SECURITY.TXT (RFC 9116)

Le Concept : RFC 9116 (La Théorie)

Dans le Far West d'Internet, quand un chercheur en sécurité (un "White Hat") découvre une faille sur votre site, il se pose une question critique : **"Qui dois-je prévenir ?"**

Souvent, il ne trouve qu'un formulaire de contact commercial ou une adresse support@ qui ne répond jamais.

Résultat ? La faille reste ouverte, ou pire, elle est publiée sur Twitter sans préavis.

La **RFC 9116** standardise la réponse à ce problème. Elle définit un fichier texte simple, security.txt, placé à un endroit prévisible (/.well-known/security.txt).

C'est l'équivalent numérique de l'étiquette **"EN CAS D'URGENCE, APPELEZ CE NUMÉRO"** sur un tableau électrique. Il indique clairement :

- Où envoyer le rapport de faille.
- Quelle clé PGP utiliser pour chiffrer l'échange.
- Quelle est la politique de divulgation.

L'Origine : L'Affaire "Pablo" (Storytelling)

L'implémentation de ce standard sur mon infrastructure ne vient pas d'une exigence bureaucratique, mais d'une **anomalie observée dans les logs**.

Lors d'une routine de "Log Hunting" (analyse des logs d'accès Nginx), une erreur récurrente a attiré mon attention : des dizaines de requêtes **404 Not Found** ciblant précisément le chemin /.well-known/security.txt.

L'analyse des IPs sources a révélé qu'il ne s'agissait pas de script-kiddies ou de botnets russes, mais d'adresses appartenant à **Palo Alto Networks** (surnommé affectueusement *"Pablo Alto", vous avez l'image d'un cartel mexicain de la sécurité* ?!).

Leurs scanners de sécurité (Cortex Xpanse) parcourent le web pour cartographier la surface d'attaque des entreprises.

Le constat était double

Le Bruit : Ces 404 polluaient nos logs et déclenchaient potentiellement des faux positifs sur Fail2Ban.

La Réputation : Ne pas avoir ce fichier renvoie l'image d'une administration système négligente.

j’ai donc décidé de transformer cette erreur 404 en code 200. "Pablo" cherchait une porte, je lui est construit une sonnette.

Le Contenu du Fichier (La Structure)

Le fichier doit respecter une syntaxe stricte. Voici un exemple de modèle a déployer.

Chemin fichier physique : /var/www/html/.well-known/security.txt

```
# Notre politique de sécurité
Contact: mailto:admin@domain.ext
Contact: https://matrix.to/#/@admin.domain:matrix.org

# Pour chiffrer vos communications (Optionnel mais recommandé)
Encryption: https://domain.ext/pgp-key.txt

# Politique de remerciement (Hall of Fame)
Acknowledgments: https://domain.ext/hall-of-fame.html

# Langue préférée
Preferred-Languages: fr, en

Canonical: https://www.domain.ext/.well-known/security.txt

# Politique officielle (Scope, Bounty...)
Policy: https://domain.ext/security-policy.html

# OBLIGATOIRE : Date d'expiration du fichier (Format ISO 8601)
# Pensez à mettre une alerte dans votre calendrier pour le renouveler !
Expires: 2026-12-31T23:59:00z
```

> **Note :** Le champ Expires est obligatoire. C'est pour éviter que des chercheurs ne contactent un admin qui a quitté la boîte depuis 5 ans. Si la date est passée, le fichier est considéré comme invalide.

Implémentation Nginx (La Pratique)

Pour que "Pablo" (et les autres scanners) trouvent le fichier, la RFC impose qu'il soit accessible via le chemin /.well-known/security.txt.

Nous allons configurer Nginx pour servir ce fichier, peu importe où il est rangé sur le disque.

Fichier de conf : /etc/nginx/sites-available/mon-site (ou default)

Ajoutez ce bloc à l'intérieur de votre bloc server { ... } :

```
# --- RFC 9116 : SECURITY.TXT ---
# Redirection du chemin racine (optionnel, pour les paresseux)
location /security.txt {
    return 301 /.well-known/security.txt;
}

# Le chemin officiel
location /.well-known/security.txt {
    # Alias vers le fichier réel (changez le chemin selon votre structure)
    alias /var/www/html/.well-known/security.txt;

    # Sécurité : On autorise tout le monde à le lire (c'est le but)
    allow all;

    # Type MIME correct (texte brut)
    default_type text/plain;
}
```

Vérification

```
sudo nginx -t
```

Reload

```
sudo systemctl reload nginx
```

Test curl

```
curl -I https://nosec.fr/.well-known/security.txt
```

Doit retourner : HTTP/2 200 et Content-Type: text/plain.