SSRF “server-side request forgery”

As seen from the name it is a server-side vulnerability not from the user side, which means that it is targeting the backend of a website and doesn’t need any reaction from a victim.

This vulnerability enables the threat actor to perform an unauthorized action on the server like reading or writing on the internal resources or requesting internal services that are not meant to be reached by this user like system configurations, credentials, source codes, admin panels, and others depending on the privilege.

Types

- The basic one displays the response to the threat actor
- The blind SSRF: doesn’t display any response to the threat actor

How it arises?

When you try to request data from your normal website directly to an internal server or page the firewall would block you and won’t allow you to reach this data or pages but if you tried to reach this server with an internal one with it through a vulnerable website the firewall won’t block you as you’re your request is coming from an internal resource with this server

The problem is not only exposure of the website data, but the threat actor can also reach all the server data leading to all the websites hosted on that server!

- You can get LFI from this SSRF which allows you to read sensitive information from the server
- You can get RCE using this SSRF which allows you to exploit the server to control and take over it

Exploitation Methodology

It is depending on the website's functionality if it is requesting data from the server or any other URL “reading or sending”

- After reviewing the website and predicting the method of how it is working and found that a location is requesting a price of an item from an endpoint then!!
    - First, use your own tools assuming using the burp suite and opening the proxy to see the request
    - In the request, if you changed this endpoint and you can put any website to see if it would make any interaction or not
        - At this point, if you found that there is no response you may try to use the burp collaborator to see if there is a blind SSRF or not it would check if there were DNS or HTTP interaction

- Then you try to request an internal resource as this external interaction isn't considered a vulnerability, like requesting http://127.0.0.1 or http://localhost you can try to change the protocol
    - At this point, maybe the WAF would predict you or won't request that URL as it is blocked by the admin so what?
        - You would try to manipulate it by using other payloads like
            - http://127.0.01
            - http://127.001
            - http://127.01
            - http://127.1
        - You can try to use other Waf bypass like
            - Long IP http://2130706433
            - IP version 6 http://::1
            - Using domain localhost
            - Search in virus total to find subdomains maybe it is private or local, Then try to inject it in SSRF to find data

✓ You can try to scan the ports by adding ":80" ":443" ":22" etc.

- Finally, you can try to read files from the server using file:///etc/passwd

tip: - you have to try to check about SSRF if you found any location accepting or requiring a URL in the request

SSRF to RCE

Cloud providers use Metadata files to store sensitive information of the service user, such as SSH keys. This data is only accessible on the local network of the running instance. You can find it by searching about the cloud provider followed by metadata

For prevention, you can take a look at the OWASP cheat sheet

https://cheatsheetseries.owasp.org/cheatsheets/Server_Side_Request_Forgery_Prevention_Cheat_Sheet.html

for other vulnerabilities take a look at my notes:"

https://stemschools-my.sharepoint.com/:o:/g/personal/zyad_1318034_stemdakahlia_moe_edu_eg/Es7CMjdoc-5Gs45WJm-RJjYBZgNec1hVyFuWUzCYC27bDw