Sites vulnerable to XSS can be used to phish Googlebot

One common security vulnerability developers must take into account is the Cross-Site Scripting (XSS) problem. The XSS exploit can be mitigated or defended against by using sanitization procedures that test values of variables in GET and POST requests. Server-side attacks exist, as well, but are well beyond the scope here. Apparently, Googlebot and indexing currently suffers from this vulnerability.

Phishing Googlebot?

Although XSS attacks can be used to deface or sabotage websites, it’s also one method for phishing. An attacker crafts a malicious link and sends users an email with that link to a website vulnerable to the XSS exploit. When a user clicks the malicious link, a script runs when the page loads. Newer versions of Chrome have a XSS Auditor sanity check for urls that might otherwise fool users.

Unfortunately, Googlebot currently operates running Chrome 41, an earlier version of the browser that does not have the XSS Auditor. Does this mean Googlebot is vulnerable to phishing-style URLs where an attacker might inject a malicious SEO attack script? For Google, an attacker might use JavaScript to inject elements into the DOM such as backlinks, and worse, manipulate the canonical.

XSS Googlebot exploit

The proof of concept (PoC) for this attack was published by SEO (and Security Researcher) Tom Anthony depicting the success of the attack method, including evidentiary screenshots of Google Search Console’s URL Inspection Tool

displaying a modified source code as a result of the payload. Keep in mind that running a malicious script in this way has the potential to entirely rewrite the page for indexing.

Tom describes having undertaken the proper steps of vulnerability disclosure, listing a timeline of his communications with Google, and characterizing the responses. At this stage, Tom's disclosure is an iffy prospect because the vulnerability conceivably still works in the wild, despite Google having told him in March it has "security mechanisms in place" for it. Security researchers sometimes publish 0day (unpatched) vulnerabilities in order to prompt companies into action.

Google's response

Tom noted that people see evidence Googlebot has a pending upgrade that would presumably include the XSS Auditor URL filter. Once Googlebot upgrades to a newer instance of Chrome with the XSS Auditor in place, this attack will no longer work. In the meantime, Google can conceivably index and publish malicious links in SERPs that unwitting users of Firefox (which doesn't currently have a XSS Auditor of its own) could conceivably click and get phished.

We received the following statement from Google: "We appreciate the researcher bringing this issue to our attention. We have investigated and have found no evidence that this is being abused, and we continue to remain vigilant to protect our systems and make improvements."

Exploits using XSS techniques are so widespread it's conceivable that it's happening somewhere. It's simultaneously believable that no one other than the researcher has tried it.

How to protect against XSS attacks

To prevent the most common attacks, you need to make sure no malicious code (Javascript, PHP, SQL etc.) gets through to be processed by your application. Use built-in expectations of values such as an assurance that only the exact number and correctly named set of variables are present with every request. You should also encode data-type restrictions to test incoming values before proceeding.

For example, if your application is expecting a number then it should throw an exception, redirect, and maybe temporarily blacklist the IP address if it gets a string value as part of a bad request. The trouble is, there are a lot of fairly popular websites which are vulnerable to this sort of attack because they don't take such steps. It's fairly commonplace for attackers to pick at applications modifying request variable values to look for exploit opportunities.

One of Google's proposals against XSS attacks takes the data-type sanity test as described above to the HTTP response header level with what it is calling: Trusted Types. Although it hasn't yet been widely adopted Trusted Types may eventually serve as an important tactic for protecting your pages. That the vulnerability is not currently patched, however, is why it's iffy to publish 0days. Google is vulnerable to this exploit.

About The Author



Detlef Johnson is Editor at Large for Third Door Media. He writes a column for Search Engine Land entitled "Technical SEO for Developers." Detlef is one of the original group of pioneering webmasters who established the professional SEO

field more than 20 years ago. Since then he has worked for major search engine technology providers, managed programming and marketing teams for Chicago Tribune, and consulted for numerous entities including Fortune 500 companies. Detlef has a strong understanding of Technical SEO and a passion for Web programming. As a noted technology moderator at our SMX conference series, Detlef will continue to promote SEO excellence combined with marketing-programmer features and webmaster tips.

http://feeds.searchengineland.com/~r/searchengineland/~3/6r9XA3FM8pE/sites-vulnerable-to-xss-can-phish-googlebot-316338