

# Network Protocols and Vulnerabilities

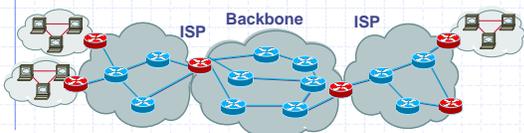
John Mitchell

## Outline

- ◆ Basic Networking
- ◆ Network attacks
  - Attacking host-to-host datagram protocols
    - SYN flooding, TCP Spoofing, ...
  - Attacking network infrastructure
    - Routing
    - Domain Name System

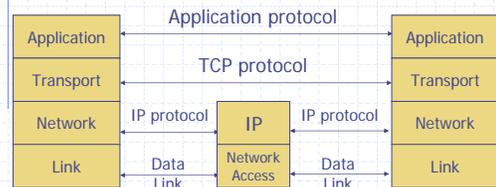
This lecture is about the way things work now and how they are not perfect. Next lecture – some security improvements (still not perfect)

## Internet Infrastructure

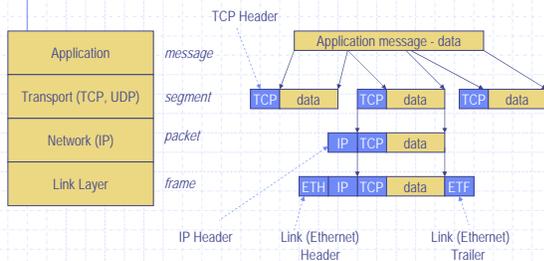


- ◆ Local and interdomain routing
  - TCP/IP for routing, connections
  - BGP for routing announcements
- ◆ Domain Name System
  - Find IP address from symbolic name ([www.cs.stanford.edu](http://www.cs.stanford.edu))

## TCP Protocol Stack



## Data Formats

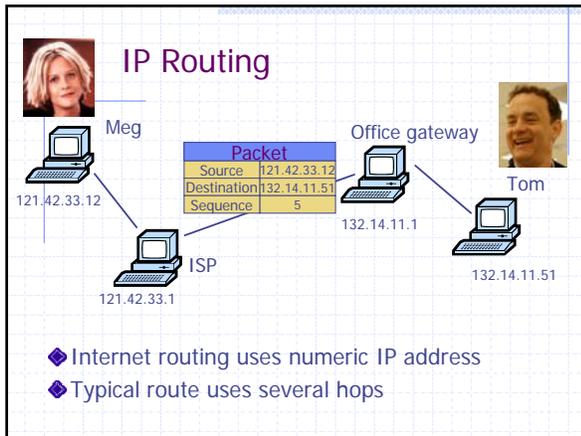


## IP

### Internet Protocol

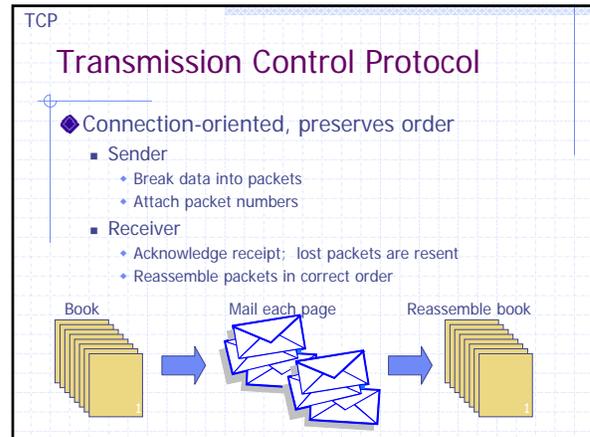
- ◆ Connectionless
  - Unreliable
  - Best effort
- ◆ Transfer datagram
  - Header
  - Data

Version	Header Length
Type of Service	
Total Length	
Identification	
Flags	Fragment Offset
Time to Live	
Protocol	
Header Checksum	
Source Address of Originating Host	
Destination Address of Target Host	
Options	
Padding	
IP Data	



- ## IP Protocol Functions (Summary)
- ◆ Routing
    - IP host knows location of router (gateway)
    - IP gateway must know route to other networks
  - ◆ Fragmentation and reassembly
    - If max-packet-size less than the user-data-size
  - ◆ Error reporting
    - ICMP packet to source if packet is dropped

- ## UDP
- ### User Datagram Protocol
- ◆ IP provides routing
    - IP address gets datagram to a specific machine
  - ◆ UDP separates traffic by port
    - Destination port number gets UDP datagram to particular application process, e.g., 128.3.23.3, 53
    - Source port number provides return address
  - ◆ Minimal guarantees
    - No acknowledgment
    - No flow control
    - No message continuation

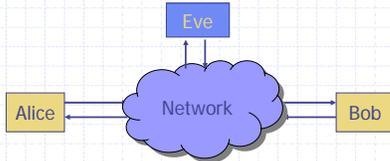


- ## ICMP
- ### Internet Control Message Protocol
- ◆ Provides feedback about network operation
    - Error reporting
    - Reachability testing
    - Congestion Control
  - ◆ Example message types
    - Destination unreachable
    - Time-to-live exceeded
    - Parameter problem
    - Redirect to better gateway
    - Echo/echo reply - reachability test
    - Timestamp request/reply - measure transit delay

- ## Basic Security Problems
- ◆ Network packets pass by untrusted hosts
    - Eavesdropping, packet sniffing (e.g., "ngrep")
  - ◆ IP addresses are public
    - Smurf
  - ◆ TCP connection requires state
    - SYN flooding attack
  - ◆ TCP state can be easy to guess
    - TCP spoofing attack

## Packet Sniffing

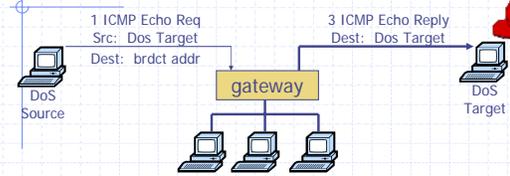
- ◆ Promiscuous NIC reads all packets
  - Read all unencrypted data (e.g., "ngrep")
  - ftp, telnet send passwords in clear!



Sweet Hall attack installed sniffer on local machine

Prevention: Encryption, improved routing (Next lecture: IPsec)

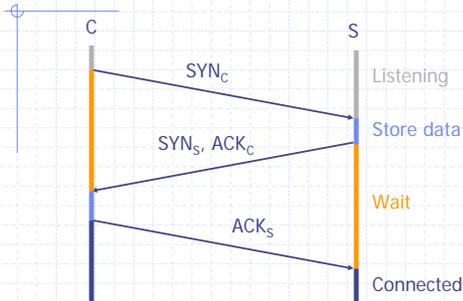
## Smurf DoS Attack



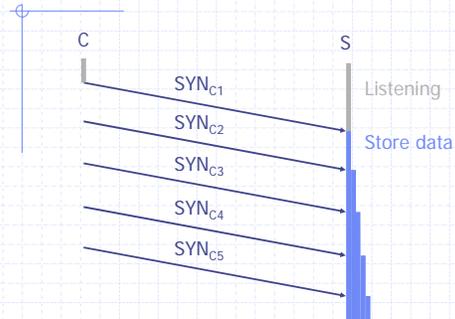
- ◆ Sending ping request to broadcast addr (ICMP Echo Req)
- ◆ Lots of responses:
  - Every host on target network generates a ping reply (ICMP Echo Reply) to victim
  - Ping reply stream can overload victim

Prevention: reject external packets to broadcast address

## TCP Handshake



## SYN Flooding



## SYN Flooding

- ◆ Attacker sends many connection requests
  - Spoofed source addresses
- ◆ Victim allocates resources for each request
  - Connection requests exist until timeout
  - Fixed bound on half-open connections
- ◆ Resources exhausted  $\Rightarrow$  requests rejected

## Protection against SYN Attacks

[Bernstein, Schenk]

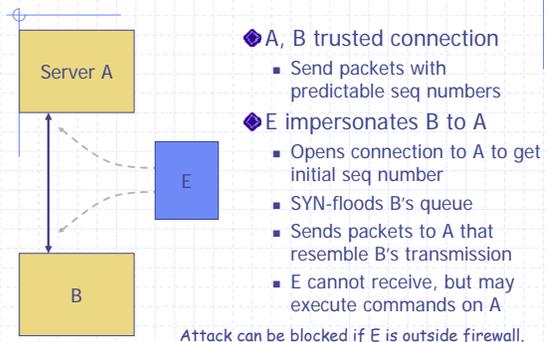
- ◆ Client sends SYN
- ◆ Server responds to Client with SYN-ACK cookie
  - $sqn = f(\text{src addr, src port, dest addr, dest port, rand})$
  - Normal TCP response but server does not save state
- ◆ Honest client responds with ACK(sq<sub>n</sub>)
- ◆ Server checks response
  - If matches SYN-ACK, establishes connection
    - "rand" is top 5 bits of 32-bit time counter
    - Server checks client response against recent values

See <http://cr.jp.to/syncookies.html>

## TCP Connection Spoofing

- ◆ Each TCP connection has an associated state
  - Client IP and port number; same for server
  - Sequence numbers for client, server flows
- ◆ Problem
  - Easy to guess state
    - Port numbers are standard
    - Sequence numbers often chosen in predictable way

## IP Spoofing Attack



## TCP Sequence Numbers

- ◆ Need high degree of unpredictability
  - If attacker knows initial seq # and amount of traffic sent, can estimate likely current values
  - Send a flood of packets with likely seq numbers
  - Attacker can inject packets into existing connection
- ◆ Some implementations are vulnerable

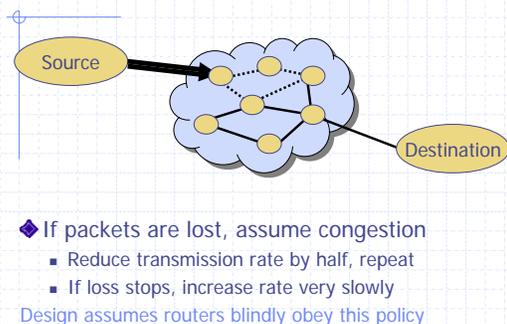
## Recent DoS vulnerability [Watson'04]

- ◆ Suppose attacker can guess seq. number for an existing connection:
  - Attacker can send Reset packet to close connection. Results in DoS.
  - Naively, success prob. is  $1/2^{32}$  (32-bit seq. #'s).
  - Most systems allow for a large window of acceptable seq. #'s
    - Much higher success probability.
- ◆ Attack is most effective against long lived connections, e.g. BGP.

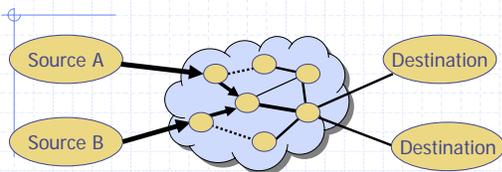
## Cryptographic network protection

- ◆ Solutions above the transport layer
  - Examples: SSL and SSH
  - Protect against session hijacking and injected data
  - Do not protect against denial-of-service attacks caused by spoofed packets
- ◆ Solutions at network layer
  - Use cryptographically random ISNs [RFC 1948]
  - More generally: IPsec
  - Can protect against
    - session hijacking and injection of data
    - denial-of-service attacks using session resets

## TCP Congestion Control



## Competition



- ◆ Amiable Alice yields to boisterous Bob
  - Alice and Bob both experience packet loss
  - Alice backs off
  - Bob disobeys protocol, gets better results

## Routing Vulnerabilities

- ◆ Source routing attack
  - Can direct response through compromised host
- ◆ Routing Information Protocol (RIP)
  - Direct client traffic through compromised host
- ◆ Exterior gateway protocols
  - Advertise false routes
  - Send traffic through compromised hosts

## Source Routing Attacks

- ◆ Attack
  - Destination host may use reverse of source route provided in TCP open request to return traffic
    - Modify the source address of a packet
    - Route traffic through machine controlled by attacker
- ◆ Defenses
  - Only accept source route if trusted gateways listed in source routing info
  - Gateway rejects external packets claiming to be local
  - Reject pre-authorized connections if source routing info present

## Routing Table Update Protocols

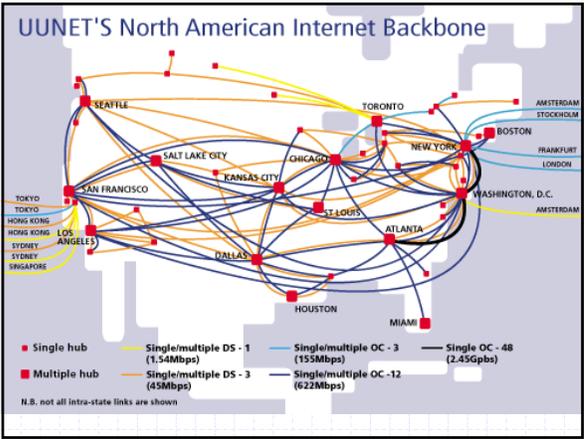
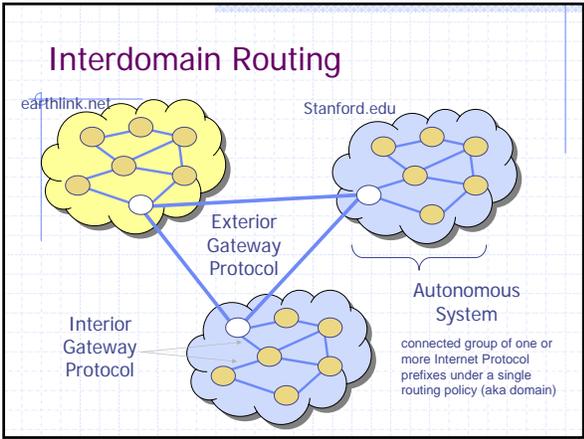
- ◆ Interior Gateway Protocols: IGPs
  - distance vector type - each gateway keeps track of its distance to all destinations
    - Gateway-to-Gateway: GGP
    - Routing Information Protocol: RIP
- ◆ Exterior Gateway Protocol: EGP
  - used for communication between different autonomous systems

## Routing Information Protocol (RIP)

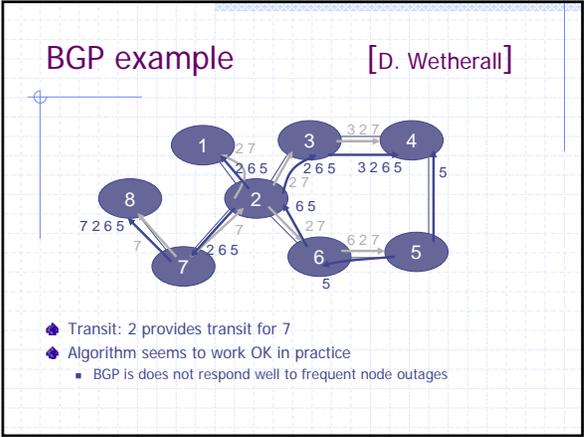
- ◆ Attack
  - Intruder sends bogus routing information to a target and each of the gateways along the route
    - Impersonates an unused host
      - Diverts traffic for that host to the intruder's machine
    - Impersonates a used host
      - All traffic to that host routed to the intruder's machine
      - Intruder inspects packets & resends to host w/ source routing
      - Allows capturing of unencrypted passwords, data, etc

## Routing Information Protocol (RIP)

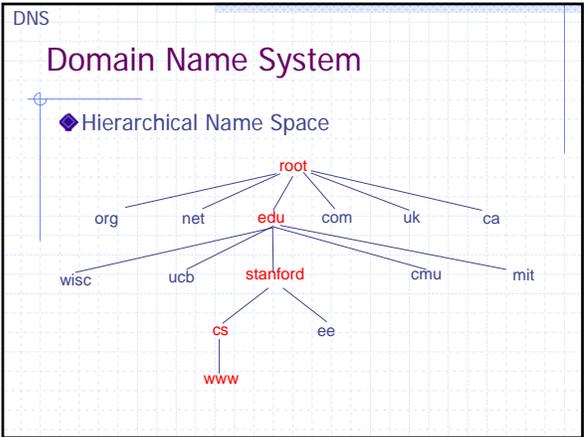
- ◆ Defense
  - Firewall at the gateway
    - Filters packets based on source and/or destination addresses
  - Don't accept new routes to local networks
    - Interferes with fault-tolerance but detects intrusion attempts
  - Authenticate RIP packets
    - Difficult in a broadcast protocol
    - Only allows for authentication of prior sender



- ### BGP overview
- ◆ Iterative path announcement
    - Path announcements grow from destination to source
    - Packets flow in reverse direction
  - ◆ Protocol specification
    - Announcements *can* be shortest path
    - Nodes allowed to use other policies
      - E.g., "cold-potato routing" by smaller peer
    - Not obligated to use path you announce

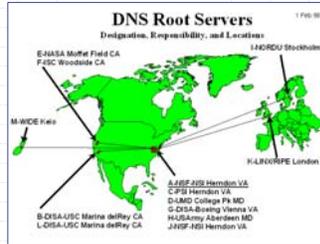


- ### Issues
- ◆ Security problems
    - Potential for disruptive attacks
    - BGP packets are un-authenticated
  - ◆ Incentive for dishonesty
    - ISP pays for some routes, others free

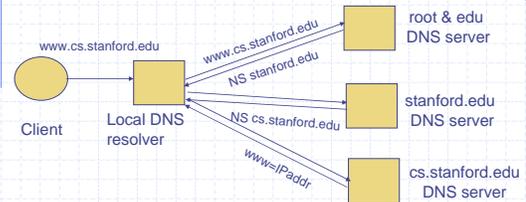


## DNS Root Name Servers

- ◆ Hierarchical service
  - Root name servers for top-level domains
  - Authoritative name servers for subdomains
  - Local name resolvers contact authoritative servers when they do not know a name



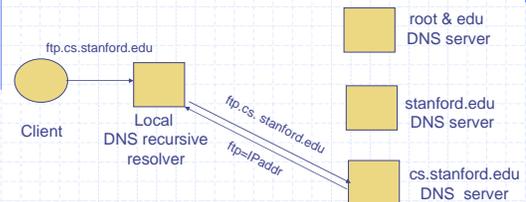
## DNS Lookup Example



## Caching

- ◆ DNS responses are cached
  - Quick response for repeated translations
  - Useful for finding servers as well as addresses
    - NS records for domains
- ◆ DNS negative queries are cached
  - Save time for nonexistent sites, e.g. misspelling
- ◆ Cached data periodically times out
  - Lifetime (TTL) of data controlled by owner of data
  - TTL passed with every record
- ◆ Some funny stuff allowed by RFC
  - Discuss cache poisoning in a few slides

## Lookup using cached DNS server



## DNS Implementation Vulnerabilities

- ◆ DNS implementations have had same kinds of vulnerabilities as other software
  - Reverse query buffer overrun in BIND Releases 4.9 (4.9.7 prior) and Releases 8 (8.1.2 prior)
    - gain root access
    - abort DNS service
  - MS DNS for NT 4.0 (service pack 3 and prior)
    - crashes on chargin stream
    - telnet ntbbox 19 | telnet ntbbox 53
- ◆ Moral
  - Better software quality is important
  - Defense in depth!

## Inherent DNS Vulnerabilities

- ◆ Users/hosts typically trust the host-address mapping provided by DNS
- ◆ Obvious problems
  - Interception of requests or compromise of DNS servers can result in incorrect or malicious responses
  - Solution – authenticated requests/responses
- ◆ Some funny stuff allowed by RFC
  - Name server may delegate name to another NS (this is OK)
  - If name is delegated, may also supply IP addr (this is trouble)
  - Details in a couple of slides

## Bellovin/Mockapetris Attack

- ◆ Trust relationships use symbolic addresses
  - /etc/hosts.equiv contains friend.stanford.edu
- ◆ Requests come with numeric source address
  - Use reverse DNS to find symbolic name
  - Decide access based on /etc/hosts.equiv, ...
- ◆ Attack
  - Spoof reverse DNS to make host trust attacker

## Reverse DNS

- ◆ Given numeric IP address, find symbolic addr
- ◆ To find 222.33.44.3,
  - Query 44.33.222.in-addr.arpa
  - Get list of symbolic addresses, e.g.,

1	IN	PTR	server.small.com
2	IN	PTR	boss.small.com
3	IN	PTR	ws1.small.com
4	IN	PTR	ws2.small.com

## Attack

- ◆ Gain control of DNS service for evil.org
- ◆ Select target machine in good.net
- ◆ Find trust relationships
  - SNMP, finger can help find active sessions, etc.
  - Example: target trusts host1.good.net
- ◆ Connect
  - Attempt rlogin from coyote.evil.org
  - Target contacts reverse DNS server with IP addr
  - Use modified reverse DNS to say "addr belongs to host1.good.net"
  - Target allows rlogin

## Defense against this attack

- ◆ Double-check reverse DNS
  - Modify rlogind, rshd to query DNS server
  - See if symbolic addr maps to numeric addr
  - But then must deal with DNS cache poisoning ...
- ◆ Authenticate entries in DNS tables
  - Relies on some form of PKI?
  - Next lecture ...

See <http://cr.yp.to/djbdns/notes.html>

## DNS cache poisoning

- ◆ DNS resource records (see RFC 1034)
  - An "A" record supplies a host IP address
  - A "NS" record supplies name server for domain
- ◆ Example
  - www.evil.org NS ns.yahoo.com /delegate to yahoo
  - ns.yahoo.com A 1.2.3.4 / address for yahoo
- ◆ Result
  - If resolver looks up www.evil.org, then evil name server will give resolver address 1.2.3.4 for yahoo
  - Lookup yahoo through cache goes to 1.2.3.4

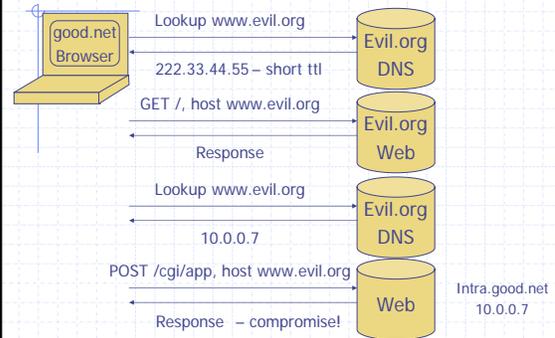
## Pharming

- ◆ DNS poisoning attack (less common than phishing)
  - Change IP addresses to redirect URLs to fraudulent sites
  - Potentially more dangerous than phishing attacks
  - No email solicitation is required
- ◆ DNS poisoning attacks have occurred:
  - January 2005, the domain name for a large New York ISP, Panix, was hijacked to a site in Australia.
  - In November 2004, Google and Amazon users were sent to Med Network Inc., an online pharmacy
  - In March 2003, a group dubbed the "Freedom Cyber Force Militia" hijacked visitors to the Al-Jazeera Web site and presented them with the message "God Bless Our Troops"

## JavaScript/DNS intranet attack (I)

- ◆ Consider a Web server `intra.good.net`
  - IP: `10.0.0.7`, inaccessible outside `good.net` network
  - Hosts sensitive CGI applications
- ◆ Attacker at `evil.org` wishes to subvert
- ◆ Gets `good.net` user to browse `www.evil.org`
- ◆ Places JS that has accesses web app on `intra.good.net`
  
- ◆ This doesn't work: JS enforces "same-origin" policy
- ◆ But: attacker controls `evil.org` DNS ...

## JavaScript/DNS intranet attack (II)



## Summary (I)

- ◆ Eavesdropping
  - Encryption, improved routing (Next lecture: IPsec)
- ◆ Smurf
  - Drop external packets to brdcst address
- ◆ SYN Flooding
  - SYN Cookies
- ◆ IP spoofing
  - Use less predictable sequence numbers

## Summary (II)

- ◆ Source routing attacks
  - Additional info in packets, tighter control over routing
- ◆ Interdomain routing
  - Authenticate routing announcements
  - Many other issues
- ◆ DNS attacks
  - Double-check reverse DNS
  - Authenticate entries in DNS tables
  - Do not trust addresses except from authoritative NS