DNS Rebinding

Based on “There’s No Place Like 127.0.0.1 -- Achieving Reliable DNS Rebinding in Modern Browsers” by Luke Young @ DEFCON 25

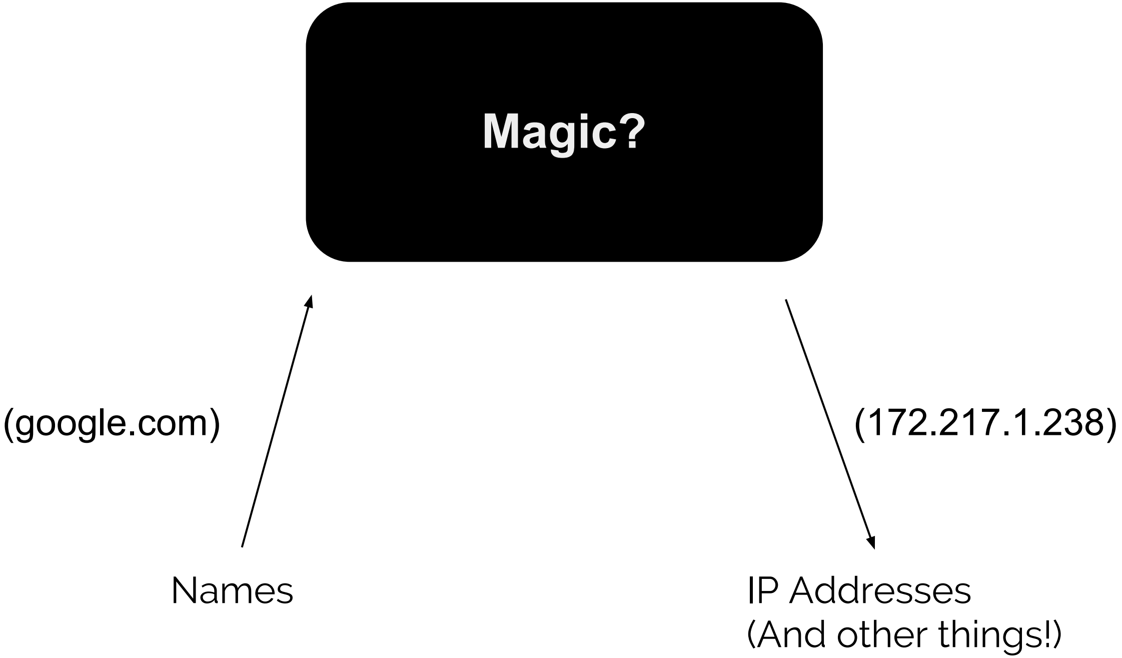
Outline

- Background Information
    - The Domain Name System
    - Cross Site Request Forgery
    - The Same Origin Policy
- The DNS Rebinding Attack
- Limitations
- Applications
    - Exfiltrating from Intranets
    - Attacking Routers
- Advanced Techniques
- Countermeasures

The Domain Name System

- Computers on the internet are addressed by numerical values (IP Addresses).
- For instance, http://2899902958/ will take you to google.
- Remembering numbers is not convenient.
- In the beginning, SRI maintained hosts.txt, a centralized list of hostnames.
- This quickly became unmanageable.
- In 1987, the Domain Name System was described by RFCs 1034 and 1035.
- The Domain Name System is a hierarchical, distributed system for resolving names to addresses on the internet.

The Domain Name System



Magic?

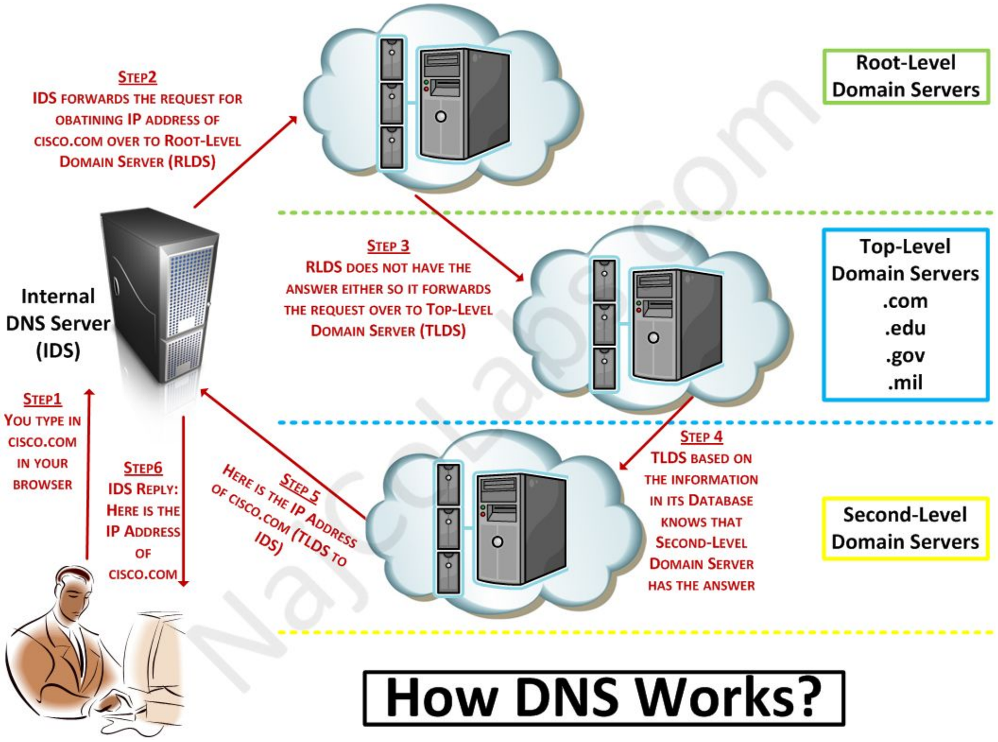
(google.com)

Names

(172.217.1.238)

IP Addresses
(And other things!)

The Domain Name System


Root-Level Domain Servers
Top-Level Domain Servers
.com
.edu
.gov
.mil
Second-Level Domain Servers
Internal DNS Server (IDS)
Step1
You type in cisco.com in your browser
Step2
IDS forwards the request for obatining IP address of cisco.com over to Root-Level Domain Server (RLDS)
Step 3
RLDS does not have the answer either so it forwards the request over to Top-Level Domain Server (TLDS)
Step 4
TLDS based on the information in its database knows that Second-Level Domain Server has the answer
Step 5
Here is the IP Address of cisco.com (TLDS to IDS)
Step6
IDS Reply: Here is the IP Address of cisco.com
How DNS Works?
NajcoLabs.com

(http://najcolabs.com/?p=257)

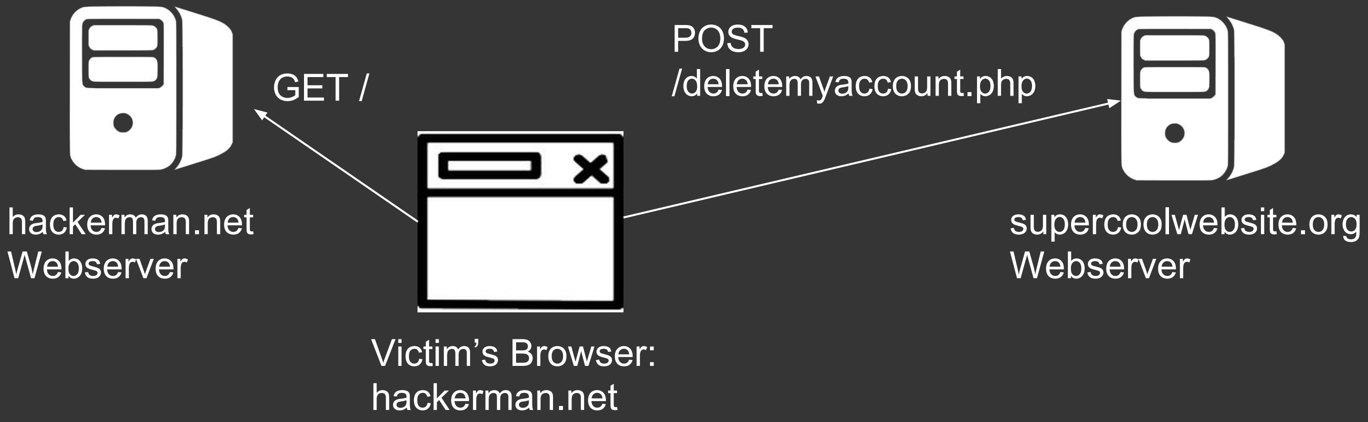
The Domain Name System

The dnsutils “dig” program can be used to inspect the process.

Important: You can set up your own DNS server for your own domains.

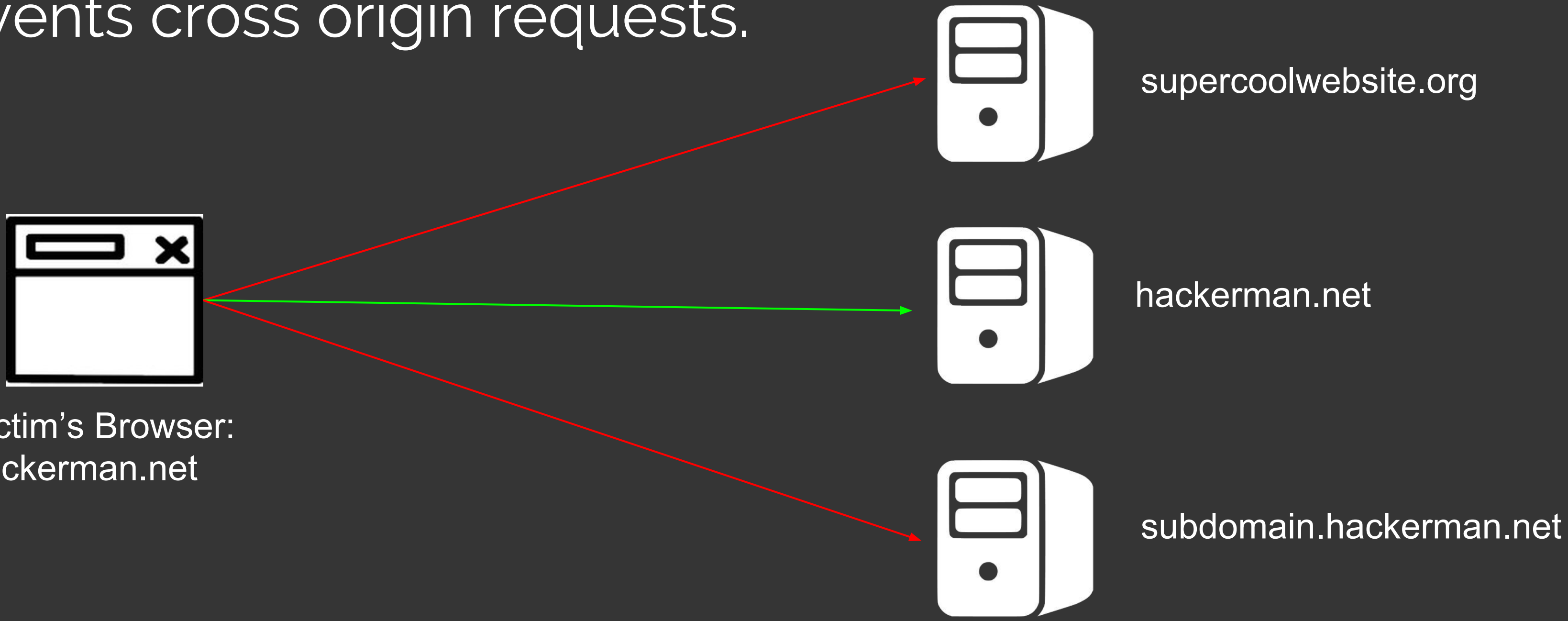
Cross Site Request Forgery

The attacking website sends a request to the target website.



The Same Origin Policy

Prevents cross origin requests.



DNS Rebinding

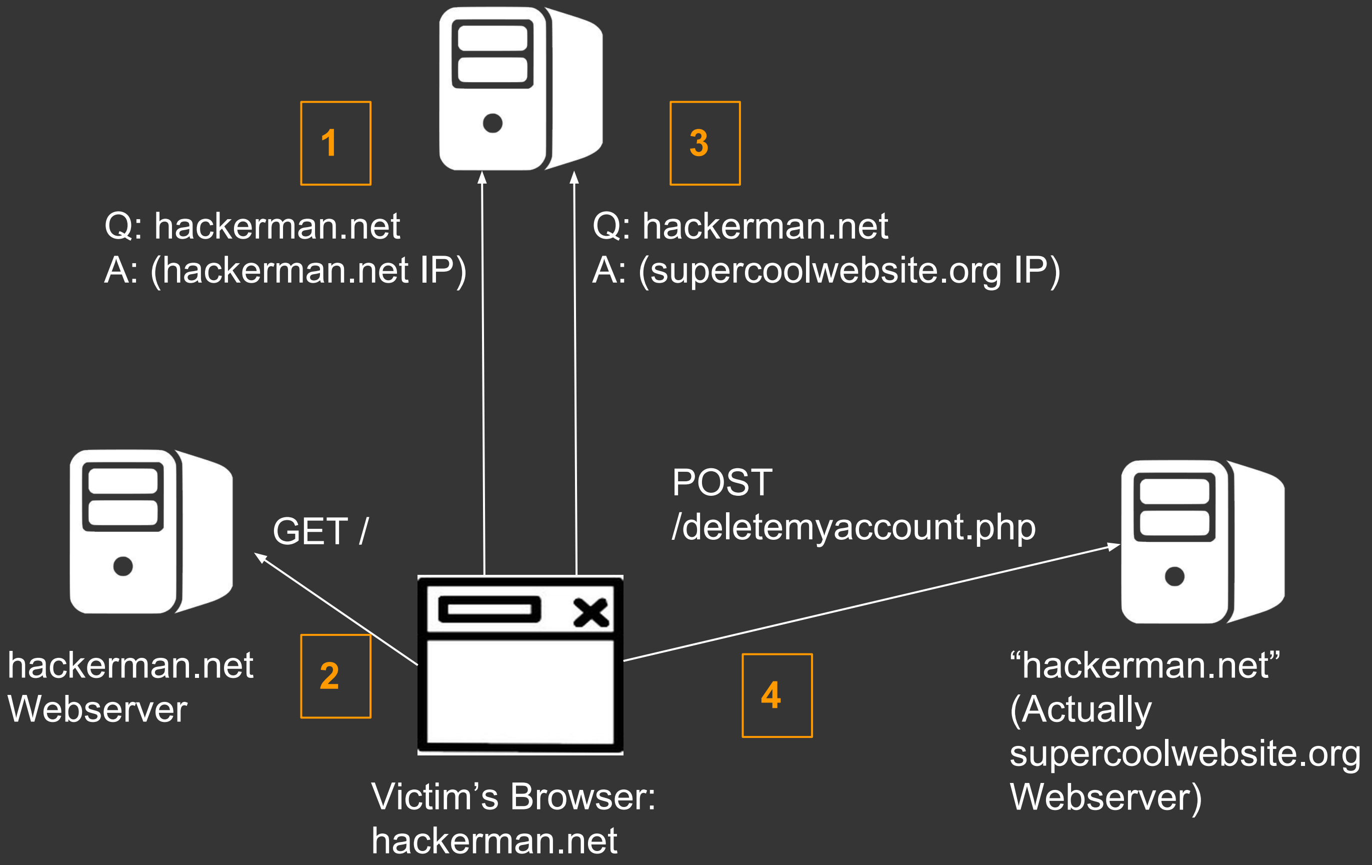
What if hackerman.net pointed to both our webserver and the target website?

DNS rebinding involves the use of a malicious DNS server.

DNS Rebinding

The Basic Technique

1. When asked about hackerman.net, our DNS server first responds with a record pointing to our website. The record is set to expire in a very short amount of time.
2. The next time we are asked about hackerman.net, respond with the address of the victim’s site.



1

Q: hackerman.net
A: (hackerman.net IP)

3

Q: hackerman.net
A: (supercoolwebsite.org IP)

GET /

2

hackerman.net
Webserver

Victim’s Browser:
hackerman.net

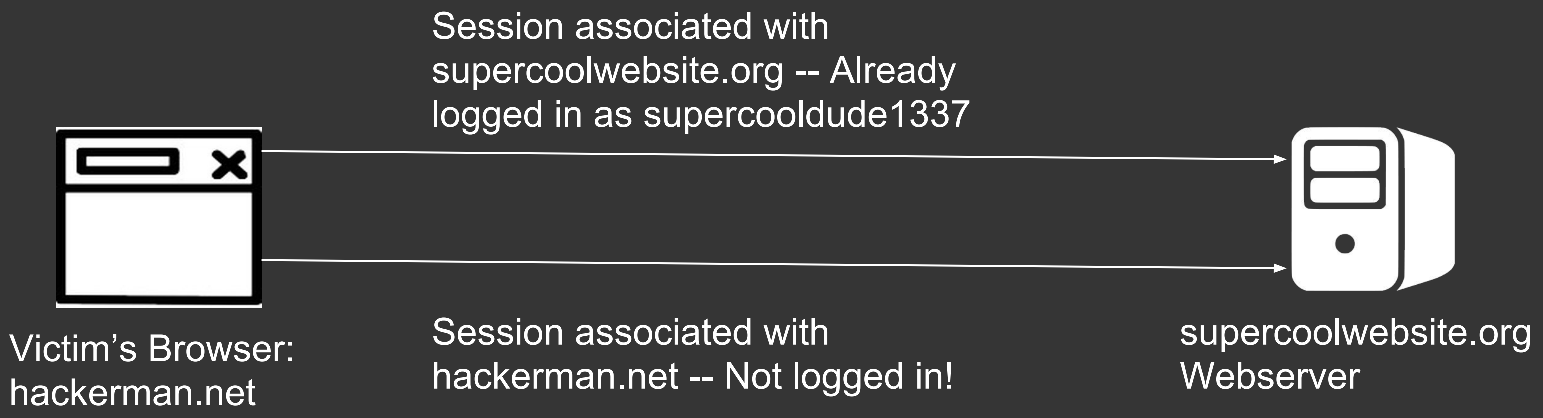
POST
/deletemyaccount.php

4

“hackerman.net”
(Actually
supercoolwebsite.org
Webserver)

Limitations

- This technique does not work for websites that require authentication.
- Most sites use a session cookie to tell if a user is logged in.
- Browsers associate cookies with individual domains.
- If we use a different domain, we get a different session.



Session associated with supercoolwebsite.org -- Already logged in as supercooldude1337

Victim’s Browser: hackerman.net

Session associated with hackerman.net -- Not logged in!

supercoolwebsite.org Webserver

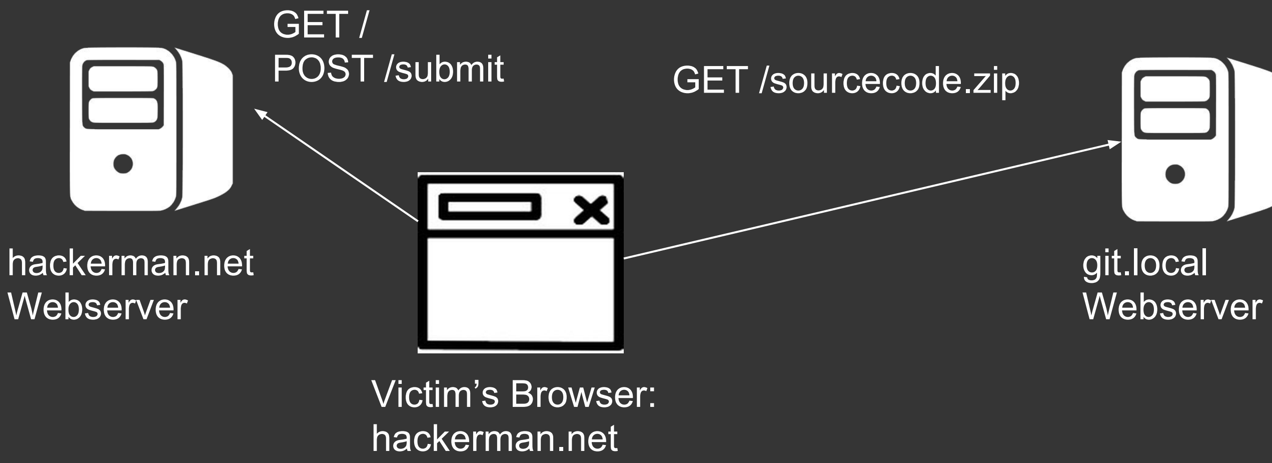
So what good is this attack?

Applications: Intranet Sites

This was the use described by Mr. Young at DEFCON.

- Corporate networks often host sites on their internal network.
  - Private wikis, version control (git, svn), accounting software, customer records, etc.
- These services are often set up not to use authentication.
  - They're on our intranet so they're safe, right?

Applications: Intranet Sites



GET /
POST /submit

GET /sourcecode.zip

hackerman.net
Webserver

git.local
Webserver

Victim’s Browser:
hackerman.net

Applications: Intranet Sites

This attack has a number of requirements:

- You need a victim in the company.
- You need to know the layout of the target’s network or the internal domain name used by the target service.
- The service needs to be secured poorly enough for this to work.

Applications: Home Routers

Most home routers can be configured using HTTP.

- They require authentication, but default credentials are often used.
- The address of the router is easily predictable.
- Can we upload firmware?
- If not, we can still do a lot with a misconfigured router.
- Less targeted.
- This attack is fairly old, so this is probably not an issue in newer routers.

Advanced Techniques

Mr. Young released a tool that automates DNS rebinding attacks. It handles the DNS server, HTTP server, ad client-side javascript portions of the attack.

https://github.com/linkedin/jaqen

Advanced Techniques

#1: Threshold Rebind

Answers some number of requests by pointing to the attacker's server, then answers subsequent requests with the address of the victim's server.

Used if the client makes multiple DNS requests.

Advanced Techniques

#2: Multirecord Rebind

Serves multiple options for the client to choose from.

Once the malicious site is served, the attacking server refuses requests, forcing the victim server to be used.

Countermeasures

- Caches often make these attacks hard to pull off.
- Browsers can pin a domain name to a specific IP.
- Firewalls can be used to filter DNS traffic.
- Rebinding can be blocked when a domain changes from public to local.
- The best way to deal with rebinding this is to check the **host** header.

Questions?