IPv4 Overview

CS460 - Cyber Security Lab
Spring 2009

Outline

- Review Layered Network Architecture
- Network Layer protocols
- Transport Layer Protocols
- Application Layer Protocols

Reading Material

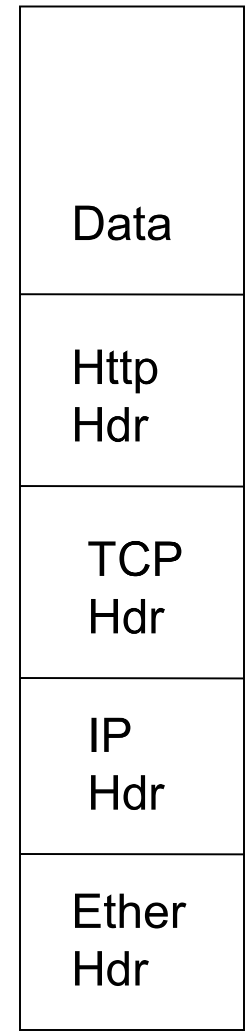
- Many texts on IP networking
  - Computer Networks, Andrew Tannenbaum
  - Data and Computer Communications, William Stallings
  - Internetworking with TCP/IP Vol 1, Douglas Comer
- Plus all the originals from the Internet Engineering Task Force (IETF)
  - http://ietf.org/

OSI Reference Model

- The layers
  - 7: Application, e.g., HTTP, SMTP, FTP
  - 6: Presentation
  - 5: Session
  - 4: Transport, e.g. TCP, UDP
  - 3: Network, e.g. IP, IPX
  - 2: Data link, e.g., Ethernet frames, ATM cells, 802.11
  - 1: Physical, e.g., Ethernet media, ATM media, radio waves
- Standard software engineering reasons for thinking about a layered design

Various network devices

- Hosts and servers – Operate at Level 7 (application)
- Proxies – Operate at level 7
- Firewalls – Operate between levels 2 and 7. From the outside world make changes at levels 2 (in transparent mode) or 3 (in routing mode)
- Routers – Operate at Level 3 (network)
- Switches or Hubs – Operate at level 2 (data link)
- Gateways – Operate at level 2



IPv4

- See Wikipedia for field details
  - http://en.wikipedia.org/wiki/IPv4

<table>
<tr><td>Version</td><td>IHL</td><td>Type of service</td><td colspan="4">Total length</td></tr>
<tr><td colspan="3">Identification</td><td></td><td>DF</td><td>MF</td><td>Frag Offset</td></tr>
<tr><td colspan="2">Time to live</td><td>Protocol</td><td colspan="4">Header checksum</td></tr>
<tr><td colspan="7">Source address</td></tr>
<tr><td colspan="7">Destination Address</td></tr>
<tr><td colspan="7">0 or more words of options</td></tr>
</table>

Ipv4 Addressing

- Each entity has at least one address
- Addresses divided into networks
- Addresses in your networks are “directly” connected
  - Broadcasts should reach them
  - No need to route packets to them

IP Network Specification

- Classful routing (up until around '93)
  - Class A (8 bit prefix)
    - 0.0.0.0 - 127.255.255.255
  - Class B (16 bit prefix)
    - 128.0.0.0 - 191.255.255.255
  - Class C (24 bit prefix) networks
    - 192.0.0.0 - 223.255.255.255
  - Specific prefix hardcoded to be one of these classes
- Classless Inter-Domain Routing (CIDR)
  - Specify prefix and and prefix size
  - 192.168.1.0/24 = 192.168.1.0 255.255.255.0 =192.168.1.0 - 192.168.1.255

Switches

- Original Ethernet broadcast all packets
- Layer two means of passing packets
  - Learn or config which MAC's live behind which ports
  - Only pass traffic to the appropriate port
- Span ports
  - Mirror all traffic

Address spoofing

- Sender can put any source address in packets he sends:
  - Can be used to send unwelcome return traffic to the spoofed address
  - Can be used to bypass filters to get unwelcome traffic to the destination
- Reverse Path verification can be used by routers to broadly catch some spoofers

Fragmentation

- May need to fragment an IP packet if one data link along the way cannot handle the packet size
  - Perhaps path is a mix of different HW
  - Perhaps unexpected encapsulation makes the packet larger than the source expected
  - Hosts try to understand Maximum Transmission Unit (MTU) to avoid the need for fragmentation (which causes a performance hit)
- Any device along the way can fragment
  - Identification field identifies all elements of the same fragment
  - Fragmentation stored in the MF (more fragments) and fragment offset fields
  - Devices can reassemble too
  - But generally the destination does the reassembly

Fragmentation Flaws?

- Split packet to fool simple firewall and IDS
  - Intermediate content observers must do reassembly
- Overlapping fragments
  - Can be used to trick IDS by hiding, e.g. a “get /etc/password” request
  - Different clients reassemble overlapping fragments differently
  - Just drop overlapping fragments
- Bad fragment offsets exploit poor stack implementations
  - E.g. Teardrop attack, negative offsets or overlarge offsets cause buffer overflows
  - Firewalls can check for well formed packets.
- Resource attacks on re-assemblers
  - Send all but one fragment for many packets

Address Resolution Protocol (ARP)

- Used to discover mapping of neighboring ethernet MAC to IP addresses.
  - Need to find MAC for 192.168.1.3 which is in your interface's subnetwork
  - Broadcast an ARP request on the link
  - Receive an ARP reply giving the correct MAC
  - The device stores this information in an ARP cache or ARP table

Does Anyone Remember ARP Cache Poisoning?

ARP cache poisoning

- Bootstrap problem with respect to security. Anyone can send an ARP reply
  - The Ingredients to ARP Poison, http://www.governmentsecurity.org/articles/TheIngredientstoARP
- Classic Man-in-the-middle attack
  - Send arp reply messages to device so they think your machine is someone else
  - Better than simple sniffing because not just best effort.
- Solutions
  - Encrypt all traffic
  - Monitoring programs like arpwatch to detect mapping changes
    - Which might be valid due to DHCP

Basic IPv4 Routing

- Static routing. Used by hosts, firewalls and routers.
  - Routing table consists of entries of
    - Network, Next hop address, metric, interface
  - May have routing table per incoming interface
  - To route a packet, take the destination address and find the best match network in the table. In case of a tie look at the metric
    - Use the corresponding next hop address and interface to send the packet on.
    - The next hop address is on the same link as this device, so you use the next hop's data-link address, e.g. ethernet MAC address
  - Decrement "time to live" field in IP header at each hop. Drop packet when it reaches 0
    - Attempt to avoid routing loops
    - As internet got bigger, TTL fields got set bigger. 225 maximum

Routing example

- Receive a packet destined to 192.168.3.56 on inside interface
- Local routing table for inside interface

1. 192.168.2.0/30, 127.0.0.1, 1, outside
2. 192.168.5.0/29, 127.0.0.1, 1, dmz
3. 192.168.3.0/24, 192.168.5.6, 1, dmz
4. 192.168.3.0/24, 192.168.1.2, 3, outside
5. 0.0.0.0/0, 192.168.1.2, 1, outside

- Entries 3 and 4 tie. But metric for 3 is better
- Entries 1 and 2 are for directly connected networks

Source Based Routing

- In the IP Options field, can specify a source route
  – Was conceived of as a way to ensure some traffic could be delivered even if the routing table was completely screwed up.
- Why is this bad?
  – Can be used by the bad guy to avoid security enforcing devices
  – Most folks configure routers to drop packets with source routes set

IP Options in General

- Originally envisioned as a means to add more features to IP later
- Most routers drop packets with IP options set
  - Stance of not passing traffic you don't understand
  - Therefore, IP Option mechanisms never really took off
- In addition to source routing, there are security Options
  - Used for DNSIX, a MLS network encryption scheme

Dynamic Routing Protocols

- For scaling, discover topology and routing rather than statically constructing routing tables
  - Open Shortest Path First (OSPF): Used for routing within an administrative domain (Autonomous System)
  - RIP: not used much anymore
  - Border Gateway Protocol (BGP): Used for routing between administrative domains. Can encode non-technical transit constraints, e.g. Domain X will only carry traffic of paying customers
    - Receives full paths from neighbors, so it avoids counts to infinity.

Dynamic Routing

- Injecting unexpected routes a security concern.
  - BGP supports TCP MD5 authentication
    - Creates a hash of the TCP header and data portion
    - Keyed with shared secret
  - Filter out route traffic from unexpected (external) points
  - OSPF has MD5 authentication, and can statically configure neighbour routers, rather than discover them.

Secure BGP

- Renewed government emphasis
- BBN prototype done earlier this decade
- Like Secure DNS add PKI
  - Bind certificates with ownership of address blocks and Autonomous systems
- BBN Site
  - http://www.ir.bbn.com/sbgp/
  - Secure Border Gateway Protocol (S-BGP) Kent, S.; Lynn, C.; Seo, K. Selected Areas in Communications, IEEE Journal on Volume 18, Issue 4, Apr 2000

Internet Control Message Protocol (ICMP)

- Used for diagnostics
  - Destination unreachable
  - Time exceeded, TTL hit 0
  - Parameter problem, bad header field
  - Source quench, throttling mechanism rarely used
  - Redirect, feedback on potential bad route
  - Echo Request and Echo reply, ping
  - Timestamp request and Timestamp reply, performance ping
- Can use information to help map out a network
  - Some people block ICMP from outside domain

Smurf Attack

- An amplification DoS attack
  - A relatively small amount of information sent is expanded to a large amount of data
- Send ICMP echo request to IP broadcast addresses. Spoof the victim's address as the source
- The echo request receivers dutifully send echo replies to the victim overwhelming it
- Fraggle is a UDP variant of the same attack

Transport layer

- UDP and TCP
- Transport flows are defined by source and destination ports
  - A pair of devices can have numerous flows operating simultaneously by communicating between different pairs of ports
- Applications are associated with ports (generally just destination ports)
  - IANA organizes port assignments http://www.iana.org/
- Source ports generally dynamically selected
  - Ports under 1024 are considered well-known ports
  - Would not expect source ports to come from the well-known range
- Scanners probe for listening ports to understand the services running on various machines

Datagram Transport

- User Datagram Protocol (UDP)
  - A best-effort delivery, no guarantee, no ACK
  - Lower overhead than TCP
  - Good for best-effort traffic like periodic updates
  - No long lived connection overhead on the endpoints
- Some folks implement their own reliable protocol over UDP to get “better performance” or “less overhead” than TCP
  - Such efforts don’t generally pan out
- TFTP and DNS protocols use UDP
- Data channels of some multimedia protocols, e.g., H.323 also use UDP

UDP Header

Source Port	Destination Port
UDP Length	UDP checksum

Reliable Streams

- Transmission Control Protocol (TCP)
  - Guarantees reliable, ordered stream of traffic
  - Such guarantees impose overhead
  - A fair amount of state is required on both ends
- Most Internet protocols use TCP, e.g., HTTP, FTP, SSH, H.323 control channels

TCP Header

<table>
<tr><td colspan="8">Source Port</td><td>Destination Port</td></tr>
<tr><td colspan="9">Sequence Number</td></tr>
<tr><td colspan="9">Acknowledgement number</td></tr>
<tr><td>HDR
Len</td><td></td><td>U
R
G</td><td>A
C
K</td><td>P
S
H</td><td>R
S
T</td><td>S
Y
N</td><td>F
I
N</td><td>Window
Size</td></tr>
<tr><td colspan="8">Checksum</td><td>Urgent Pointer</td></tr>
<tr><td colspan="9">Options (0 or more words)</td></tr>
</table>

Three-way Handshake

Syn flood

- A resource DoS attack focused on the TCP three-way handshake
- Say A wants to set up a TCP connection to B
  1. A sends SYN with its sequence number X
  2. B replies with its own SYN and sequence number Y and an ACK of A's sequence number X
  3. A sends data with its sequence number X and ACK's B's sequence number Y

– Send many of the first message to B. Never respond to the second message.
  – This leaves B with a bunch of half open (or embryonic) connections that are filling up memory
  – Firewalls adapted by setting limits on the number of such half open connections.

Syn Flood protections

- Adjust limits on half open connections
- Syn proxying
- Syncookies
  - Add structure to the ack number
    - Top 4 bits: t mod 32, where t is a running counter
    - Next 3 bits: encoding of MSS
    - Bottom 24 bits: Server selected secret function of client IP address and port, server IP address and port, and t
  - http://cr.yp.to/syncookies.html

Application Protocols

- Single connection protocols
  - Use a single connection, e.g. HTTP, SMTP
  - Expand on some of the SMTP commands...
- Dynamic Multi-connection Protocols, e.g. FTP and H.323
  - Have a well known control channel
  - Negotiate ports and/or addresses on the control channel for subsidiary data channels
  - Dynamically open the negotiated data channels
- Protocol suites, e.g. Netbios and DNS

Spoofing Applications

- Often times ridiculously easy
- Fake Client
  - Telnet to an SMTP server and enter mail from whoever you want
  - Authenticating email servers
    - Require a password
    - Require a mail download before server takes send requests
- Fake server
  - Phishing: misdirect user to bogus server

Example

- > telnet target.com 25
- *HELO target.com*
- *MAIL FROM:<obama@whitehouse.gov>*
- *RCPT TO:<target@target.com>*
- *DATA*
- *Just kidding about that stimulus package.*
- *.*
- *QUIT*
-
- See RFC 821 for SMTP syntax

DHCP

- Built on older BOOTP protocol (which was built on even older RARP protocol)
  - Used by diskless Suns
- Enables dynamic allocation of IP address and related information
- Runs over UDP
- No security considered in the design. What are the problems?
  - Bogus DHCP servers handing out addresses of attackers choice
  - Hand out DNS and default gateways of attacker's choice
  - Bogus clients grabbing addresses
- IETF attempted to add DHCP authentication but rather late in the game to do this.
- Other solutions?
  - Physically secure networks
  - Use IPSec

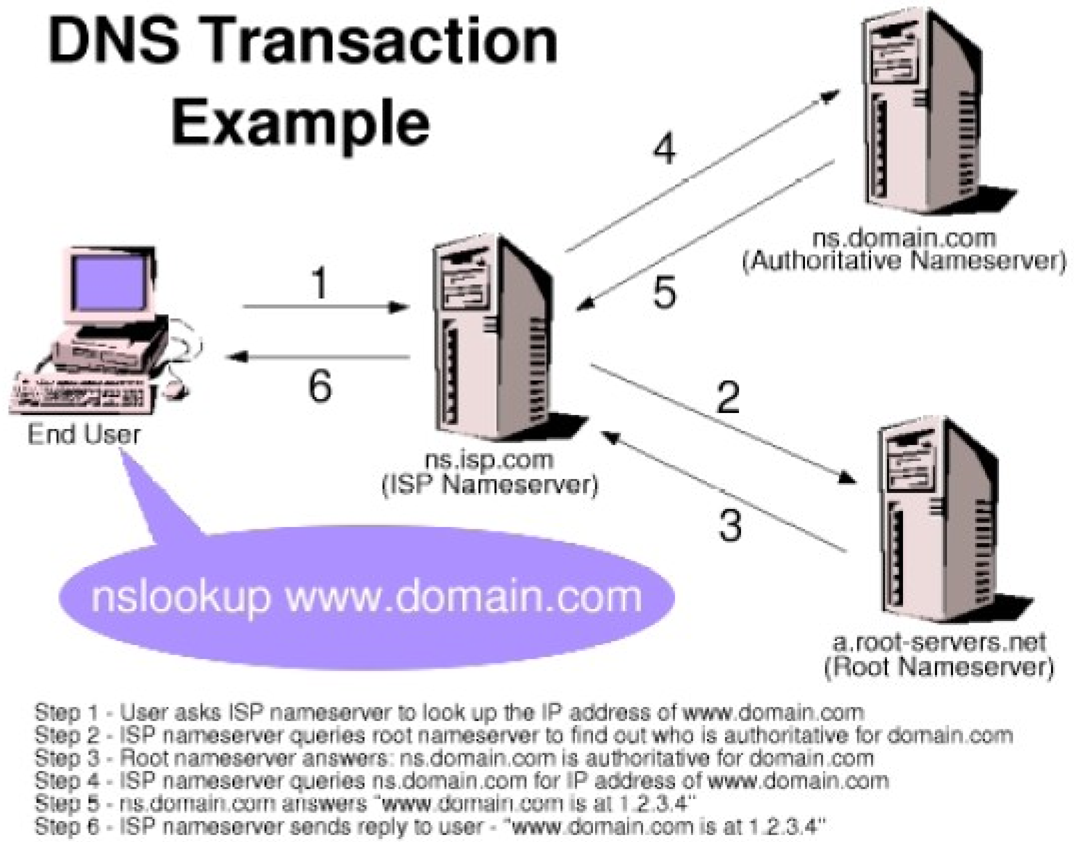
Domain Name System (DNS)

- Hierarchical service to resolve domain names to IP addresses.
  - The name space is divided into non-overlapping zones
  - E.g., consider shinrich.cs.uiuc.edu.
  - DNS servers in the chain. One for .edu, one for .uiuc.edu, and one for .cs.uiuc.edu
- Can have primary and secondary DNS servers per zone. Use TCP based zone transfer to keep up to date
- Like DHCP, no security designed in
  - But at least the DNS server is not automatically discovered
  - Although this information can be dynamically set via DHCP

DNS Problems

- DNS Open relays
  - Makes it look like good DNS server is authoritative server to bogus name
  - Enables amplification DoS attack
  - http://www.us-cert.gov/reading_room/DNS-recu
- DNS Cache Poisoning
  - Change the name to address mapping to something more desirable to the attacker

DNS Transaction

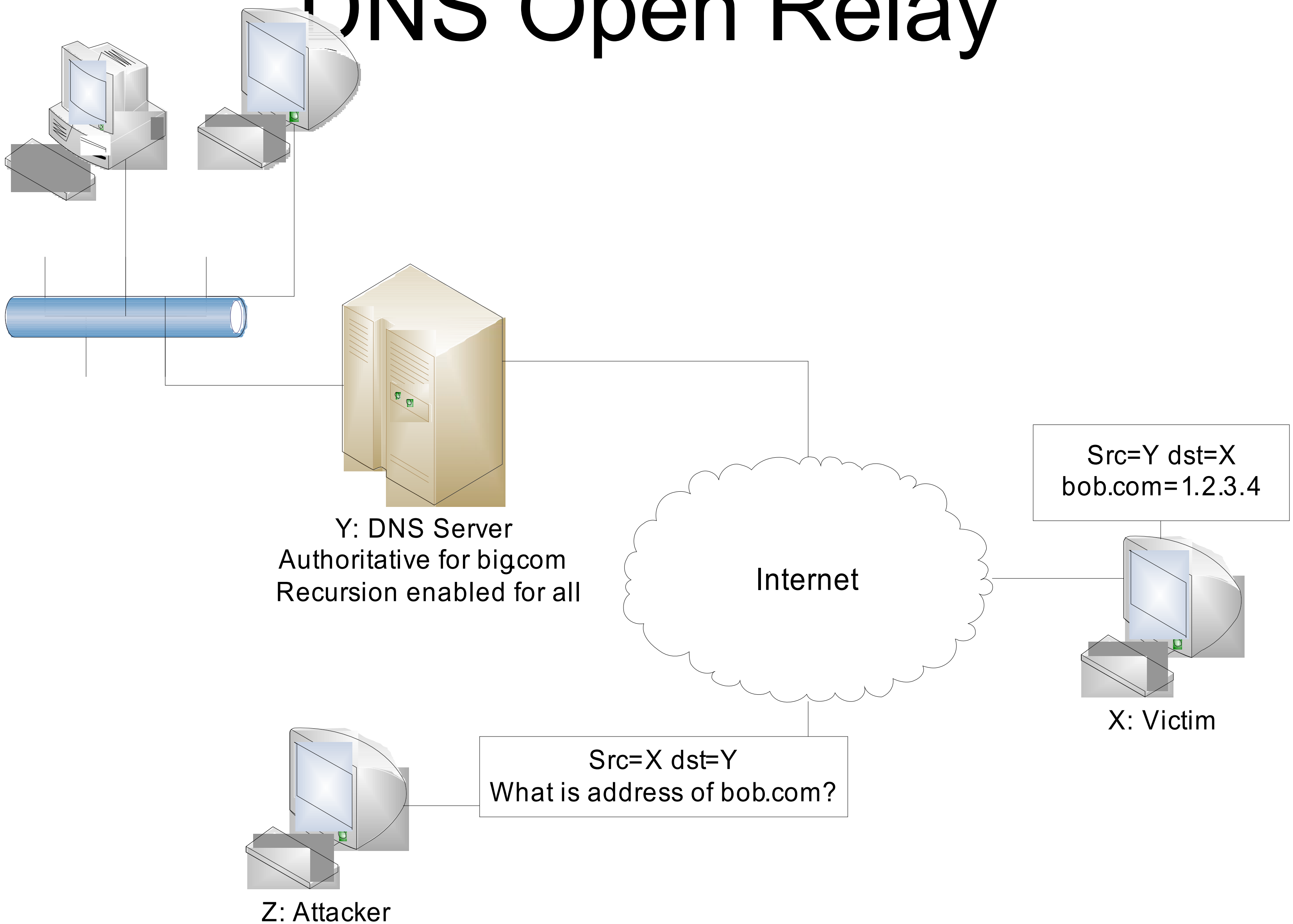


DNS Pictures thanks to http://www.lurhq.com/dnscache.pdf

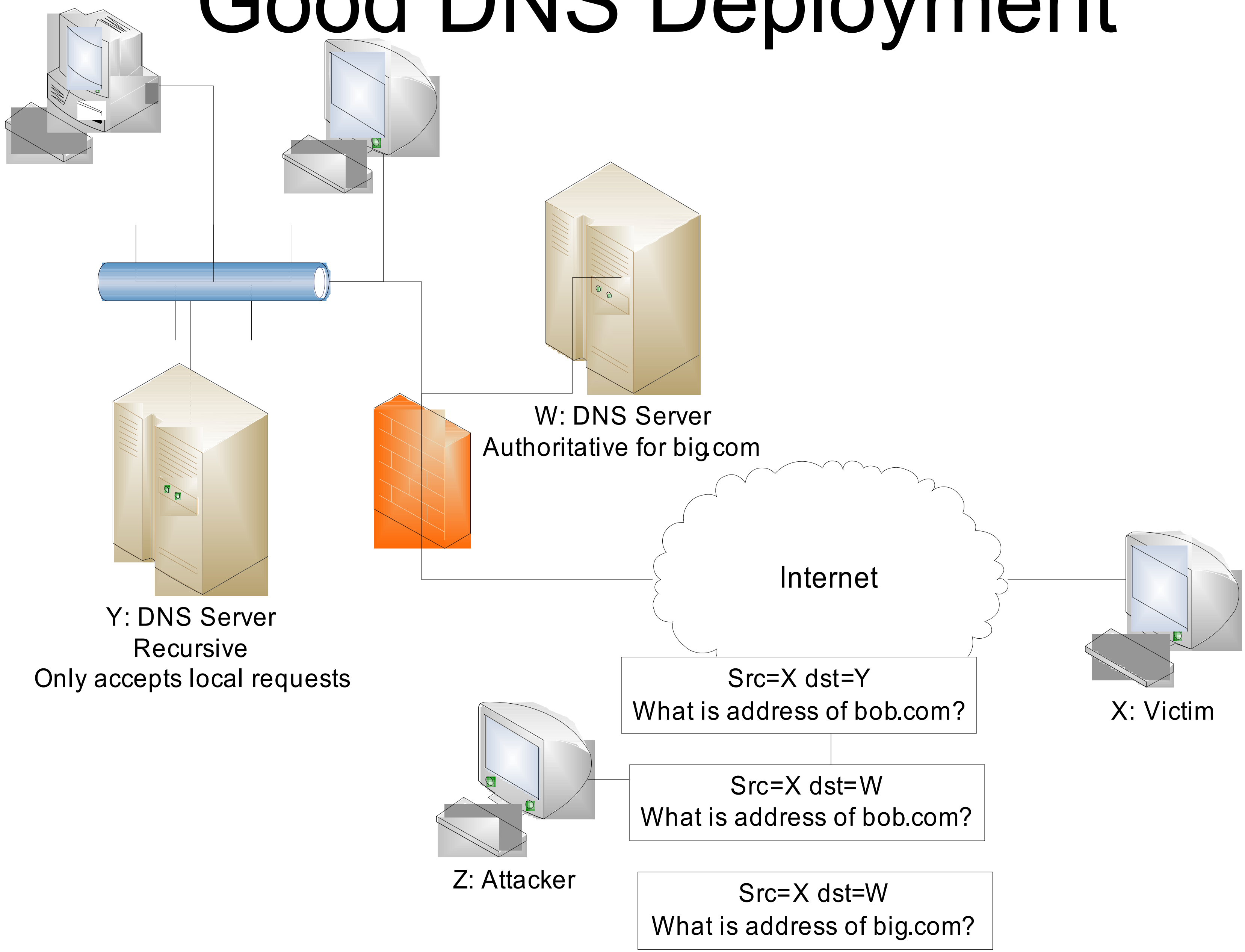
DNS Communication

- Use UDP
- Requests and responses have matching 16 bit transaction Ids
- Servers can be configured as
  - Authoritative Nameserver
    - Officially responsible for answering requests for a domain
  - Recursive
    - Pass on requests to other authoritative servers
  - Both (this can be the problem)

DNS Open Relay



Good DNS Deployment



DNS Cache Poisoning

- Older implementations would just accept additional information in a reply
  - e.g. A false authoritative name server
  - Fixed by bailiwick checking. Additional records only include entries from the requested domain
- Now to spoof a reply must anticipate the correct transaction ID
  - Only 16 bits
  - Random selection of ID isn't always the greatest

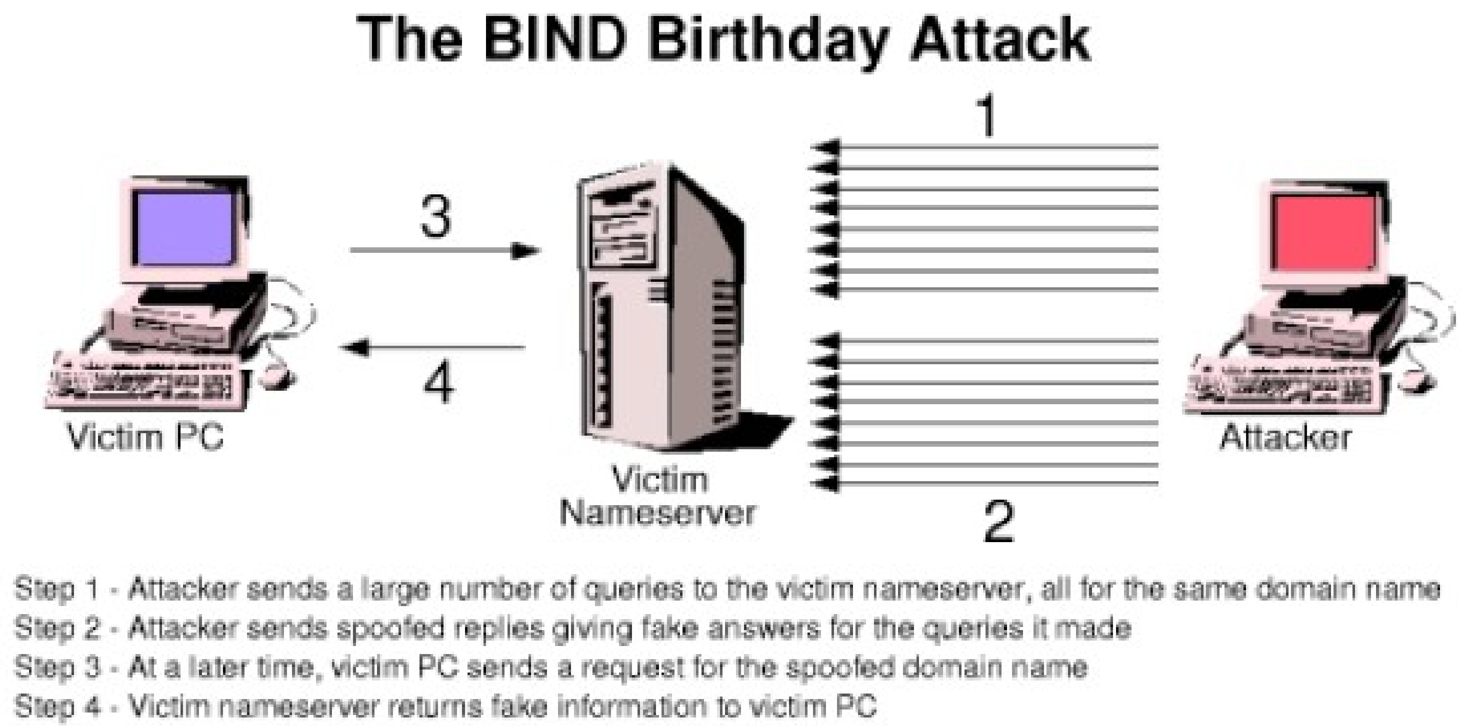
Bailiwick Checks

```
$ dig @ns1.example.com www.example.com
;; ANSWER SECTION:
www.example.com.    120     IN    A    192.168.1.10

;; AUTHORITY SECTION:
example.com.        86400    IN    NS
ns1.example.com.
example.com.        86400    IN    NS
ns2.example.com.

;; ADDITIONAL SECTION:
ns1.example.com.    604800   IN    A    192.168.2.20
ns2.example.com.    604800   IN    A    192.168.3.30
www.linuxjournal.com. 43200  IN    A    66.240.243.113
```

Tricking the Transaction ID's



Kaminsky's Observations

- Most implementations don't randomize source ports (making the TID collision more likely)
- Try to poison through the additional information (side stepping the bailiwick check)

```
$ dig doesnotexist.example.com
;; ANSWER SECTION:
doesnotexist.example.com.  120   IN  A    10.10.10.10

;; AUTHORITY SECTION:
example.com.            86400   IN  NS   www.example.com.

;; ADDITIONAL SECTION:
www.example.com.       604800   IN  A    10.10.10.20
```

DNSSEC

- Seeks to solve the trust issues of DNS
- Uses a key hierarchy for verification
- Has been under development for a decade and still not really deployed
- Provides authentication, not confidentiality
- DNS Threat Analysis in RFC 3833.

Summary

- IPv4 not designed with security in mind
- Complexity can be exploited
  - Poor implementations
  - Edge cases in standards
- Bootstrapping can be exploited
  - Easy of configuration vs strong trust