DNS Rebinding

> DNS rebinding changes the IP address of an attacker controlled machine name to the IP address of a target application, bypassing the same-origin policy and thus allowing the browser to make arbitrary requests to the target application and read their responses.

Summary

Tools

- Singularity of Origin - is a tool to perform DNS rebinding attacks.
- Singularity of Origin Web Client (manager interface, port scanner and autoattack)

Exploitation

First, we need to make sure that the targeted service is vulnerable to DNS rebinding. It can be done with a simple curl request:

```
curl --header 'Host: <arbitrary-hostname>' http://<vulnerable-service>:8080
```

If the server returns the expected result (e.g. the regular web page) then the service is vulnerable. If the server returns an error message (e.g. 404 or similar), the server has most likely protections implemented which prevent DNS rebinding attacks.

Then, if the service is vulnerable, we can abuse DNS rebinding by following these steps:

1. Register a domain.
2. Setup Singularity of Origin.
3. Edit the autoattack HTML page for your needs.
4. Browse to "http://rebinder.your.domain:8080/autoattack.html".
5. Wait for the attack to finish (it can take few seconds/minutes).

Protection Bypasses

> Most DNS protections are implemented in the form of blocking DNS responses containing unwanted IP addresses at the perimeter, when DNS responses enter the internal network. The most common form of protection is to block private IP addresses as defined in RFC 1918 (i.e. 10.0.0.0/8, 172.16.0.0/12, 192.168.0.0/16). Some tools allow to additionally block localhost (127.0.0.0/8), local (internal) networks, or 0.0.0.0/0 network ranges.

In the case where DNS protection are enabled (generally disabled by default), NCC Group has documented multiple DNS protection bypasses that can be used.

0.0.0.0

We can use the IP address 0.0.0.0 to access the localhost (127.0.0.1) to bypass filters blocking DNS responses containing 127.0.0.1 or 127.0.0.0/8.

CNAME

We can use DNS CNAME records to bypass a DNS protection solution that blocks all internal IP addresses. Since our response will only return a CNAME of an internal server, the rule filtering internal IP addresses will not be applied. Then, the local, internal DNS server will resolve the CNAME.

```
$ dig cname.example.com +noall +answer
; <<>> DiG 9.11.3-1ubuntu1.15-Ubuntu <<>> example.com +noall +answer
;; global options: +cmd
cname.example.com.            381     IN      CNAME   target.local.
```

localhost

We can use "localhost" as a DNS CNAME record to bypass filters blocking DNS responses containing 127.0.0.1.

```
$ dig www.example.com +noall +answer
; <<>> DiG 9.11.3-1ubuntu1.15-Ubuntu <<>> example.com +noall +answer
;; global options: +cmd
localhost.example.com.            381     IN      CNAME   localhost.
```

References

- How Do DNS Rebinding Attacks Work? - nccgroup, 2019