

Lab: Manipulating WebSocket messages to exploit vulnerabilities

This online shop has a live chat feature implemented using [WebSockets](#).

Chat messages that you submit are viewed by a support agent in real time.

To solve the lab, use a WebSocket message to trigger an `alert()` popup in the support agent's browser.

Click "Live chat" and send a chat message.

Live chat

You: hellop cunt

Hal Pline: Are you sure you want to know the answer t

You: <

Hal Pline: When is your next holiday, I could use a col

CONNECTED: -- Now chatting with Hal Pline --

You: 

Hal Pline: I can hear you, there is no need to shout

Your message:

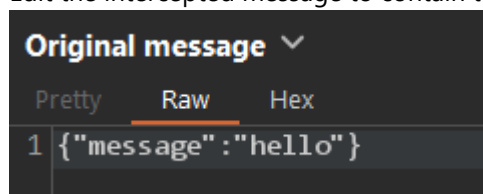
In Burp Proxy, go to the WebSockets history tab, and observe that the chat message has been sent via a WebSocket message.

Using the browser, send a new message containing a `<` character.

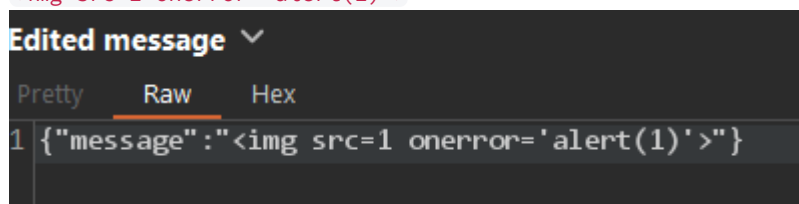
In Burp Proxy, find the corresponding WebSocket message and observe that the `<` has been HTML-encoded by the client before sending.

Ensure that Burp Proxy is configured to intercept WebSocket messages, then send another chat message.

Edit the intercepted message to contain the following payload:



``



Observe that an alert is triggered in the browser. This will also happen in the support agent's browser.

Live chat


You: hellop cunt

Hal Pline: Are you sure you want to know the answer to that?

You: <

Hal Pline: When is your next holiday, I could use a couple of weeks without your stupid questions

CONNECTED: -- Now chatting with Hal Pline --

You: 

Hal Pline: I can hear you, there is no need to shout

Your message:

🌐 0a6c005f03bb90f0c00d7eb500d700e2.web-security-academy.net

1

OK

Send