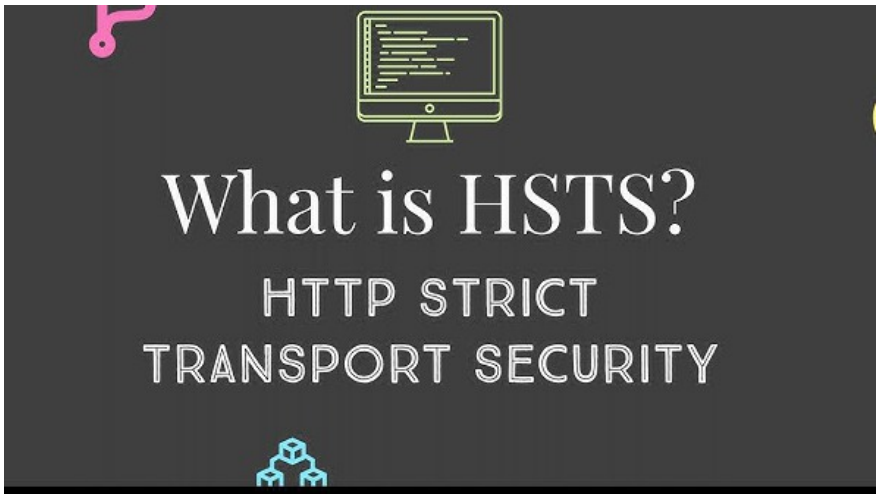
What is HSTS? How does it work?

HSTS is a security standard developed by the Internet Engineering Task Force (IETF) that helps prevent remote attacks that bad guys can perform to steal information.



What is HSTS? HSTS (HTTP Strict Transport Security) is a web security mechanism that ensures that connections between browsers and servers are always made over the HTTPS protocol. This means that all transmitted information is encrypted, helping to prevent 'man-in-the-middle' attacks and other suspicious behavior during data exchange.

In this article, ***TipsMake*** will learn in detail about HSTS, how it works, the benefits it brings as well as the safety aspects of using HSTS in today's internet environment.

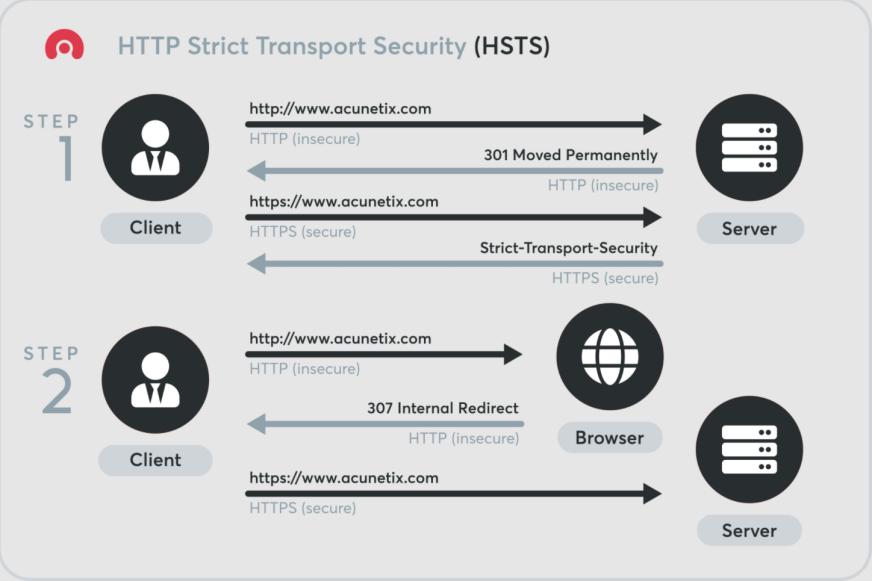
What is HSTS?

HSTS is a security standard developed by the Internet Engineering Task Force (IETF). The primary purpose of HSTS is to prevent remote attacks that bad actors can use to steal sensitive user information. By requiring browsers to only connect to servers over HTTPS, HSTS creates an additional layer of protection for users when they visit websites.

How does HSTS work?

To better understand how HSTS works, let's look at the detailed process that HSTS takes place when a user visits a website.

When you first visit a website that supports HSTS, the server sends an HSTS header to your browser. This header might look like this:



How does HSTS work?

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

1. max-age specifies how long (in seconds) the browser should remember this HSTS rule.
2. includeSubDomains indicates that all subdomains will also have HSTS applied.

After receiving this header, the browser will cache the information and ensure that all subsequent connections to this server take place over HTTPS for the specified period of time.

If a user attempts to access an insecure website via HTTPS, the browser will immediately reject the request and automatically redirect to the HTTPS version of the website instead. This not only protects user data but also prevents external attacks.

By automatically redirecting, HSTS not only enhances security but also creates a better user experience. Users will not have to worry about accidentally entering personal information on an insecure website.

What are the benefits of using HSTS?

HSTS is not only a security tool but also brings many benefits to both users and web developers. Here are some of the prominent benefits that HSTS can bring.

User information security

The biggest benefit that HSTS brings is protecting sensitive user information. When data is transmitted over HTTPS, all information is encrypted, making it more difficult for hackers to steal information.

This is especially important in today's context, when more and more people are engaging in online transactions, from shopping to banking. HSTS helps users feel more secure when providing personal information.

Increase user trust

When users know that the website they are using has HSTS, this creates a greater sense of security and trust. They will feel more comfortable providing personal information, which in turn increases conversion rates and revenue for e-commerce sites.

Implementing HSTS also helps to enhance brand reputation. Organizations and businesses that are committed to protecting customer information will attract more users.

Minimize the risk of attack

HSTS not only helps protect user information but also reduces the risk of attacks. Attackers often take advantage of vulnerabilities in the HTTP protocol to carry out attacks.

By forcing connections to take place over HTTPS, HSTS reduces the vulnerability to attack and protects both users and websites from unnecessary risks.

Easy to integrate into the system

HSTS is easy to integrate into any website. Developers only need to add a few lines of code to their server configuration to enable HSTS. This saves time and effort in implementing security measures.

HSTS works effectively without requiring any changes to the structure of the website. Implementing HSTS has absolutely no impact on website performance and provides clear security benefits.

Is HSTS really safe?

An interesting and necessary question is whether HSTS is actually safe. To answer this, we need to consider several aspects related to the level of safety that HSTS provides.

Strong security features

HSTS is designed with maximum security in mind. By requiring the use of HTTPS, HSTS protects users' personal information from man-in-the-middle attacks. Since information is encrypted, even if someone tries to eavesdrop, they will not be able to read the data content.

HSTS also helps reduce the risk of website spoofing. By automatically redirecting users from HTTP to HTTPS, phishing attacks become much more difficult.

Risk of forgetting or misconfiguring

While HSTS provides many security benefits, it can create vulnerabilities if not configured properly. If a website has an incorrect HSTS configuration or forgets to set up HSTS, this can result in users accidentally accessing an insecure version.

This can have serious consequences, especially if users enter sensitive information. Therefore, developers need to be careful in setting up HSTS and maintaining cache information.

Depends on the browser

HSTS relies on browser support. If a user is using a browser that does not support HSTS, the feature will not work. While most popular browsers today support HSTS, there are some older browsers that do not meet this requirement.

Therefore, in some cases, HSTS may not be enough to protect users. This shows that while HSTS is a useful tool, it is not a perfect solution for every situation.

How to add domains to the HSTS preload list

To add a domain to the HSTS (HTTP Strict Transport Security) preload list, you need to follow these steps:

Necessary requirements

Before adding a domain to the HSTS preload list, make sure your website meets the following requirements:

1. **Valid SSL Certificate** : Website must have a valid and updated SSL certificate.
2. **Redirect from HTTP to HTTPS:** All traffic must be redirected from HTTP to HTTPS.
3. **Applies to domains and subdomains:** The above conditions apply to both the main domain and the subdomains.
4. **HSTS Header:** Provide a Strict-Transport-Security header with the value max-age=31536000; includeSubDomains; preload for the base domain.

Domain addition process

Add HSTS header: Make sure that HSTS header is added to the web server response. For example, in Apache, you can add it to your .htaccess file like this:

Header always set Strict-Transport-Security "max-age=31536000; includeSubDomains; preload"

In Nginx, use:

add_header Strict-Transport-Security "max-age=31536000; includeSubDomains; preload" always;

Submit a request: Once configured, you need to visit hstspreload.org to submit your domain to the HSTS preload list. Enter your domain name and follow the instructions to complete the submission process.

Add HSTS header: Make sure that HSTS header is added to the web server response. For example, in Apache, you can add it to your .htaccess file like this:

How to remove a domain from the HSTS cache on different browsers

Once you've added a domain to the HSTS preload list, there may come a time when you need to remove it from the HSTS cache. Here's how to do this in popular browsers.

Remove HSTS on Google Chrome

To remove a domain from the HSTS cache in Google Chrome, you can follow these steps:

1. Open Chrome and type chrome://net-internals/
2. Here you will see a section that provides information about HSTS. You need to enter the domain name in the "Delete domain security policies" box and press the "Delete" button.
3. Once done, try accessing the website again to check if HSTS has been removed.

Remove HSTS on Mozilla Firefox

If you're using Firefox, the way to clear HSTS is a little different:

Open Firefox and type about:preferences

1. Scroll down to the 'Cookies and Site Data' section and click 'Manage Data'.
2. Search for the domain you want to remove and click "Remove Selected".
3. Once done, restart your browser and check the website again.

Remove HSTS on Microsoft Edge

For Microsoft Edge, you can remove a domain from HSTS by:

1. Open Edge and go to edge://net-internals/
2. Enter the domain name in the "Delete domain security policies" box and click "Delete".
3. Check the website again to make sure HSTS has been removed.

HSTS or HTTP is more secure?

When comparing HSTS and HTTP, it is clear that HSTS has more advantages in terms of security. However, we need to understand more about the differences between these two protocols.

HTTP (HyperText Transfer Protocol) is the protocol for transmitting data on the web. It allows browsers and servers to exchange information with each other. However, one of the biggest problems with HTTP is its lack of security.

Data sent over HTTP is not encrypted, which means that anyone can easily eavesdrop and steal information. Therefore, using HTTP to handle sensitive information is very risky.

When comparing HSTS vs HTTP, it is clear that HSTS is the more secure option. HSTS not only prevents data from being transmitted over HTTP, but also automatically redirects users to the HTTPS version. This not only protects personal information but also creates a better user experience.

While HSTS doesn't replace HTTPS, it is an important tool in a web developer's security toolkit, helping them create a safer environment for users.

Conclude

HSTS is a powerful tool that helps protect user information when accessing websites. With the ability to automatically redirect and enforce the use of HTTPS, HSTS plays an important role in preventing attacks and protecting sensitive data.

You finished reading the article "**What is HSTS? How does it work?**" edited by the **TipsMake** team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.
