(/)

SpiderLabs® Blog (/Resources/SpiderLabs-Blog) Understanding and Discovering Open Redirect Vulnerabilities

April 10, 2017 | Posted By Vladimir Zakharevich | Comments (0)

One of the most common and largely overlooked vulnerabilities by web developers is Open Redirect (also known as "Unvalidated Redirects and Forwards"). A website is vulnerable to Open Redirect when parameter values (the portion of URL after "?") in an HTTP GET request allow for information that will redirect a user to a new website without any validation of the target of redirect. Depending on the architecture of a vulnerable website, redirection could happen after certain action, such as login, and sometimes it could happen instantaneously upon loading of a page.

An example of a vulnerable website link could look something like this: https://www.example.com/login.html?RelayState=http%3A%2F%2Fexample.com%2Fnext

In this example, "RelayState" parameter indicates where to send user upon successful login (In our example it is "http://example.com/next"). If website doesn't validate the "RelayState" parameter value to make sure that target web page is legitimate and intended, attacker could manipulate that parameter to send a victim to a fake page crafted by attacker: https://www.example.com/login.html?RelayState=http%3A%2F%2FEvilWebsite.com

Open Redirect vulnerabilities don't get enough attention from developers because they don't directly damage website and do not allow an attacker to directly steal data that belong to the company. However, that doesn't mean that Open Redirect attacks are not a threat. One of the main uses for this vulnerability is to make phishing attacks more credible and effective.

When an Open Redirect is used in a phishing attack, the victim receives an email that looks legitimate with a link that points to a correct and expected domain. What the victim may not notice, is that in a middle of a long URL there are parameters that manipulate and change where the link will take them. To make identification of the Open Redirect even more difficult, redirection could take place after victim provides login on a legitimate website first. Attackers have found that an effective way to trick a victim is to redirect him to a fake website after they enter their credentials on a legitimate page. The fake website would look identical to a legitimate website, and it would ask the victim to re-enter their password. After the victim re-enters their password it would be recorded by the attacker and victim would be redirected back to a valid website. If done correctly, victim would think that he mistyped password once and would not notice that his username and password were stolen.

Phishing is used in most successful targeted hacks and also regularly in opportunistic attacks. Considering how prominent phishing is in our daily lives, Open Redirect vulnerabilities should not be dismissed.

It would have been unfair to single out any specific website or company as being vulnerable to Open Redirect because so many companies have it. Instead, it's more useful to demonstrate how common those websites are and how easy it is to find them.

Doing a web search is one of the best tools to find Open Redirect on your own website and across a wider Internet.

Google Search allows for a great flexibility in writing search queries, including queries that specifically search through URLs of pages.

The following operators and special symbols allow anyone to craft very targeted Google Searches for finding Open Redirects:

allinurl - operator that tells Google to search within URL for all provided keywords
Example: allinurl:ReturnUrl which searches for web pages that have "ReturnUrl" as part of their URL

site - operator that tells to only return results that are on specific domain or a web site
Example: site:example.com which searches for web pages from example.com

"" - double quotes are a special symbols that used to indicate to search for exact combination of words and symbols within quotes

***** - The asterisk is a wildcard that represents one or more words

Using these allows us to search for certain tell tale signs of potential Open Redirect:

We can look for the general presence of "http" or "https" within parameter area of GET request. For instance:

allinurl:%3Dhttps*
allinurl:%253Dhttps*
allinurl:%3Dhttp*
allinurl:%253Dhttp*

We can also search for specific, common words related to forwarding within parameter area of GET request. For instance:

allinurl:"<keyword>=https"
allinurl:"<keyword>=http"
allinurl:<keyword>=https
allinurl:<keyword>=http
allinurl:<keyword>%3Dhttps
allinurl:<keyword>%3Dhttps*
allinurl:<keyword>%253Dhttps
allinurl:<keyword>%253Dhttps*
allinurl:<keyword>%3Dhttp
allinurl:<keyword>%3Dhttp*
allinurl:<keyword>%253Dhttp
allinurl:<keyword>%253Dhttp*
allinurl:<keyword>

Instead of <keyword>, we would use one of the following words typical of redirects: *RelayState, ReturnUrl, RedirectUri, Return, Return_url, Redirect, Redirect_uri, Redirect_url, RedirectUrl, Forward, ForwardUrl, Forward_URL, SuccessUrl, Redir, Exit_url, Destination*. This is by no means a comprehensive list of keywords. You can find more by analyzing results from the more general queries looking for a URL in the parameter section of the GET request.

For targeted searches, you can add "site:<domain_name>" to the end of your Google Queries. This can help you identify Open Redirect vulnerabilities on your own website.

Using this simple search technique you can find dozens of Open Redirect vulnerabilities within minutes. List of vulnerable websites includes banking websites, websites of international corporations, trusted companies, beloved projects and numerous websites of smaller organizations. As an additional bonus, each time Google's web crawler comes across new website that has Open Redirect, we will get updated results through our queries.

The best way to avoid Open Redirect vulnerability is to avoid redirecting based on parameter controlled by users or supplied through GET method. If redirecting is unavoidable, it can be dealt with by validating a redirect target and sanitizing it using whitelist of approved URLs.

I would encourage you to inform and educate your web developer friends about Open Redirects. You can do it by forwarding this post to them or by finding vulnerabilities using technique mentioned above and reporting directly to company with details. Let's make phishing attacks harder to succeed!

If you would like to share some of your favorite clever Google Queries that you use for vulnerability discovery, please post them in comment section.

Tags: Phishing (/Resources/SpiderLabs-Blog/?page=1&year=0&month=0&tag=Phishing), Security Research (/Resources/SpiderLabs-Blog/?page=1&year=0&month=0&tag=Security+Research), Web Security (/Resources/SpiderLabs-Blog/?page=1&year=0&month=0&tag=Web+Security)

‹ Prev (/Resources/SpiderLabs-Blog/Microsoft-Patch-Tuesday,-April-2017/?page=1&year=0&month=0)

Next › (/Resources/SpiderLabs-Blog/Trustwave-Web-Application-Firewall-Signature-Update-4-47-Now-Available/?page=1&year=0&month=0)

Recent Posts

Spammed JScript Phones Home To Download NemucodAES And Kovter (/Resources/SpiderLabs-Blog/Spammed-JScript-Phones-Home-To-Download-NemucodAES-And-Kovter/)
Jul 25, 2017 | Nicholas Ramos

Petya From The Wire: Detection using IDPS (/Resources/SpiderLabs-Blog/Petya-From-The-Wire--Detection-using-IDPS/)
Jul 13, 2017 | Bryant Smith

ModSecurity Web Application Firewall - Commercial Rules Update(1) (/Resources/SpiderLabs-Blog/ModSecurity-Web-Application-Firewall---Commercial-Rules-Update(1)/)
Jul 12, 2017 | SpiderLabs Research

Microsoft Patch Tuesday, July 2017 (/Resources/SpiderLabs-Blog/Microsoft-Patch-Tuesday,-July-2017/)
Jul 11, 2017 | Karl Sigler

A Computational Complexity Attack against Racoon and ISAKMP Fragmentation (/Resources/SpiderLabs-Blog/A-Computational-Complexity-Attack-against-Racoon-and-ISAKMP-Fragmentation/)
Jul 10, 2017 | Neil Kettle

Stay Connected

Archive

‹ Prev (/Resources/SpiderLabs-Blog/Microsoft-Patch-Tuesday,-April-2017/?page=1&year=0&month=0)

Next › (/Resources/SpiderLabs-Blog/Trustwave-Web-Application-Firewall-Signature-Update-4-47-Now-Available/?page=1&year=0&month=0)

Subscribe Now

Sign up to receive the latest security news and trends from Trustwave.

Email Address Subscribe

No spam, unsubscribe at any time.