Investigation of DNS Rebinding Attacks

F20AN Advanced Network Security

BSc Computer Science, Year 4

Christian Gregg - H00224463 - cg23@hw.ac.uk

Ryan Shah - H00206511 - rs10@hw.ac.uk

Contents

1. Introduction

For the coursework assignment, we decided to study DNS rebinding attacks. A DNS rebinding attack is an exploit in which an attacker subverts the same-origin policy of browsers, by running a client-side script used to attack target machines on a network, and converts them into open network proxies[1]. This allows attackers to breach private networks, as well as use a victim machine for distributed denial-of-service (DDoS) attacks amongst other malicious activities.

This report describes our implementation of a DNS rebinding attack[2] which is demonstrated using two Virtual Machines (VMs) running Ubuntu 17.10, where one of the VMs plays the role of an attacker and the other, the role of a victim.

2. Same-Origin Policy (SOP)

The same-origin policy is a security mechanism of modern browsers, which controls the communication between scripts running in a browser. Scripts that are contained within a web page are permitted to access data in another web page, so long as they both have the same *origin*. The origin is defined as a combination of:

- URI scheme (http(s), ftp, file, etc.)
- Host name
- Port number (21, 80, 9091, etc.)

Without this policy, an attacker can obtain access to sensitive data. For example, assume that an individual is using a website that handles personal data such as a social networking site. Without SOP, and assuming the individual visits a malicious website in another browser tab, JavaScript on that website can do anything on the social network account that the person would be able to do, such as reading private messages and analysing the HTML DOM-tree after the person entered their password before submitting the form.
Taking this example, and applying it to a scenario involving online banking, it can be easily understood that the policy is essential.

3. Domain Name System (DNS)

DNS is a protocol within the TCP/IP protocol suite, which defines how devices exchange data on the Internet. It is described as a "hierarchical decentralized naming system for computers, services, or other resources connected to the Internet or a private network"[3].

A DNS associates a domain name with a corresponding IP address. The DNS identifies and locates devices and resources on the Internet or a private network, for instance when a URL is entered on a web browser the DNS will match it with an IP address for that location and setup a connection.

4. DNS Rebinding

To reiterate, DNS rebinding is a computer attack, for which an attacker subverts the same-origin policy of browsers, by running a client-side script used to attack target machines on a network, and converts them into open network proxies[1].

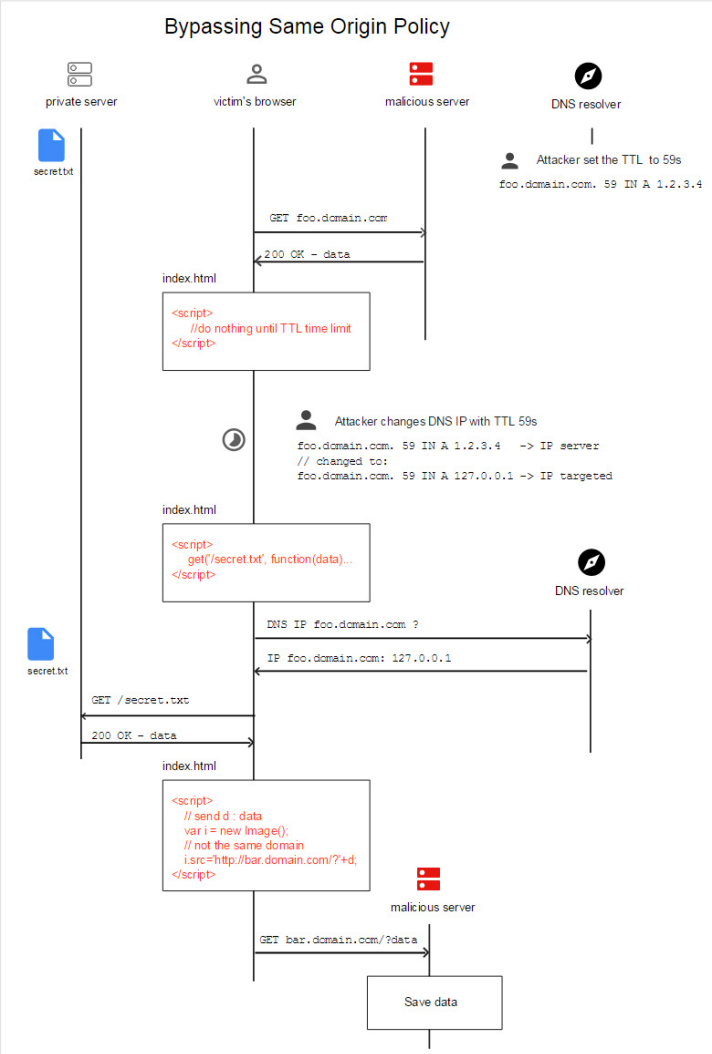

Figure 4.1: DNS Rebinding Flow Diagram[4]

An attacker would register a domain and assign it to a DNS server under their control. When a target/victim visits the attacker's domain, the DNS server initially responds with the IP address of a server which contains malicious client-side scripts (JavaScript or Flash). The code makes additional accesses to the original domain, which is permitted by the SOP. However when the target's browser runs the script, a new DNS request is made for the original domain and the attacker replies with a new IP address.

4.1 Types of DNS Rebinding

4.1.1 Multiple A Records

4.1.2 Time-Varying DNS

4.2 Motives

5. Implementing DNS Rebinding

6. DNS Rebinding Countermeasures

There have been attempts made to eradicate DNS rebinding, but only a few methods have been proven effective at mitigating the effects or stopping it entirely.

6.1 Extended Same-Origin Policy

With early iterations of the Same-Origin policy, many variations of DNS rebinding attacks would exploit the policy. To counter this, web browsers implemented countermeasures (as described in this section) to protect resources. The Same-Origin policy relies on information obtained from the DNS, which may not be under control of the owners of a web server. Therefore a light-weight extension to the policy, called the Extended Same-Origin policy, was made to also take into account information provided by a web server to avoid exploitation by DNS rebinding[5].

6.2 DNS Pinning

DNS Pinning is a technique web browsers implement, which locks an IP address to the value provided by the initial DNS response. It has however has been deprecated due to it unintentionally blocking a few legitimate uses of Dynamic DNS. An example is load balancing, which is a service provided by DNS. Load balancing is vital for major web servers and DNS pinning interferes with this. As well as this, DNS pinning does not protect against **sophisticated** DNS rebinding attacks.

6.3 DNS Filtering

Another technique used to prevent DNS rebinding attacks includes filtering out private IP addresses from DNS responses, which can be done in a number of ways.

6.3.1 Filtering through external DNS servers

OpenDNS is a service that extends the DNS by adding several security features, in addition to regular DNS services. One of these features includes optional content filtering and the ability to filter out IP addresses from DNS responses. It supports a protocol which authenticates traffic that moves between a host computer and the OpenDNS nameservers, called the DNSCrypt protocol.

6.3.2 Local nameserver configuration

System admins can configure their organisation's local nameservers such that external names cannot be resolved/mapped to the organisation's internal IP addresses. This is a useful technique to prevent against DNS rebinding, however a potential attacker could map the internal IP address ranges that the organisation is currently using.

6.3.3 DNS filtering in a firewall

dnswall?

6.4 Router Configuration

DNS Rebinding attacks have been stopped more successfully through more secure router configurations. Services, such as the HTTP server on a router, are bound to all network interfaces and can be accessible by all IP addresses in a range that it has. The routers drop anything that comes in through the external port, which doesn't matter because it accesses external IP addresses from the internal Local Area Network (LAN).

6.5 Firefox NoScript Extension

??

7. Conclusion

References

[1] Collin Jackson, Adam Barth, Andrew Bortz, Weidong Shao, and Dan Boneh. Protecting browsers from dns rebinding attacks. *ACM Transactions on the Web (TWEB)*, 3(1):2, 2009.

[2] Cve-2018-5702. `https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2018-5702`. Accessed: 27/02/2018.

[3] Domain name system. `https://en.wikipedia.org/wiki/Domain_Name_System`. Accessed: 27/02/2018.

[4] Bypass same origin policy - by-sop (github). `https://github.com/mpgn/ByP-SOP`. Accessed: 27/02/2018.

[5] Martin Johns, Sebastian Lekies, and Ben Stock. Eradicating dns rebinding with the extended same-origin policy. In *USENIX security symposium*, pages 621–636, 2013.