Engineer

HSTS

次回もHTTPSを使うとブラウザーが記憶する規則

どのようなものか

HTTP Strict Transport Securityは、ほぼ常にHSTSと略されます。指定した秒数の間、サイトへHTTPSだけで接続し、証明書が不正なら接続を無条件に拒否するよう、ブラウザーへ伝える応答ヘッダーです。名前は `Strict-Transport-Security` で、2012年11月からRFC 6797が定義しています。自ら暗号化するものではなく、HTTPSでの接続に関してブラウザーが許す動作を変えます。

典型的な値には、二つのディレクティブがあります。

```
Strict-Transport-Security: max-age=31536000; includeSubDomains
```

`max-age` は、ブラウザーが規則を覚えておく長さで、ここでは一年です。`includeSubDomains` を付けると、規則はホストの下のすべての名前へ及び、`api.example.com` や `old.example.com` も、 `example.com` が求めた規則を引き継ぎます。多くの設定には、三つ目のトークン `preload` もあります。RFCではなく、hstspreload.orgが運営するプリロードリストに由来し、サイト名をブラウザーへ組み込むことへの、サイト側の同意です。

HTTPSで保護された応答で一度このヘッダーを見たブラウザーは、期限まで、そのホストに対して三つのことをします。要求がマシンから出る前に、すべての `http://` アドレスを `https://` へ書き換えます。証明書のどんなエラーも致命的なものとして扱い、無視して進むボタンは出しません。そして、再びヘッダーが届くたびに、時計を初めから数え直します。 `max-age=0` を含む応答は、そのホストの規則を忘れさせます。

よく比較されるのは、ポート80から443への転送です。転送は機能しますが、その前に、ブラウザーは最初の要求を平文で送っています。同じネットワーク上の誰でも、自分で応答して、暗号化されていないサイトのコピーへ訪問者をとどめ

られます。HSTSは、初回を除くすべての訪問で、この平文の要求をなくします。もう一つ近いものは、Content Security PolicyのUpgrade Insecure Requestsディレクティブです。ページが自分のスクリプトや画像を求める要求をHTTPSへ切り替えますが、ほかのサイトの `http://` リンクから来る訪問者は、リダイレクトに任せます。

概念を判断に生かす

「HSTSとは？」を実際の判断に生かしましょう。その概念を、開発または購入するシステムに結び付けて考えるお手伝いをします。 アカウント、パスワード、権限を整理し、業務に必要なアクセスだけを付与し、退職者の権限は当日に解除します。

サービスを見る お問い合わせ →

用語の背後にある前提を検証する

解説とリンク先の実績を参考に、プロジェクトについての質問を整理してください。ツールを選ぶ前に、前提を一緒に検討できます。 Linuxサーバーを守る1,000件のRBACルールを作成 — 1,000 セキュリティ上の穴を塞ぐアクセスルール

Linuxサーバーを守る1,000件のRBACルールを作成 お問い合わせ →

なぜ重要なのか

HSTSを有効にする費用は、設定一行です。本当に決めるべきなのは、どこまで進めるかです。段階を上げるほど、多くの訪問者を守り、元へ戻しにくくなります。

ヘッダーだけでは、RFC 6797自身が挙げる隙間が残ります。そのサイトを一度も見ていないか、覚えていたポリシーが失効したブラウザーは、なお平文の要求から始めます。長い `max-age` は、後者の隙間を縮めるため、一年が一般的な設定です。前者を閉じるのは、プリロードリストだけです。Chromeにホスト名を組み込み、Firefox、Safari、Edgeも、それを基にしたリストを持つため、インストール直後のブラウザーでも、HTTPSを使うと分かっています。

リストへの登録には、多くの条件があります。基底ドメインの有効な証明書、同じホストにとどまるHTTPからHTTPSへの転送、全サブドメインでのHTTPS、少なくとも31536000秒の `max-age` と `includeSubDomains` 、 `preload` を含むヘッダーが必要です。保守者は、一般公開していない内部サブドメインも対象だと明言します。判断の決め手になるのは、リストから外れるときの負担です。削除がChromeの更新を通じて利用者へ届くまで数か月かかり、ほかのブラウザーには保証がありません。同サイトは、削除を「遅く、苦痛を伴う」ものと説明します。

したがって、選択は、階段を一段ずつ上がる形になります。

- `includeSubDomains` **なしのヘッダー**は、すでにHTTPSを提供するほぼすべてのサイトに適します。TLSの現行ベストプラクティスRFC 9325は、ウェブサーバーが送ることを推奨します。
- `includeSubDomains` **の追加**は、ドメイン配下の名前をすべて洗い出し、事業者や別の部署が動かすものも含め、各々がHTTPSで応答すると確認した後なら安全です。
- **プリロード**が適するのは、配下のどの名前も、二度と平文HTTPを必要としないと、所有者が確信するドメインです。ログイン用、決済用、自分のDNSをすべて管理する会社などです。キャンペーンツールが近いうちにCNAMEを求めるかもしれないマーケティング用ドメインや、事務機器と共有するドメインは、適しません。

リストの保守者は、段階を進める速さも示します。`max-age`を300秒、次に一週間、次に一か月とし、各段階で`includeSubDomains`を使い、前の有効期間を丸ごと待ってから次へ進みます。ブラウザーは、最後に見た値を覚えるため、五分の段階での間違いが続くのは五分です。

実際の活用例

専門職や組織に関するデータを持つ、ある基盤で、同社はセキュリティエンジニアリングを担当し、すべての層を強化しました。HSTSも、その一つです。Next.jsのプロキシーミドルウェア`proxy.ts`で、応答にHSTSを載せます。要求ごとに新しいnonceを持つContent Security Policyと`strict-dynamic`、SameSite Cookieも併用します。プロキシーミドルウェアは、一致する経路の前段で動くため、ブラウザー向けの規則を、各ページへ散らさず、一つのファイルへまとめられます。

同じ業務は、HSTSの限界も示します。前提は、クライアントを敵対的と考えることでした。ブラウザーが強制することは、ブラウザーを操作する本人なら無効にできます。HSTSは、誠実な訪問者を、敵対的なネットワークから守ります。敵対的なクライアントには何もしないため、対策はその下へ続きました。APIには、レート制限、CORS、要求サイズの制限、データベースへ届く前の入力検証、セキュリティに関わる事象の記録を加えました。APIは、アプリケーションの関数だけを実行できるロールでデータベースへ接続し、権限は、要求のたびにサーバーで再検査しました。フロントエンドの制御は、利用体験のためだけのものと扱いました。この構成でHSTSは、通信自体へ手が届かないようにし、最も安く追加できる層になります。

問題が起きるところ

設定にはヘッダーがあるのに、ブラウザーは、ないかのように動きます。 通常は転送である平文HTTPの応答か、ブラウザーが証明書を拒否したHTTPS応答に載せています。RFC 6797は、どちらも無視するようブラウザーへ求め、平文応答には一切載せないようサーバーへ求めます。似た誤りとして、`example.com` を直接 `https://www.example.com` へ送り、`www` だけにヘッダーを載せると、基底ドメインとほかの名前を覆えません。転送も含め、すべてのHTTPS応答に載せ、両方の入口を調べます。

```
curl -sI https://example.com/ | grep -i strict-transport-security
curl -sI http://example.com/ | grep -i strict-transport-security
```

一行目は、ヘッダーを出力するはずです。二行目は、転送の応答であり、何も出力しないはずです。

本サイトを訪れた全員に対して、ドメイン配下の古いサービスが読み込めなくなります。 基底ドメインの `includeSubDomains` は、事業者がホストする稼働状況ページ、プリンターの管理画面、証明書が一度もなかったイントラネットの道具など、下のすべての名前へ及びます。その訪問者には、回避できない接続エラーが出る一方、本サイトを開いたことのない同僚は、問題なく接続できるため、報告を再現しにくくなります。誰も一覧を作る前に、ディレクティブを出してしまいました。まず、DNSゾーンと、公開証明書が発行された名前を記録する証明書透明性ログから、一覧を作ります。各名前をHTTPSか別のドメインへ移し、それから、短い `max-age` で追加します。

スタッフが無視して進めた警告が、壁になります。 自己署名証明書のステージング、訪問者のノートパソコンが信頼しない内部認証局、週末に期限が切れた証明書は、HSTS以前なら、どれも先へ進める警告ページでした。導入後のブラウザーは、先へ進む手段を用意しません。RFC 9325は、HSTSでかえって安全性が悪化し得る例として、自己署名証明書を挙げます。ステージングに広く信頼される証明書か、ポリシーの届かないドメインを与え、証明書更新の失敗は、今や訪問者に選択の余地なくサイトを停止させるものとして扱います。

プリロードしたドメインで、平文HTTPが再び必要になります。 HTTPしか話さない機器、独自ドメイン機能で証明書を提供しない事業者、買収で同じ名前の配下へ入る別のホストなどです。出荷済みのブラウザーには、すでに項目が入っており、サーバーにできることは限られます。削除の依頼は、ブラウザーが更新され

て初めて利用者へ届きます。新しい名前を別ドメインの下へ置き、一年の `max-age` と `includeSubDomains` で長く運用しても問い合わせが生じなかった後にだけ、プリロードします。

参考資料

- RFC 6797：HTTP Strict Transport Security（HSTS） — ヘッダーと二つのディレクティブ、ブラウザーが従わなければならない場合と、無視しなければならない場合の厳密な条件です。
- HSTSプリロードリストへの登録 — 掲載の条件、保守者が勧める段階導入、脱退の費用です。
- RFC 9325、第3.2節 — HSTSに関するTLSの現行ベストプラクティスです。クライアントとサーバーは対応必須で、サーバーによる送信が推奨されます。
- Upgrade Insecure Requests — ページ自身の要求をHTTPSへ切り替えるW3Cのディレクティブと、HSTSとの役割分担です。

用語集

実際の制約に合わせて選ぶ

デプロイの流れ、運用上のリスク、緊急対応の担当者をお知らせください。検査、復旧手段、明確な責任の所在を伴う変更の範囲を定められます。

料金を見る お問い合わせ →

用語を実務に結び付ける

「HSTSとは？」がお客様の組織にどう当てはまるかをご相談ください。現在のシステムの具体例が一つあれば、話を始められます。

サービスを見る

welcome@engineer.company +45 50 10 61 84 デンマーク、コペンハーゲン

このページを持ち歩く

連絡先カード

心を込めて丁寧に制作： D.

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

著作権 © 2025 – 現在 · Engineer ApS · すべての権利を留保します。

印刷元： https://engineer.company/ja/glossary/hsts/