

HTTP en detalle

Tarea 1: ¿Que es HTTP(S)?

¿Qué es **HTTP** ? (Protocolo de transferencia de hipertexto) **HTTP** es el protocolo que se utiliza al visitar un sitio web, desarrollado por Tim Berners-Lee y su equipo entre 1989 y 1991. **HTTP** es el conjunto de reglas que se utilizan para comunicarse con servidores web y transmitir datos de páginas web, ya sean HTML, imágenes, vídeos, etc.

¿Qué es **HTTPS**? (Protocolo Seguro de Transferencia de Hipertexto) **HTTPS** es la versión segura de **HTTP** . Los datos **HTTPS** están cifrados, lo que no solo impide que otros vean los datos que recibes y envías, sino que también te garantiza que te estás comunicando con el servidor web correcto y no con alguien que lo suplanta.

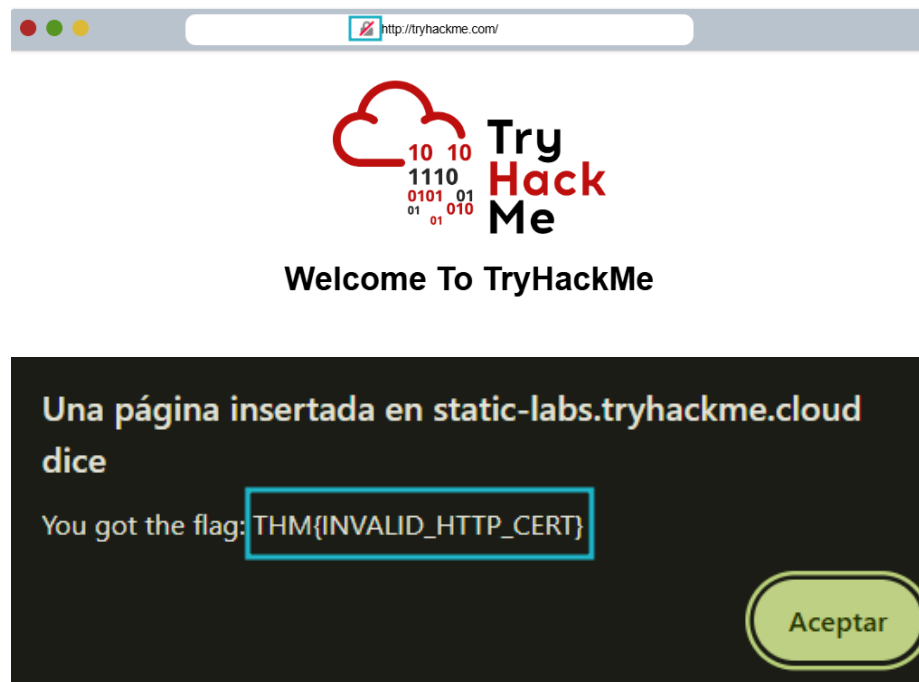
P1: ¿Qué significa HTTP?

R1: HyperText Transfer Protocol / Protocolo de transferencia de hipertexto

P2: ¿Qué significa la S en HTTPS?

R2: secure / seguro

P3: En la página web simulada de la derecha hay un problema. Una vez que lo encuentres, haz clic en él. ¿Qué es la bandera del desafío?



R3: THM{INVALID_HTTP_CERT}

Tarea 2: Solicitudes y Respuestas

P1: ¿Qué protocolo HTTP se está utilizando en el ejemplo anterior?

R1: HTTP/1.1

P2: ¿Qué encabezado de respuesta le dice al navegador cuántos datos esperar?

R2: Content-Length / Longitud de Contenido

Tarea 3: Metodos HTTP

P1: ¿Qué método se utilizaría para crear una nueva cuenta de usuario?

R1: POST

P2: ¿Qué método se utilizaría para actualizar su dirección de correo electrónico?

R2: PUT

P3: ¿Qué método utilizarías para eliminar una imagen que has subido a tu cuenta?

R3: DELETE

P4: ¿Qué método se utilizaría para ver un artículo de noticias?

R4: GET

Tarea 4: Códigos de estado HTTP

P1: ¿Qué código de respuesta podrías recibir si has creado un nuevo usuario o artículo de blog?

R1: 201

P2: ¿Qué código de respuesta podrías recibir si intentas acceder a una página que no existe?

R2: 404

P3: ¿Qué código de respuesta podría recibir si el servidor web no puede acceder a su base de datos y la aplicación falla?

R3: 503

P4: ¿Qué código de respuesta podrías recibir si intentas editar tu perfil sin iniciar sesión primero?

R4: 401

Tarea 5: Encabezados

P1: ¿Qué encabezado le dice al servidor web qué navegador se está utilizando?

R1: User-Agent / Agente de Usuario

P2: ¿Qué encabezado le dice al navegador qué tipo de datos se están devolviendo?

R2: Content-Type / Tipo de Contenido

P3: ¿Qué encabezado le dice al servidor web qué sitio web se está solicitando?

R3: Host / Anfitrión

Tarea 6: Cookies

Probablemente ya hayas oído hablar de las cookies; son simplemente pequeños fragmentos de datos que se almacenan en tu ordenador. Se guardan al recibir un encabezado "Set-Cookie" de un servidor web. Con cada nueva solicitud, enviarás los datos de las cookies al servidor web. Dado que HTTP no registra tus solicitudes anteriores, las cookies pueden usarse para recordarle al servidor web quién eres, algunas configuraciones personales del sitio web o si ya has visitado el sitio web.

P: ¿Qué encabezado se utiliza para guardar las cookies en su computadora?

R: Set-Cookie / Establecer Cookie

Tarea 7: Realizar Solicitudes

Haga clic en el botón "Ver sitio" a la derecha.

Este es un emulador para realizar solicitudes HTTP de demostración . Utilizando lo que aprendiste en las tareas anteriores, puedes usarlo para completar las siguientes preguntas.

P1: Realizar una solicitud **GET** a la página **/room**



R1: THM{YOU'RE_IN_THE_ROOM}

P2: Realice una solicitud **GET a la página /blog** y configure el parámetro **id en 1**
Nota: use el botón de engranaje a la derecha para administrar los parámetros de URI

Parameters

×

Here, you can add key-value pairs to customize your requests.

id

:

1

🗑️

+ Add Param

Save

GET

⌵

http://tryhackme.com/blog?id=1

GET /blog?id=1 HTTP/1.1
Host: tryhackme.com
User-Agent: Mozilla/5.0 Firefox/87.0
Content-Length: 0

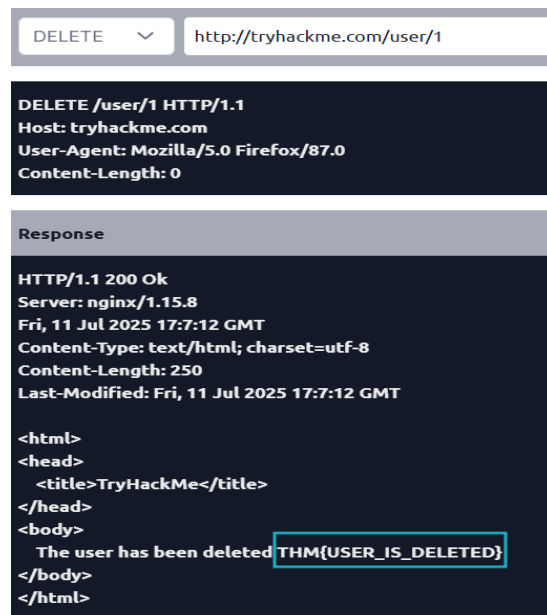
Response

HTTP/1.1 200 Ok
Server: nginx/1.15.8
Fri, 11 Jul 2025 17:6:45 GMT
Content-Type: text/html; charset=utf-8
Content-Length: 250
Last-Modified: Fri, 11 Jul 2025 17:6:45 GMT

<html>
<head>
 <title>TryHackMe</title>
</head>
<body>
 Viewing Blog article 1 THM{YOU_FOUND_THE_BLOG}
</body>
</html>

R2: THM{YOU_FOUND_THE_BLOG}

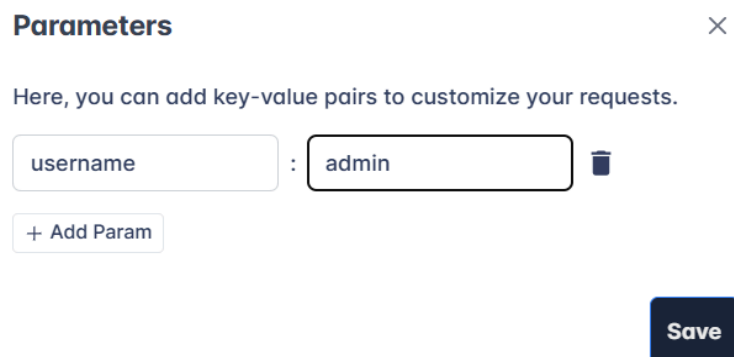
P3: Realizar una solicitud **DELETE** a la página **/usuario/1**



R3: THM{USER_IS_DELETED}

P4: Realice una solicitud **PUT** a la página **/usuario/2** con el parámetro de **nombre de usuario** establecido en **admin**

Nota: Use el botón de engranaje a la derecha para administrar los parámetros del cuerpo



PUT ▼ http://tryhackme.com/user/2

```
PUT /user/2 HTTP/1.1
Host: tryhackme.com
User-Agent: Mozilla/5.0 Firefox/87.0
Content-Length: 14
Content-Type: application/x-www-form-urlencoded

username=admin
```

Response

```
HTTP/1.1 200 Ok
Server: nginx/1.15.8
Fri, 11 Jul 2025 17:7:56 GMT
Content-Type: text/html; charset=utf-8
Content-Length: 251
Last-Modified: Fri, 11 Jul 2025 17:7:56 GMT

<html>
<head>
  <title>TryHackMe</title>
</head>
<body>
  Username changed to admin THM{USER_HAS_UPDATED}
</body>
</html>
```

R4: THM{USER_HAS_UPDATED}

P5: Realice una solicitud **POST** a la página / login con el nombre de usuario thm y la contraseña letmein Nota: Use el botón de engranaje a la derecha para administrar los parámetros del cuerpo

Parameters ×

Here, you can add key-value pairs to customize your requests.

username	:	thm	🗑
password	:	letmein	🗑

+ Add Param

Save

POST ▼ http://tryhackme.com/login

```
POST /login HTTP/1.1
Host: tryhackme.com
User-Agent: Mozilla/5.0 Firefox/87.0
Content-Length: 29
Content-Type: application/x-www-form-urlencoded

username=thm&password=letmein
```

Response

```
HTTP/1.1 200 Ok
Server: nginx/1.15.8
Fri, 11 Jul 2025 17:4:57 GMT
Content-Type: text/html; charset=utf-8
Content-Length: 256
Last-Modified: Fri, 11 Jul 2025 17:4:57 GMT

<html>
<head>
  <title>TryHackMe</title>
</head>
<body>
  You logged in! Welcome Back THM{HTTP_REQUEST_MASTER}
</body>
</html>
```

R5: THM{HTTP_REQUEST_MASTER}

 **LinkedIn:** <https://www.linkedin.com/in/mateo-rodr%C3%ADguez-suar-202695249/>

 **GitHub:** <https://github.com/MaateoSuar>