Engineer

HSTS

Regla del navegador para usar HTTPS en futuras visitas

Qué es

HTTP Strict Transport Security, casi siempre abreviado HSTS, es una cabecera de respuesta que indica al navegador que acceda a un sitio solo por HTTPS durante un número determinado de segundos y rechace por completo la conexión si el certificado es incorrecto. La cabecera se llama `Strict-Transport-Security` y la define el RFC 6797 desde noviembre de 2012. No cifra nada por sí misma: cambia lo que un navegador está dispuesto a hacer sobre HTTPS.

Un valor típico lleva dos directivas:

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

`max-age` indica cuánto conserva el navegador la regla, aquí un año. Con `includeSubDomains`, la regla alcanza todos los nombres bajo el host, de modo que `api.example.com` y `old.example.com` heredan lo que pidió `example.com`. Un tercer token, `preload`, aparece en muchas configuraciones. Procede de la lista de precarga gestionada en hstspreload.org, no del RFC, y expresa el consentimiento del sitio para incorporar su nombre a los navegadores.

Una vez vista la cabecera en una respuesta segura, el navegador hace tres cosas para ese host hasta que vence el plazo. Reescribe cada dirección `http://` como `https://` antes de que salga una solicitud de la máquina. Considera fatal cualquier error de certificado, sin botón para continuar de todos modos. Y reinicia el contador cada vez que vuelve a recibir la cabecera. Una respuesta con `max-age=0` hace que olvide el host.

La comparación habitual es una redirección del puerto 80 al 443. Funciona, pero solo después de enviar la primera solicitud en claro; cualquiera en la misma red puede responder por su cuenta y mantener al visitante en una copia no cifrada del sitio. HSTS elimina esa solicitud en claro de todas las visitas posteriores a la primera. La otra práctica próxima es Upgrade Insecure Requests, una directiva de la política de seguridad de contenidos: convierte a HTTPS las solicitudes de scripts e imágenes de la propia página y deja que la redirección se encargue de quien llega desde un enlace `http://` ajeno.

Utilice el concepto en una decisión

Aplique «¿Qué es HSTS?» a una decisión real. Podemos ayudarle a relacionar el concepto con el sistema que está construyendo o comprando. Cuentas, contraseñas y permisos organizados para que cada persona tenga exactamente el acceso que necesita y lo pierda el mismo día que deja la empresa.

Explore nuestros servicios Póngase en contacto →

Compruebe los supuestos detrás del término

Utilice la explicación y los trabajos enlazados para formular preguntas sobre su proyecto. Podemos examinar los supuestos con usted antes de elegir una herramienta. 1.000 reglas RBAC para servidores Linux — 1,000 reglas de acceso escritas para cerrar una brecha de seguridad

1.000 reglas RBAC para servidores Linux Póngase en contacto →

Por qué importa

Activar HSTS cuesta una línea de configuración. La decisión real es hasta dónde llevarlo, porque cada paso protege a más visitantes y resulta más difícil de deshacer.

La cabecera por sí sola deja una brecha que el propio RFC 6797 identifica. Un navegador que nunca ha visto el sitio, o cuya política almacenada ha caducado, sigue empezando con una solicitud en claro. Un `max-age` largo reduce la segunda mitad de esa brecha; por eso un año es el ajuste habitual. La primera solo se cierra mediante la lista de precarga: nombres compilados dentro de Chrome, con listas derivadas que mantienen Firefox, Safari y Edge, de modo que un navegador recién instalado ya sabe que debe usar HTTPS.

La lista exige mucho. El dominio base necesita un certificado válido, una redirección de HTTP a HTTPS que permanezca en el mismo host, todos los subdominios servidos por HTTPS y una cabecera con `max-age` de al menos 31.536.000 segundos, `includeSubDomains` y `preload`. Sus responsables aclaran que esto incluye los subdominios internos que el público nunca ve. Salir es lo que termina de decidir la cuestión: la retirada tarda meses en llegar a los usuarios mediante las actualizaciones de Chrome, no hay garantías de retirada en los demás navegadores y el sitio la describe como «lenta y dolorosa».

La decisión consiste, por tanto, en subir una escalera, peldaño a peldaño:

- **La cabecera sin** `includeSubDomains` sirve para casi cualquier sitio que ya ofrece HTTPS. El RFC 9325, la recomendación vigente para TLS, dice que los servidores web deberían enviarla.
- **Añadir** `includeSubDomains` es seguro cuando se han enumerado todos los nombres bajo el dominio y confirmado que cada uno responde por HTTPS, incluidos los gestionados por un proveedor u otro departamento.
- **La precarga** encaja con un dominio cuyos propietarios saben con certeza que ningún nombre subordinado necesitará HTTP en claro otra vez: un dominio de acceso, uno de pagos, una empresa que controla todo su DNS. Un dominio de marketing donde una herramienta de campaña pronto podría pedir un CNAME es mal candidato, como cualquier dominio compartido con equipos de oficina.

Los responsables de la lista también marcan el ritmo de la subida: `max-age` de 300 segundos, luego una semana y después un mes, siempre con `includeSubDomains`, esperando a que transcurra toda la duración de una etapa antes de empezar la siguiente. El navegador conserva el último valor visto, así que un error en el escalón de cinco minutos cuesta cinco minutos.

Cómo se aplica en la práctica

En una plataforma con datos profesionales y organizativos, la empresa realizó labores de ingeniería de seguridad y endureció todas las capas, entre ellas HSTS. En el middleware proxy de Next.js, `proxy.ts`, la respuesta lleva HSTS junto con una política de seguridad de contenidos con un nonce nuevo en cada solicitud y `strict-dynamic`, y cookies SameSite. Un middleware proxy se ejecuta delante de las rutas que selecciona, lo que mantiene las reglas de cara al navegador en un archivo, en vez de repartirlas entre páginas.

Ese mismo trabajo muestra dónde termina HSTS. Partía de que el cliente es hostil: quien controla el navegador puede desactivar cualquier protección que este imponga. HSTS protege a un visitante honesto frente a una red hostil, pero no hace nada frente a un cliente hostil; por eso el trabajo continuó por debajo. La API añadió límites de frecuencia, CORS, límites de tamaño de solicitudes, validación de entradas antes de alcanzar la base de datos y registro de eventos relevantes para la seguridad. Se conectó a la base con un rol que solo podía ejecutar las funciones de la aplicación; los derechos de acceso se verificaban de

nuevo en el servidor en cada solicitud y las barreras del frontend se trataban solo como experiencia de usuario. En esa pila, HSTS mantiene el propio tráfico fuera de alcance y es la capa más barata de añadir.

Dónde surgen los problemas

La cabecera está en la configuración y los navegadores se comportan como si no existiera. Se envía en la respuesta HTTP en claro, normalmente la redirección, o en una respuesta HTTPS cuyo certificado rechazó el navegador. El RFC 6797 ordena a los navegadores ignorarla en ambos casos y a los servidores omitirla por completo en respuestas en claro. Un error relacionado envía `example.com` directamente a `https://www.example.com`, con la cabecera solo en `www`, y deja sin cubrir el dominio base y sus demás nombres. Ponga la cabecera en todas las respuestas HTTPS, incluidas las redirecciones, y compruebe ambas entradas:

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

La primera línea debería mostrar la cabecera. La segunda corresponde a la redirección y no debería mostrar nada.

Un servicio antiguo bajo el dominio deja de cargar para quienes visitaron el sitio principal. `includeSubDomains` en el dominio base cubre cada nombre subordinado: una página de estado alojada por un proveedor, la administración de una impresora, una herramienta de intranet que nunca tuvo certificado. Esos visitantes reciben un error de conexión sin forma de continuar, mientras colegas que nunca abrieron el sitio principal acceden sin problema, por lo que los informes son difíciles de reproducir. La directiva se publicó antes de hacer el inventario. Hágalo primero, a partir de la zona DNS y de los registros de transparencia de certificados, que recogen cada nombre para el que se emitió un certificado público. Mueva cada nombre a HTTPS o a otro dominio y después añada la directiva con un `max-age` corto.

Una advertencia que el personal podía saltarse se convierte en un muro. Preproducción con un certificado autofirmado, una autoridad de certificación interna en la que no confían los portátiles de los visitantes, un certificado que caduca durante el fin de semana: antes de HSTS eran páginas de advertencia con una salida; después el navegador no ofrece ninguna. El RFC 9325 menciona los certificados autofirmados como un caso en que HSTS puede empeorar la

seguridad. Dé a preproducción un certificado de confianza pública o un dominio fuera de la política y trate la renovación como algo cuyo fallo ahora deja el sitio fuera de servicio, sin elección para el visitante.

Un dominio precargado vuelve a necesitar HTTP en claro. Un dispositivo que solo habla HTTP, un proveedor cuya función de dominio propio no sirve certificado, una adquisición que incorpora sus hosts bajo ese nombre. La entrada ya está en navegadores distribuidos y el servidor puede hacer poco: una solicitud de retirada solo llega a los usuarios conforme se actualizan sus navegadores. Sitúe el nombre nuevo bajo otro dominio y precargue solo después de que un período largo con `max-age` de un año e `includeSubDomains` no haya generado tickets.

Lecturas adicionales

- RFC 6797: HTTP Strict Transport Security (HSTS) — La cabecera, sus dos directivas y cuándo exactamente un navegador debe respetarla o ignorarla.
- Solicitud de inclusión en la lista de precarga HSTS — Requisitos, despliegue por etapas aconsejado por sus responsables y coste de salir.
- RFC 9325, sección 3.2 — Recomendación vigente para TLS sobre HSTS: clientes y servidores deben admitirlo y los servidores deberían enviarlo.
- Upgrade Insecure Requests — La directiva W3C que convierte a HTTPS las solicitudes de la propia página y cómo se reparte el trabajo con HSTS.

Glosario

Elija según sus restricciones reales

Describa la ruta de despliegue, los riesgos operativos y quiénes se encargan de las guardias. Podemos definir el alcance de un cambio con comprobaciones, una vía de recuperación y responsabilidades claras.

Vea cuánto cuesta Póngase en contacto →

Haga práctica la terminología

Pregunte cómo se aplica «¿Qué es HSTS?» a su organización. Basta un ejemplo concreto de su sistema actual para iniciar la conversación.

Explore nuestros servicios

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

Llévese esta página

Tarjeta de contacto

Creado con esmero y cariño por D.

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

Derechos de autor © 2025 – actualidad · Engineer ApS · Todos los derechos reservados.

Impreso desde https://engineer.company/es/glossary/hsts/