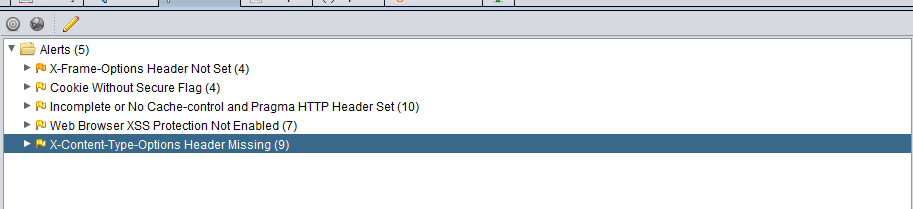
Tim Allen Security Week 4



X-Content-Type-Options Header Missing. Zap had this to say

The Anti-MIME-Sniffing header X-Content-Type-Options was not set to 'nosniff'. This allows older versions of Internet Explorer and Chrome to perform MIME-sniffing on the response body, potentially causing the response body to be interpreted and displayed as a content type other than the declared content type. Current (early 2014) and legacy versions of Firefox will use the declared content type (if one is set), rather than performing MIME-sniffing.

As such I applied the following code and it resolved the error after attacking the site again.

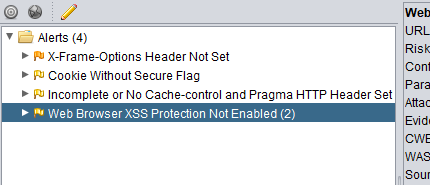
app.Use(async (context, next) =>

{

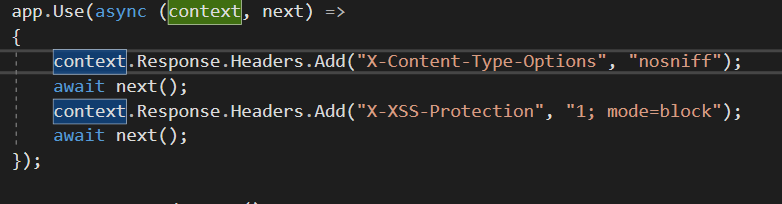
context.Response.Headers.Add("X-Content-Type-Options", "nosniff");

await next();

});



In addition, “Web Browser XSS Protection is not enabled, or is disabled by the configuration of the 'X-XSS-Protection' HTTP response header on the web server” was presented as another error. I applied the following code and interestingly if both protections were enabled zap would failed to attack. Only one could be allowed at a time and for some reason the XSS was not actually applying to the project. The Anti MIME(x-content-type-options) bit did but the anti XSS did not.



In addition ZAP also presented a “A cookie has been set without the secure flag, which means that the cookie can be accessed via unencrypted connections.”

<https://security.stackexchange.com/questions/1518/how-to-ensure-that-cookies-are-always-sent-via-ssl-when-using-asp-net-on-iis-7-5>

<https://www.itbusiness.ca/news/how-to-protect-yourself-against-firesheep-hijacking/15673>

<https://security.stackexchange.com/questions/258/what-are-the-pros-and-cons-of-site-wide-ssl-https>

its very cool that stack exchange has a security version.

Also found this podcast <https://media.grc.com/SN/sn-272-lq.mp3>

I am going to research firesheep more but in essence it seems to be a 0 day bug that allows 3rd party observation of your sites traffic and or CRUD operations being performed through an intercept attack. I am going to hold off on implementing a fix as it seems complicated to implement site wide SSL but I am going to leave my two previous fixes in. So super secure!