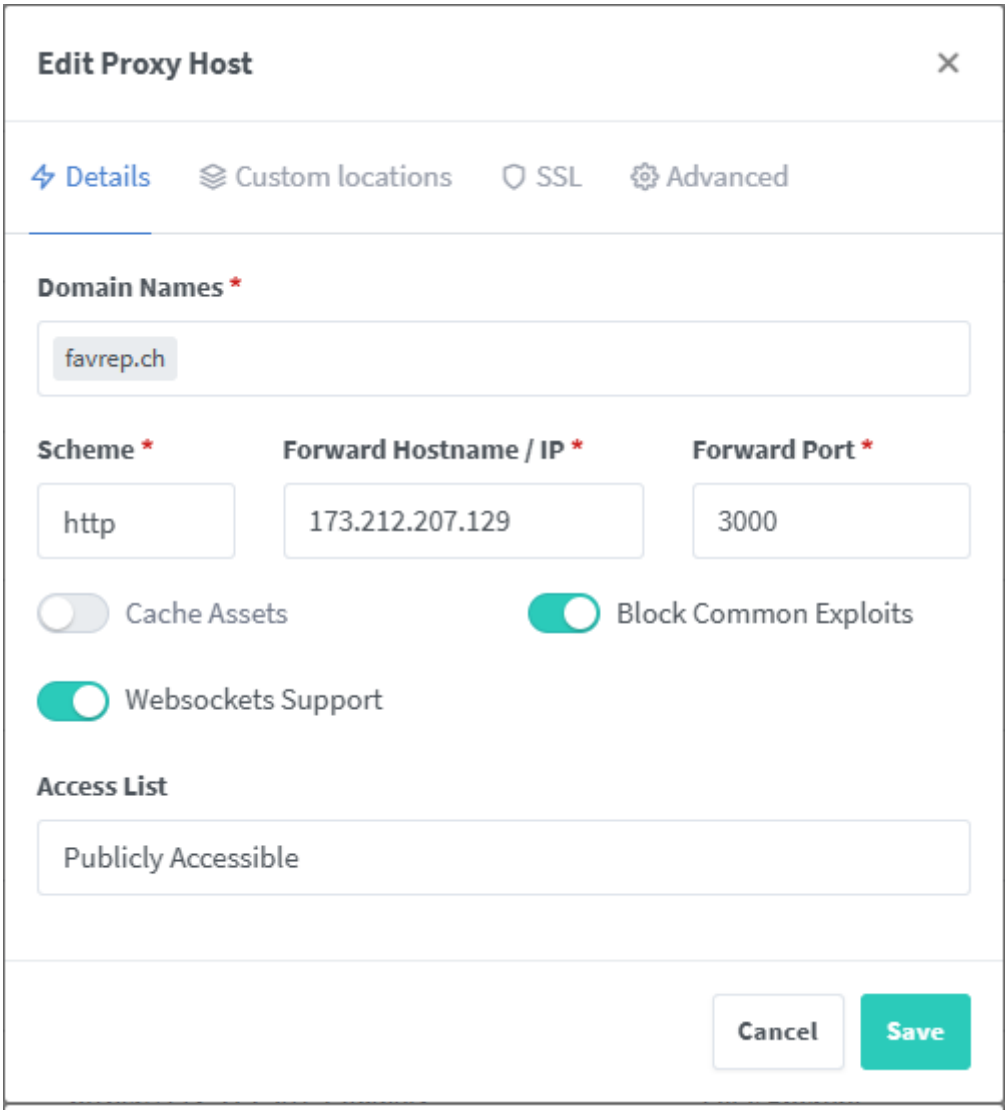
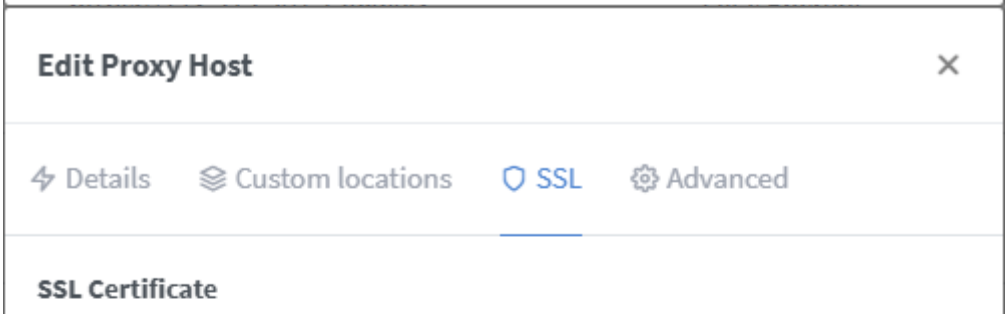
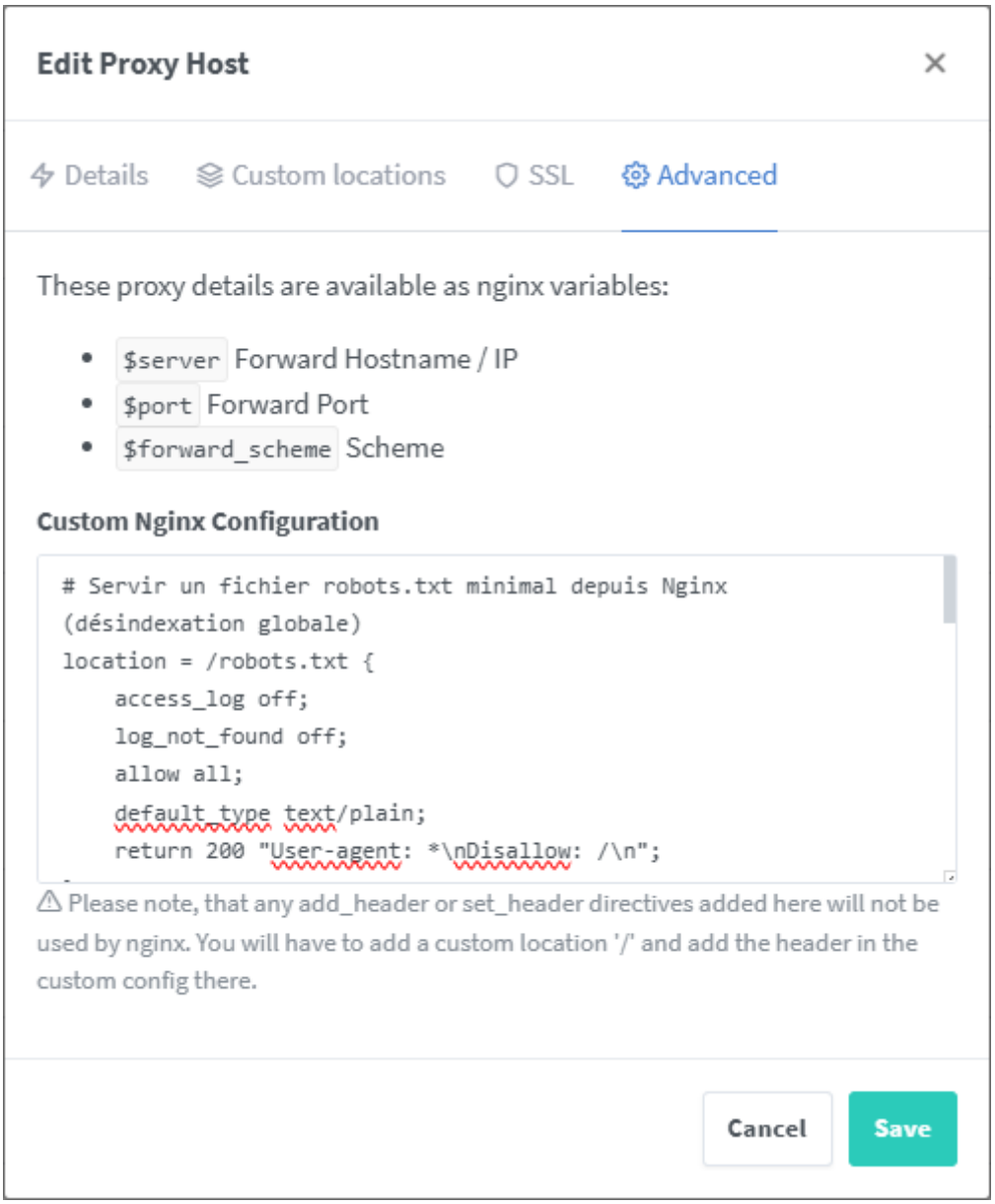
En-têtes de sécurité et robots.txt

Ce guide présente les en-têtes de sécurité recommandés pour renforcer la protection de votre site web hébergé sur Nginx Proxy Manager. L'implémentation de ces en-têtes aide à prévenir diverses attaques, notamment les attaques XSS, le sniffing de type MIME et d'autres menaces liées à la sécurité.

1. Configurations








```
# Servir un fichier robots.txt minimal depuis Nginx (désindexation globale)
location = /robots.txt {
    access_log off;
    log_not_found off;
    allow all;
    default_type text/plain;
    return 200 "User-agent: *\nDisallow: /\n";
}
# Active la protection contre les attaques XSS (Cross-Site Scripting) dans le navigateur web, en bloquant les scripts potentiellement malveillants.
more_set_headers "X-XSS-Protection: 1; mode=block";
# Empêche le navigateur d'effectuer une détection automatique du type de contenu, ce qui empêche certaines attaques de type MIME Sniffing.
more_set_headers "X-Content-Type-Options: nosniff";
```

```
# Indique aux moteurs de recherche de ne pas indexer et suivre les liens de cette page.
# Mettre index et follow pour un site web devant être référencer.
more_set_headers "X-Robots-Tag: noindex, nofollow";
# Contrôle les informations de référent envoyées lors de la navigation entre sites web, évitant ainsi les fuites d'informations sensibles.
more_set_headers "Referrer-Policy: no-referrer-when-downgrade";
# Force le navigateur à utiliser HTTPS pour les ressources chargées, au lieu d'HTTP, améliorant ainsi la sécurité.
more_set_headers "Content-Security-Policy: upgrade-insecure-requests";
# Désactive la fonctionnalité d'intérêt des cohorts, qui peut avoir des implications sur la confidentialité des données.
more_set_headers "Permissions-Policy: interest-cohort=()";
# Empêche le chargement de cette page dans un iframe, sauf si le site est de la même origine, évitant ainsi les attaques de type Clickjacking.
more_set_headers "X-Frame-Options: SAMEORIGIN";
# Indique qu'aucune politique de domaine croisé n'est autorisée, ce qui empêche certaines attaques impliquant du contenu cross-domain.
more_set_headers "X-Permitted-Cross-Domain-Policies: none";
```

2. Tester vos en-têtes

Pour vérifier la configuration de vos en-têtes de sécurité, vous pouvez utiliser le site suivant : Security Headers

3. Tester votre robots.txt

- Accédez à `https://votre-domaine/robots.txt` et vérifiez que la réponse est 200 et que le contenu est exactement : User-agent: * Disallow: /
- En ligne de commande :

```
curl -si https://votre-domaine/robots.txt
```

Vous dévriez voir une réponse comme celle-ci :

```
HTTP/2 200
server: openresty
date: Mon, 13 Oct 2025 02:38:35 GMT
content-type: text/plain
content-length: 26
```

```
x-xss-protection: 1; mode=block
x-content-type-options: nosniff
x-robots-tag: noindex, nofollow
referrer-policy: no-referrer-when-downgrade
content-security-policy: upgrade-insecure-requests
permissions-policy: interest-cohort=()
x-frame-options: SAMEORIGIN
x-permitted-cross-domain-policies: none
strict-transport-security: max-age=63072000; preload

User-agent: *
Disallow: /
```

Source

- MDN Web Docs — HTTP headers
- MDN Web Docs — Practical implementation guides: Robots.txt

Remerciement à Alix et Julien d'avoir fait des recherches à ce sujet pour leurs projets respectifs et de m'avoir donné l'astuce □□

Révision #11
Créé 2025-04-02 19:15:58 CEST par Philippe Favre
Mis à jour 2025-10-13 04:43:14 CEST par Philippe Favre