

# TCP/IP Security Problems

- History that still teaches

## remote login without a password

- `rsh` and `rcp` were programs that allowed you to login from a remote site without a password
  - The `.rhosts` file in your home directory is an access control list (ACL)
  - Example `.rhosts` file:

```
red.cs.umass.edu  brian
blue.cs.umass.edu brian
*.cs.umass.edu   brian
*      *
```

- The authentication check in `rsh` (and the other `r-tools`) is simply the IP address.
  - Is it hard to *spoof* a particular IP address?
- These attacks hold for any IP-based 'authentication

# IP Stack Reviewed (briefly)

- |                 |                                  |
|-----------------|----------------------------------|
| 1. Physical     | 1. Physical security, hubs, wlan |
| 2. Link         | 2. Arp attacks, NAT, DHCP        |
| 3. Network      | 3. Spoofing, routing, etc        |
| 4. Transport    | 4. Hijacking                     |
| 5. Session      | 5. Session stealing              |
| 6. Presentation | 6. ===== (XML?)                  |
| 7. Application  | 7. Web apps, DNS                 |

# Routing Review

- Review: When they receive a packet, how do hosts using IP route data?
  - Static routing is largely based on subnetting, ARP, and ICMP.
- IP hosts are always on some specific subnet. They search routing tables looking for longest matching prefix.
- This means, you find routes in this order:
  1. Matching host address (128.119.48.55)
  2. Matching subnet address (128.119.48.\*)
  3. Matching network address (128.119.\*)
  4. Default route (gateway router)
- This process tells the host what IP address is the next hop.
- Now the host must determine the link layer address of the next hop. How is that done in IP? ...

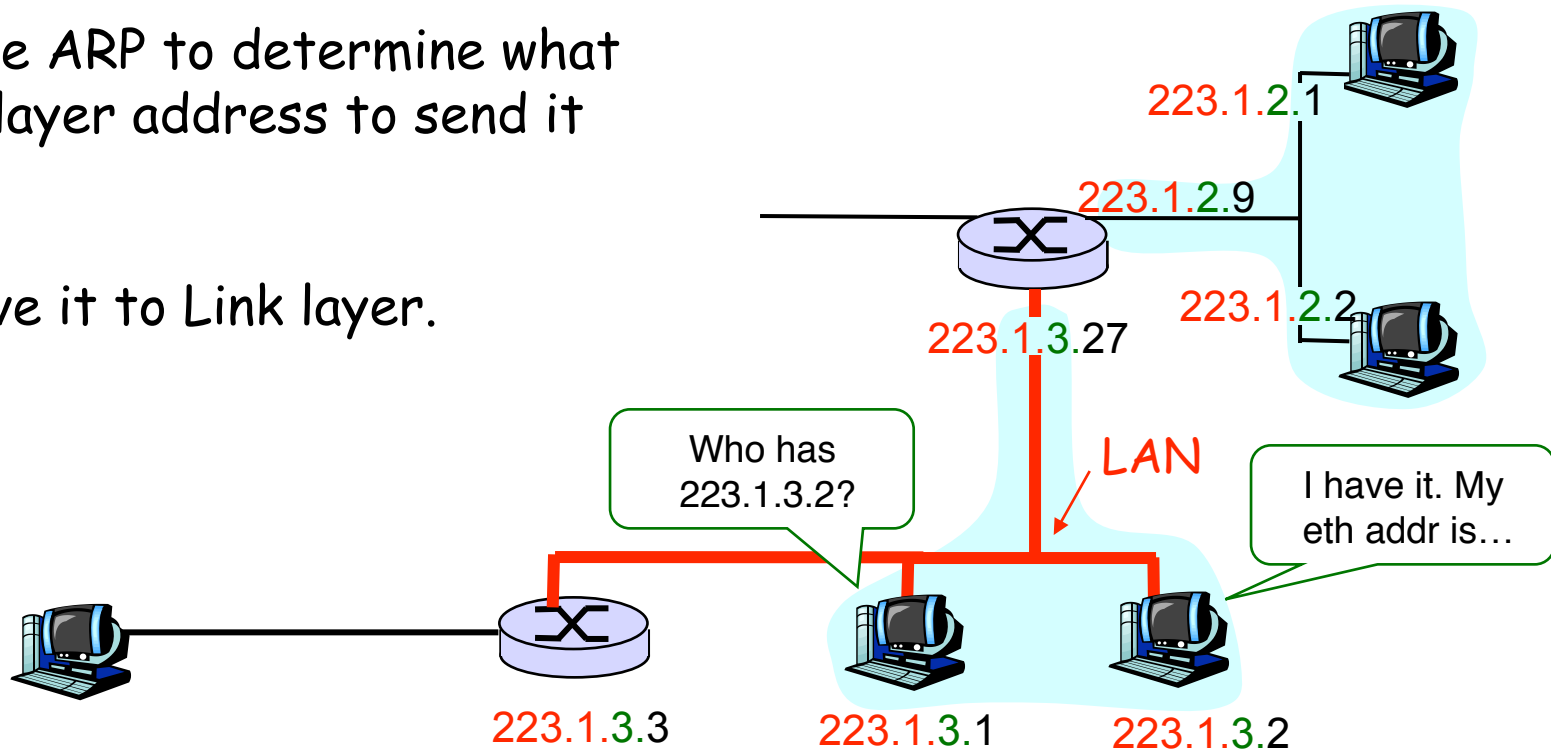
## Review:

# Address Resolution Protocol (ARP) and ICMP

- ARP is the interface between the *Link* layer and *Network* layer.
  - Allows hosts to query who owns an IP address on the same LAN.
  - Owner responds with hardware address.
  - Allows changes to link layer to be independent of IP addressing.
  - That's why we can have IP on everything (wireless, radio waves, buses, etc.)
- ICMP is used for routing *error* messages
  - "TTL expired" (that's how traceroute works)
  - "Host unreachable"
  - "Echo request" (that's how the ping program works)

# On-the-same-LAN routing

1. Route lookup determines it is on the same subnet.
2. Use ARP to determine what link layer address to send it to.
3. Give it to Link layer.



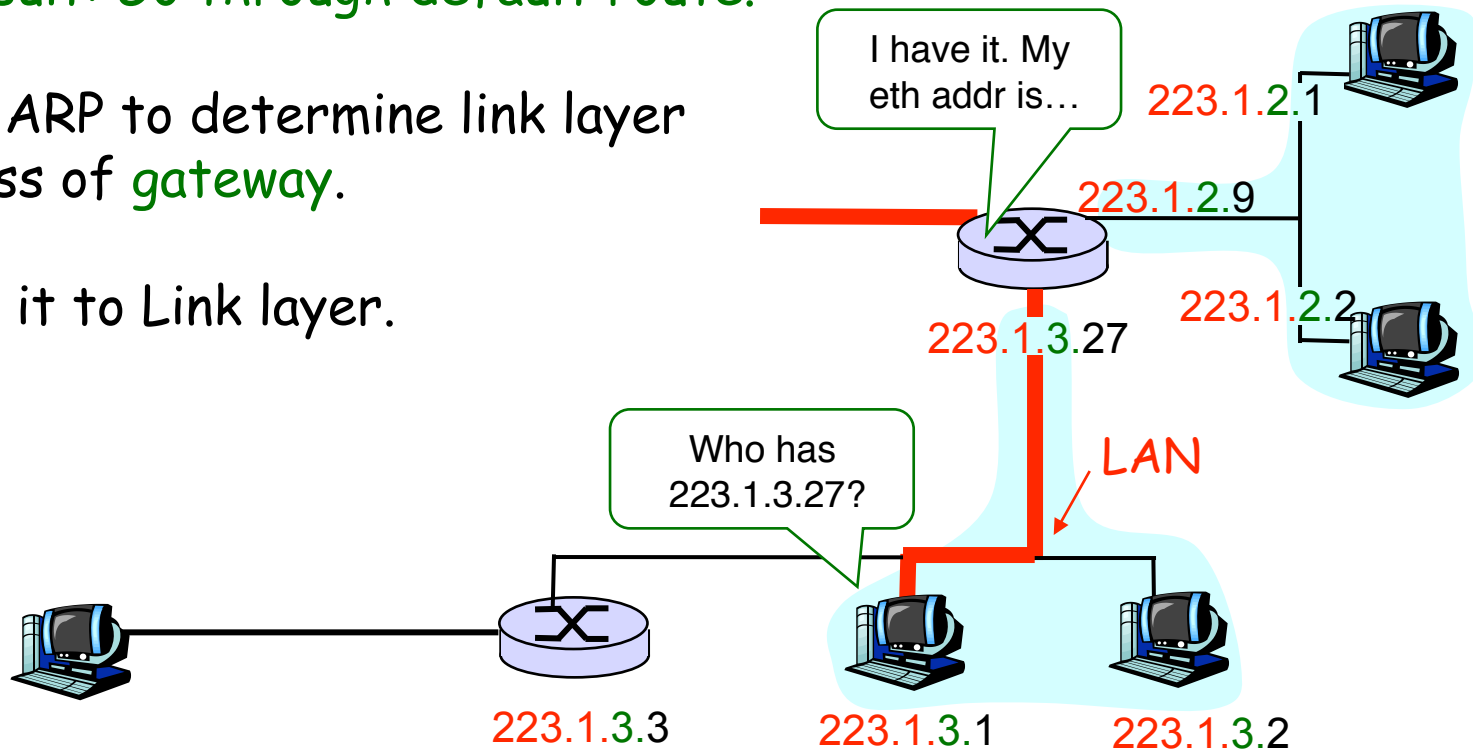
# Through-the-gateway Routing

1. Route lookup determines it's on a different subnet.

Result: Go through default route.

2. Use ARP to determine link layer address of gateway.

3. Give it to Link layer.



# ARP Attacks

- When a machine sends an ARP request out, attackers can reply, falsely stating they own the address.
  - But this starts a race condition with the real machine.
- Unfortunately, ARP will just accept replies without requests!
  - Some OS's provide defenses to this
- Just send a spoofed reply message saying your MAC address owns a certain IP address.
  - Repeat frequently so that cache doesn't timeout
- Messages are routed through you to sniff or modify.

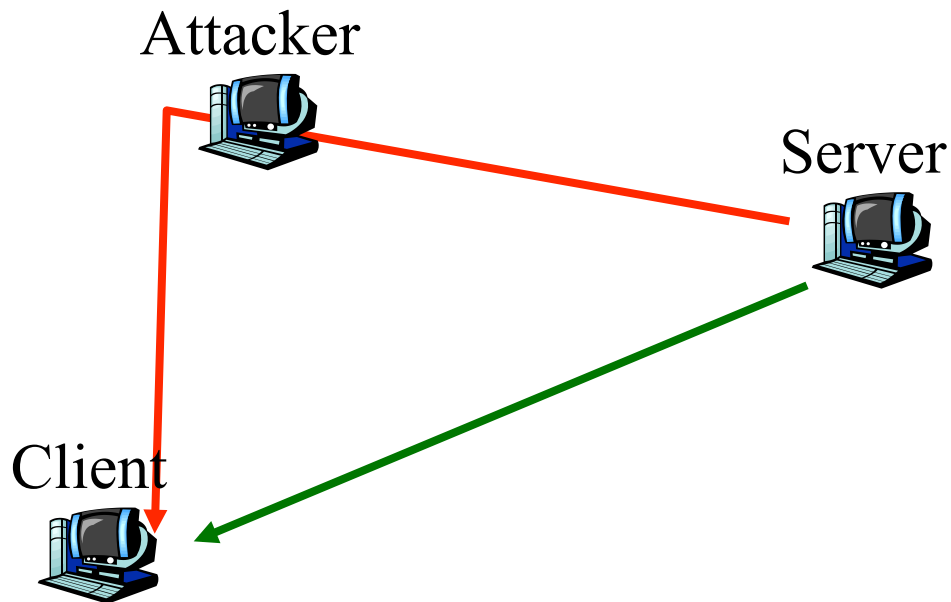
# ARP Spoofing - Countermeasures

- “Publish” MAC address of router/default gateway and trusted hosts to prevent ARP spoof.
  - Statically define the IP to Ethernet address mapping.  
(“Publish” is a poor term: it’s not sent on the network)
  - This prevents someone from fooling the host into sending network traffic to a host masquerading as the router or another host via an ARP spoof.
  - Here’s how you do it in linux:  

```
arp -s hostname 00:01:02:03:04:ab pub
```
- This is generally viewed as unworkable
- Can we limit who has access to the network?
  - NAC, etc

# Attacking IP Routing

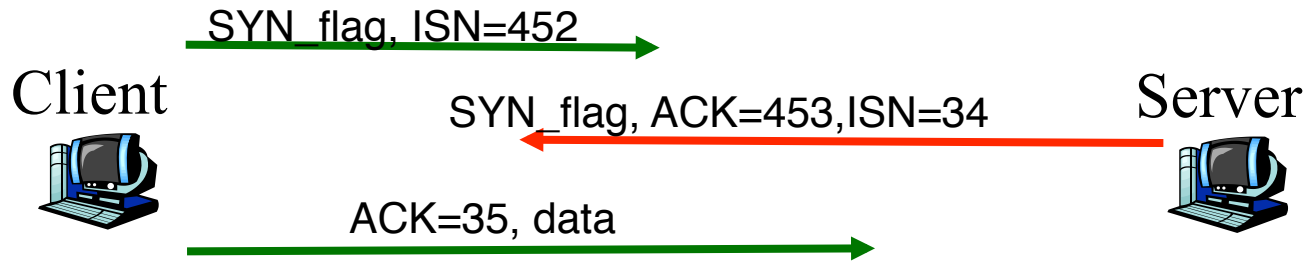
- Types of routing:
  1. dynamic intranet routing
  2. static intranet routing
  3. BGP routing
  4. Ad hoc wireless network routing
- Attacking the routing can cause
  - attacker-in-the middle eavesdropping (passive)
  - attacker-in-the-middle modifications (active)
  - Black holes in routing (DoS)
  - Redirected flooding attacks (DoS)



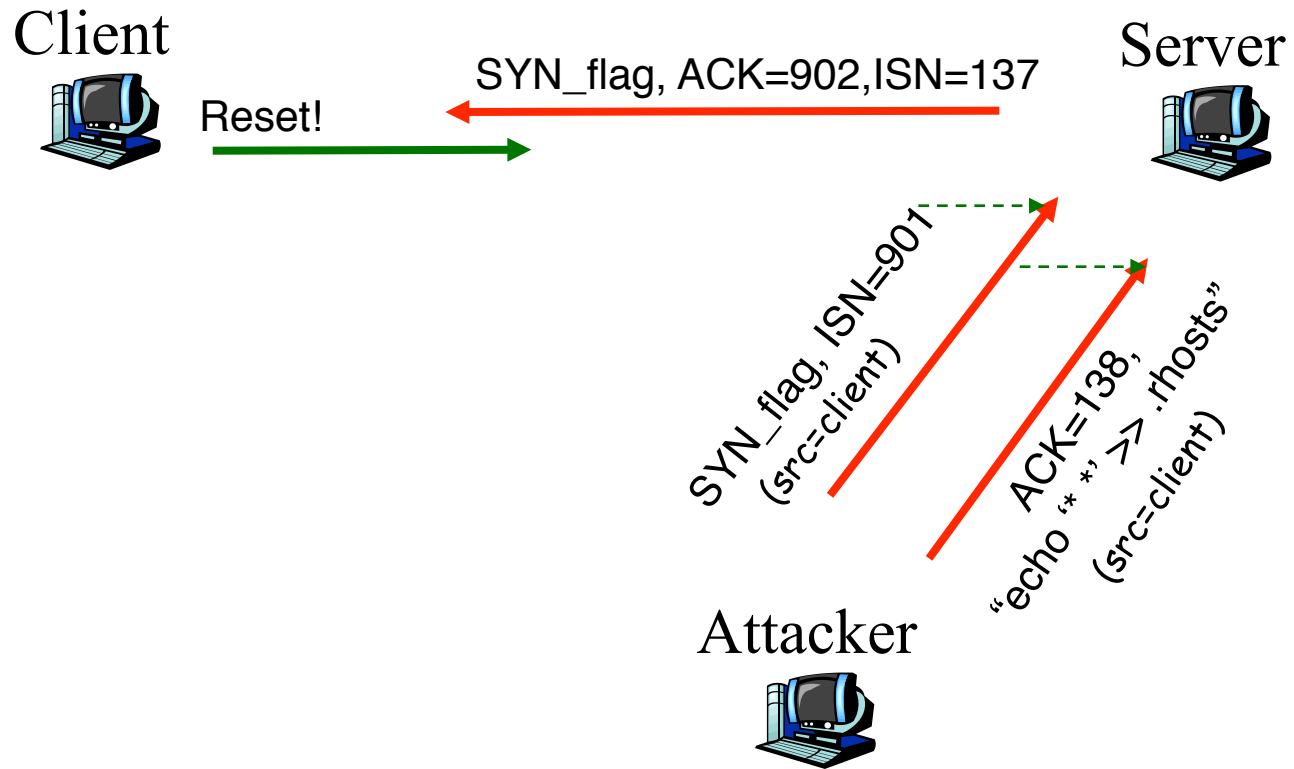
# Dynamic Routing

- An attacker can falsify routing updates send between routers.
  - Attacker injects a RIP/eIGRP/OSPF update stating she has a path to a particular (unused) host. (any unicast protocol will do)
  - All subsequent packets will be routed to her.
  - She uses `rsh` to log into the machine.
    - This is also a DOS attack and a traffic redirection attack (for sniffing or modification)
- Similar attacks exist for inter-domain routing protocols, like BGP.
- Defense: Requires secure routing protocols to defend against this attack.
  - Routers should accept only *authenticated* updates.
  - Requires key management and pre-configuration among routers.

# Normal TCP Three-way Handshake



# Blind Spoofing



# Blind Spoofing

- Normal TCP operation from **client**, C, to **server**, S.
  - C→S: SYN\_flag, ISN=x
  - S→C: SYN\_flag, ISN=y, ACK=x+1
  - C→S: ACK=y+1
  - Client and Server exchange **data**
- **Blind spoofing**. Find a client machine that's off. Guess the ISN of the server. Usually in regular increments. Use rsh to log in:
  - X(as C)→S: SYN\_flag, ISN=a [spoofs C]
  - S→C: SYN\_flag, ISN=b, ACK=a+1
  - X(as C)→S: ACK=b+1 [spoofs C]
  - X(as C)→S: [echo "\*" \*] >> ~/.rhosts] [spoofs C]
  - X(as C)→S: RESET [spoofs C]
  - X now rlogins from anywhere in the world.

# Blind Spoofing

- If C is still up, then C will send a reset message to the server thinking it's an error.
- So, either L
  - use a network address that is not in use.
  - Do a **denial of service** (DoS) attack on a machine so it can't answer.
- “Morris found that by impersonating a server port on C, and by flooding that port with apparent connection requests, he could generate queue overflows that would make it likely that the S→C message would be lost.”
- This is SYN flooding...

## An aside: SYN Flooding DoS

- Pick a machine, any machine.
- Spoof packets to it (so you don't get caught)
- Each packet is the first hand of the 3-way handshake of TCP: send a SYN packet.
- Send lots of SYN packets.
  
- Each SYN packet received causes a buffer to be allocated, and the limits of the `listen()` call to be reached.
  
- Morris invented SYN flooding just to launch a blind spoofing attack; later used by others against Yahoo!

# TCP Session Stealing

- A.k.a. IP splicing, TCP Hijacking
  - Read a detailed account  
“A Simple Active Attack Against TCP” by Laurent Joncheray.  
In *Proceedings of 5th USENIX Unix Security Symposium*.  
June 1995
- Running code widely available.
- Defense: use ssh

# Desynchronizing the client and server

- Often during normal TCP operation, the client and server become *desynchronized*.
- E.g., sometimes the client will send a retransmission that actually isn't needed by the server.
- The server will drop the incoming packets.
- The attack: during a quiet period, the attacker sends a large amount of null data.
  - Specifically, the attacker sends as many bytes as there are in the sender's receive buffer.

# The Attack

- If the client receives packets that are a window-of-packets ahead of what it is expecting, the client will drop the unlooked for data. (this is partly due to flow control)
- Null data desynchronization
  - First, the attacker watches the session without interfering.
  - During a quiet period, the attacker sends a large amount of null data.
  - Specifically, the attacker sends as many bytes as there are in the sender's receive buffer.
  - Each packet contains NOP bytes, normally used to pad the packets for the purposes of checksums.
  - Now, when the client sends data, it is dropped by the server because it's lower than the server's window.
  - The attacker does the same with the client.
- Attacker is now a woman/man/bot in the middle!

## Attacker-in-the-Middle

- Data from the client can be re-packaged into a TCP packet and sent to the server, so there is no noticeable changes.
- Attacker can insert commands into the remote account. E.g.
  - `echo "mymachine.umass.edu mitnick" > .rhosts`
- However, commands entered by the attacker might appear on a command line history.
- **Defense: ssh connection, or IPsec**