Engineer Logiciels

Notre façon de travailler en sécurité

Comment Engineer ApS construit et exploite en sécurité : nos pratiques, le durcissement de ce site et le signalement des vulnérabilités.

La sécurité n'est pas une prestation que nous vendons séparément. C'est la manière de réaliser le travail : la modélisation des menaces fait partie de la conception, le durcissement fait partie de la construction et les preuves font partie de la transmission.

Concevez les contrôles dès le départ

Intégrez l'accès, le déploiement et la reprise à l'ingénierie dès le départ.

Découvrir nos prestations Contactez-nous →

Examinez ce qui étaye une affirmation de sécurité

Demandez la configuration et les tests qui étayent une affirmation de sécurité, avec un périmètre explicite.

Études de cas Contactez-nous →

Comment nous construisons

- **Le moindre privilège comme règle de base.** Les services s'exécutent sous leurs propres utilisateurs non privilégiés, avec l'accès le plus restreint permettant leur fonctionnement. Les limites d'accès sont conçues pour limiter la propagation entre composants si l'un d'eux est compromis.
- **Les secrets n'atteignent jamais le dépôt.** Les identifiants d'accès résident dans un coffre de secrets ou dans la couche de déploiement, jamais dans le code source ni dans un artefact de compilation. Une analyse automatique s'exécute à chaque commit, et toute détection bloque le commit.
- **L'infrastructure décrite sous forme de code.** Les serveurs sont construits à partir d'une description qui peut être lue, examinée et reconstruite ; la configuration d'une machine est donc un document versionné plutôt qu'un souvenir dans la tête de quelqu'un.
- **Des dépendances aux versions fixées et auditées.** Les composants d'un système sont consignés, et la liste est confrontée aux avis de sécurité publiés plutôt que jugée fiable parce qu'elle fonctionnait le mois dernier.

- **Le chiffrement en transit, partout.** Les certificats sont délivrés et renouvelés automatiquement, et le renouvellement est supervisé ; la cause habituelle d'un échec TLS est en effet un certificat que personne ne surveillait.
- **Des sauvegardes automatisées avec des tests de restauration distincts.** Une sauvegarde reste une affirmation tant que personne n'a restauré de données à partir de celle-ci. Nous implémentons des procédures distinctes pour les sauvegardes et les tests de restauration ; la vérification repose sur les données effectivement restaurées et contrôlées.

Chaque point ci-dessus fait partie **d'une journée de travail ordinaire** — pas d'une phase ajoutée à la fin.

Convenez du périmètre de sécurité, des responsabilités et des vérifications dans le cadre de la conception.

Comment ce site est construit

Nous appliquons les mêmes exigences à notre propre site, et vous pouvez vérifier chaque ligne depuis votre navigateur.

- **Aucun composant applicatif côté serveur, aucune base de données, aucune connexion administrateur.** Chaque page est un fichier statique généré avant la réception de la requête. Cela évite les sessions applicatives et les points d'accès exécutant des requêtes sur une base de données, ce qui réduit la surface d'attaque applicative du site.
- **Aucun JavaScript exécutable.** Les pages, les menus et les liens utilisent HTML et CSS natifs ; les données structurées sont du JSON-LD inerte. Aucun script du navigateur n'effectue de requête ni ne stocke de préférence.
- **Une Content Security Policy stricte**, déclarée dans la page elle-même, avec `form-action 'none'` — le site n'envoie aucun formulaire nulle part.
- **Subresource Integrity sur la feuille de style.** L'ensemble CSS reçoit une empreinte et un hachage lors de la compilation ; votre navigateur peut ainsi vérifier qu'il est arrivé sans modification.
- **Aucune requête vers des tiers.** Aucun outil d'analyse d'audience tiers, aucun service de polices tiers ni composant cartographique tiers, et aucun bandeau de consentement. Ouvrez le panneau réseau de votre navigateur et comptez les hôtes : il n'y en a qu'un, le nôtre.
- **Aucun cookie.** Tout ce qui est ou n'est pas stocké figure sur la page Cookies.

- **Une politique de signalement lisible par les machines** à `/.well-known/security.txt` , pour qu'une personne découvrant une vulnérabilité sache où l'envoyer sans demander.

Comment il est exploité

Un petit serveur à Toronto, décrit sous forme de code, avec un journal d'accès masqué à l'écriture et conservé environ trente jours. La page des crédits nomme chaque logiciel utilisé ; la page de confidentialité précise exactement ce qu'il enregistre en répondant.

Signaler une vulnérabilité

Si vous avez trouvé une faiblesse dans ce site ou dans un système que nous exploitons, écrivez à welcome@engineer.company en décrivant votre découverte et la manière de la reproduire. Nous confirmerons la réception, vous indiquerons nos actions et vous créditerons si vous le souhaitez. Il n'y a ni programme de primes ni menace juridique — simplement une adresse qui joint une ingénieure.

Vous souhaitez appliquer la même rigueur à vos systèmes ?
demandez-nous de les examiner.

Cette page présente des affirmations **que vous pouvez vérifier** — depuis le navigateur dans lequel vous la lisez.

Pas de backend, pas de cookies, zéro JavaScript exécutable et une politique déclarée dans la page elle-même.

Demandez-nous d'examiner la vôtre Nos prestations →

Identifiez ce qui doit être protégé

Commencez par les actifs, les frontières de confiance et les autorisations du système à protéger.

Consulter les tarifs Contactez-nous →

Convenez d'une mission de sécurité ciblée

Discutez d'un examen ou d'une mise en œuvre ciblés, avec des contrôles concrets et des vérifications d'acceptation.

Découvrir nos prestations

welcome@engineer.company +45 50 10 61 84 Copenhague, Danemark

Emportez cette page avec vous

Fiche de contact

Créé avec soin et amour par D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

Droit d'auteur © 2025 – aujourd'hui · Engineer ApS · Tous droits réservés.

Imprimé depuis https://software.engineer.company/fr/security/