THE ZAP BLOG

# ZAP SSRF Setup

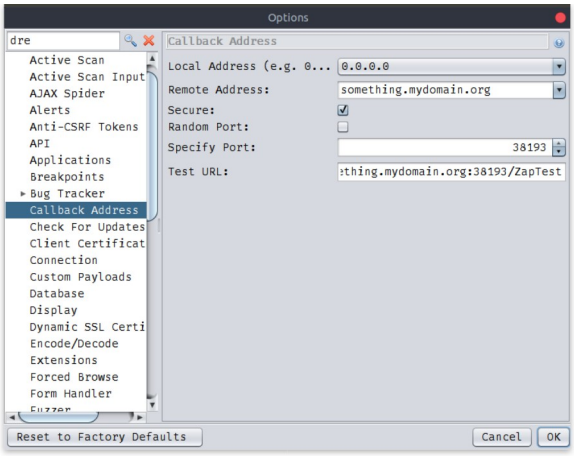
Some vulnerabilities can only be found by sending payloads that cause a callback to the tester. One example is XXE vulnerabilities when the XML rendering result is not available to the user. ZAP can find these vulnerabilities that depend on SSRF detection but the target system needs to be able to reach the ZAP callback endpoint. In many cases the computer running ZAP is behind some kind of NAT and doesn't have a public IP so it will not receive the expected callbacks and miss some of the existent vulnerabilities.

**Note**: OWASP is taking part in GSoC for 2020 and there is an open project idea to have an SSRF type external service created for use with ZAP, see: SSRF Detection/Handling for more info.

## Solution

### TLDR:

1. Have a domain like something.mydomain.org pointing to a server you own (further details below).
2. Set ZAP to always use the same callback port and define the Remote Address as your domain something.mydomain.org. (Via the Callback Options screen.)




3. Forward requests received on the server to the local ZAP with SSH port forwarding:

```
ssh -N -R 38193:localhost:38193 myUser@s
```

# My Setup:

For this setup you will need a DNS name and a server with a public address. If you don't have a DNS name and you don't want to spend money you can create one for free on https://duckdns.org (or numerous other free services). The options I use on ZAP are the ones on the TLDR image.

1. I have a domain like something.mydomain.org pointing to a Google Cloud server I own, this allows me to have a public IP.

2. I set ZAP to always use the same callback port, for instance 38193. This is important because if the port for ZAP callbacks changes you need to check it every time you start ZAP and change the system command for the port selected. (Via the Callback Options screen.)

3. I setup the Remote Address to my domain something.mydomain.org. This will make ZAP scan rules create SSRF request URLs with the domain something.mydomain.org and the static port we previously defined. For instance the XXE scan rule will send payloads with external entities leveraging a URL such as:

http://something.mydomain.org:38193/callback/XXESD215F.

4. If a vulnerability exists the server will make a request to http://something.mydomain.org:38193/callback/XXESD215F. This request will be made to my cloud server because it is the location my DNS is pointing to. To receive this request on ZAP you need to somehow redirect the request to your local machine. One way that doesn't require a public address on your machine is to use SSH port forwarding. What it does is to connect to your server through SSH, listen to a port on the server and send all the traffic received on that port to a port on your machine. The command I use is:

```
ssh -N -R 38193:localhost:38193 myUser@s
```

If you want to have a valid HTTPS certificate you can run some reverse proxy on your server (as Nginx) with a certificate, proxy the requests to other port, for instance 9999, and change the port forwarding to that address:

```
ssh -N -R 9999:localhost:38193 myUser@someth
```

# Troubleshoot:

- open the ports on the firewall
- make sure you allow port forwarding on the server ssh config `/etc/ssh/sshd_config`:

```
...
AllowTcpForwarding yes
GatewayPorts yes
...
```

# Conclusion

While other vulnerability scanners like Burp Suite have an external server (Collaborator) which facilitate reception of requests from the target server, ZAP still lacks that functionality. Luckily it can be obtained with this easy trick. This solution still lacks the ability to log DNS requests that can be used for out of band (OOB) tests, which may be necessary when the target app/system blocks outbound HTTP/HTTPS connections.

Prev: Dark Mode in the Weekly Release Next: Is ZAP the World's most Popular Web Scanner?