Protocol Stack Security

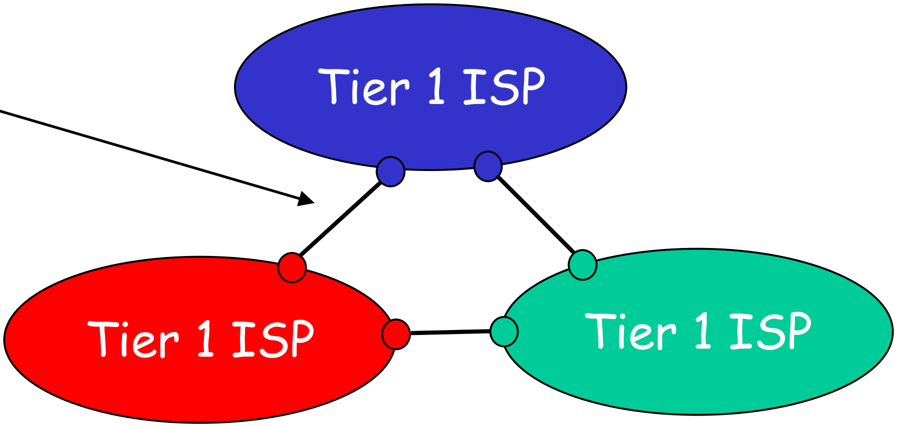
PHAM VAN HAU, PHD. EMAIL: PVHAU@HCMIU.EDU.VN

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING,
INTERNATIONAL UNIVERSITY

Internet structure: network of networks

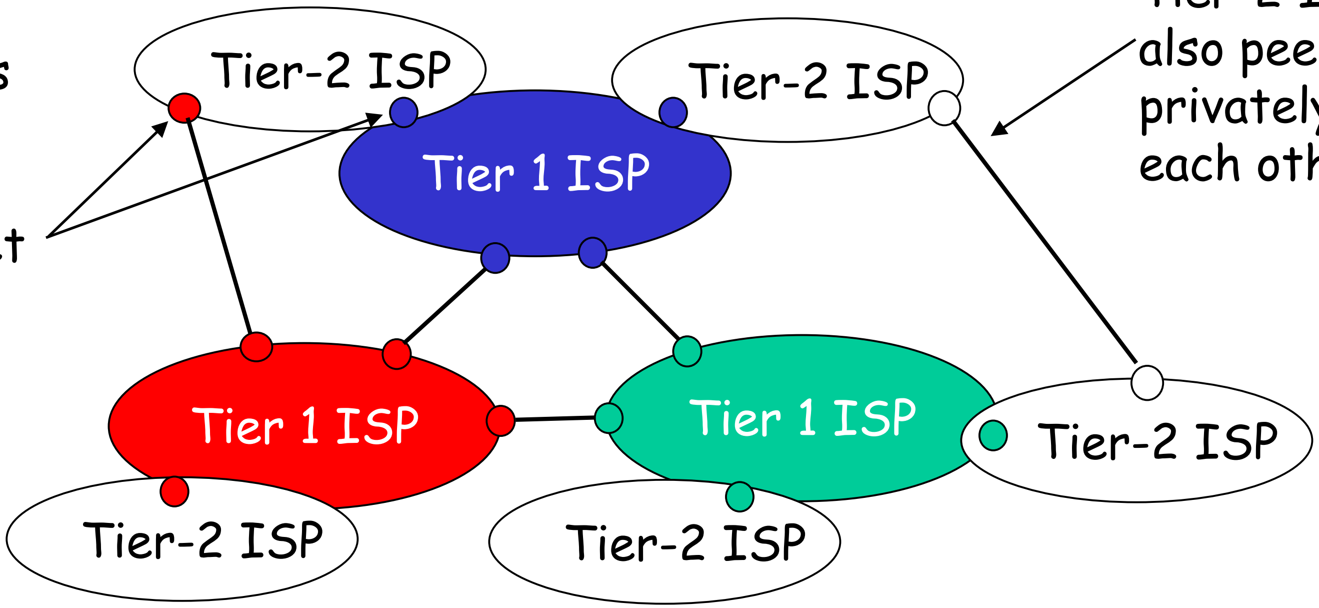
- roughly hierarchical
- at center: "tier-1" ISPs (e.g., Verizon, Sprint, AT&T, Cable and Wireless), national/international coverage
  - treat each other as equals

Tier-1 providers interconnect (peer) privately



Internet structure: network of networks

- "Tier-2" ISPs: smaller (often regional) ISPs
  - Connect to one or more tier-1 ISPs, possibly other tier-2 ISPs



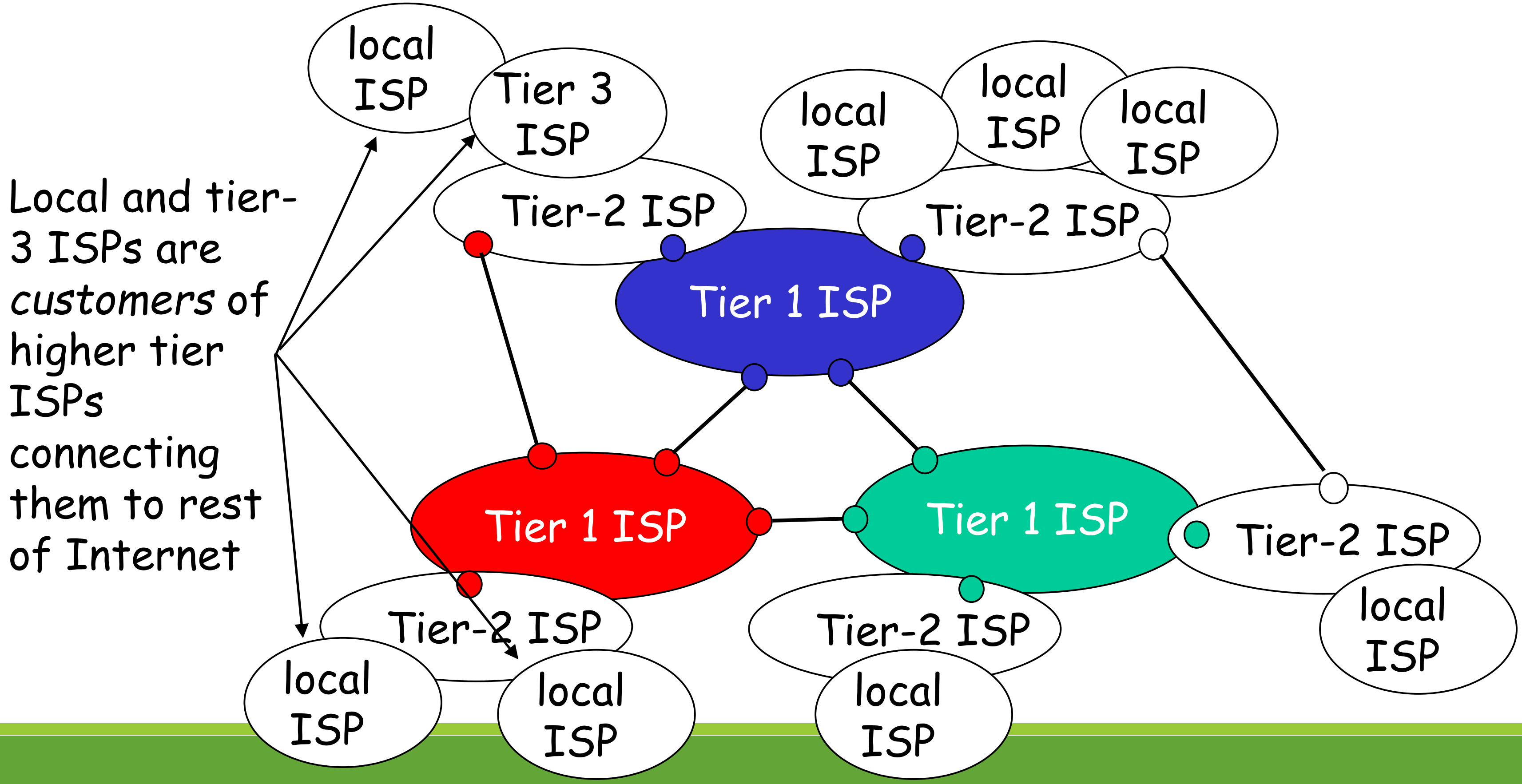
Tier-2 ISP pays tier-1 ISP for connectivity to rest of Internet

- tier-2 ISP is customer of tier-1 provider

Tier-2 ISPs also peer privately with each other.

Internet structure: network of networks

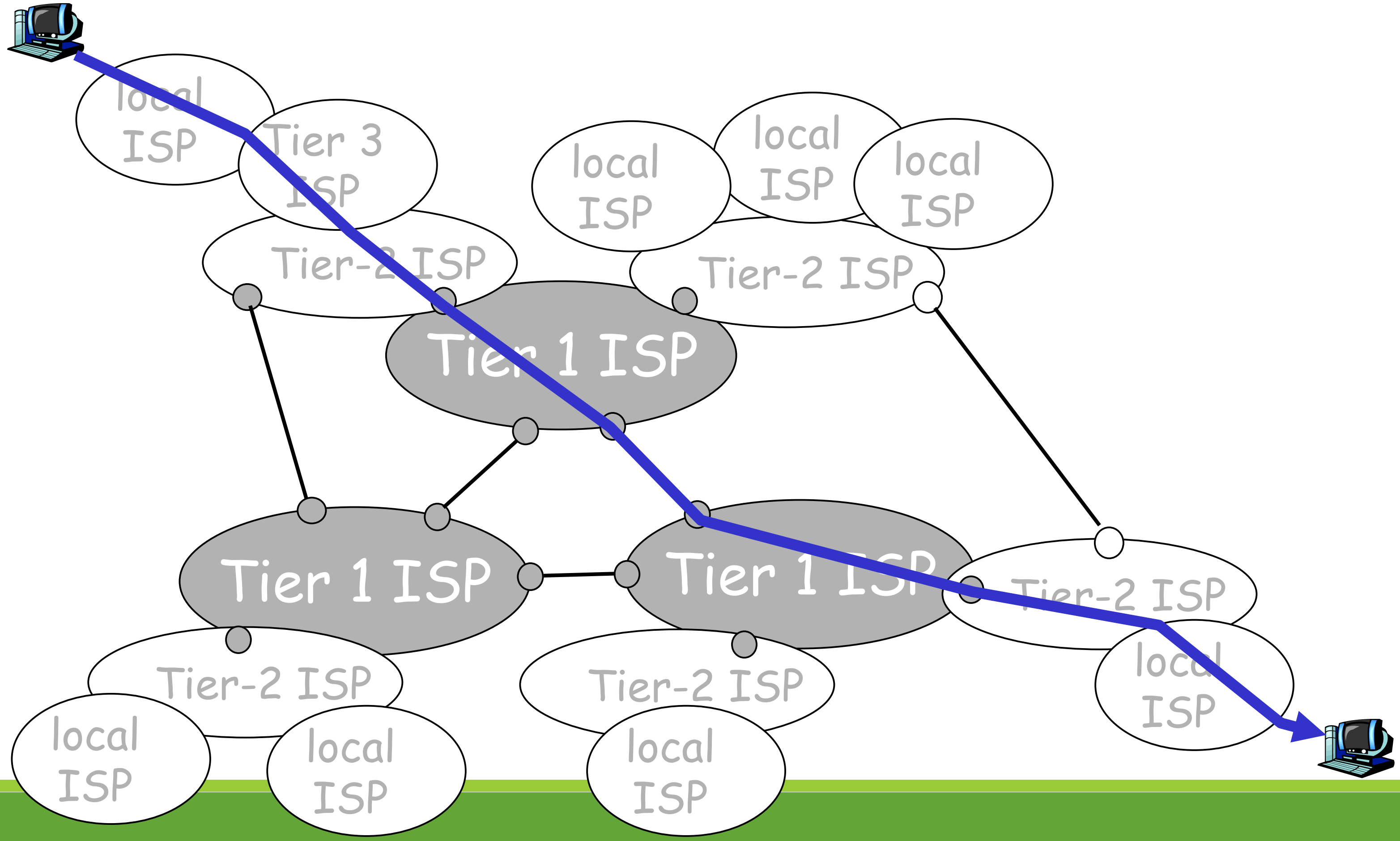
- "Tier-3" ISPs and local ISPs
  - last hop ("access") network (closest to end systems)



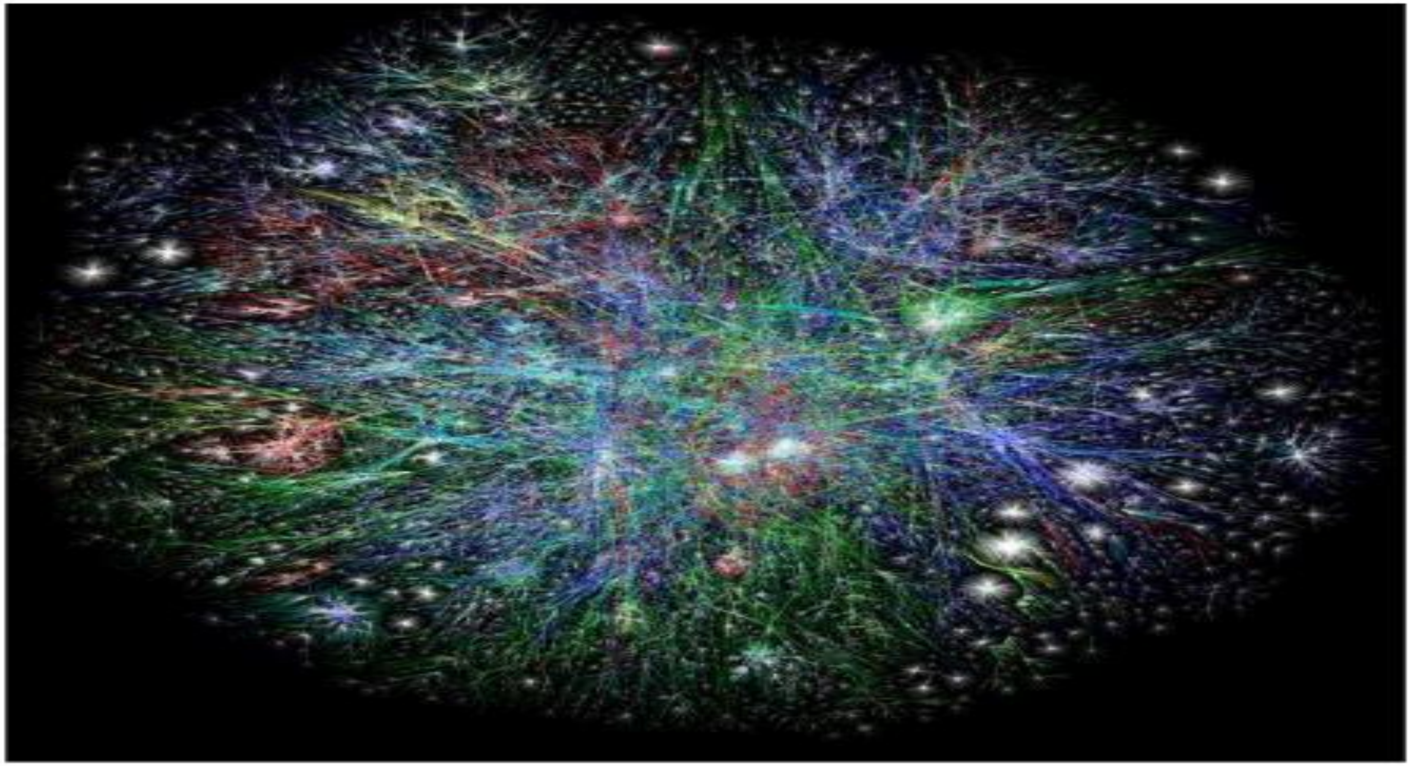
Local and tier-3 ISPs are *customers* of higher tier ISPs connecting them to rest of Internet

Internet structure: network of networks

- a packet passes through many networks!



Internet structure: network of networks

- 

“Complexity is the worst enemy of security.”
(Security expert Bruce Schneier)

Software bug

- Some vendors are trying to rush to market with their eyes set on functionality, not security
- A majority of security professionals themselves are not software engineer
- The computing society is used to receive software with bugs and then patch it
- Software vendors have not been held liable for insecure code
- Programmers are not taught secure coding in school

Carnegie Mellon University estimates that there are 5 to 15 bugs in every 1000 lines of code

Windows 2000 has 40-60 millions lines of code

Internet: A result of ad-hoc development

Security Context Change

Internet ecosystem changes: there are more and more threat agents that exploit the vulnerability of Internet infrastructure.

Attacks on datalink layer

ARP Attack

ARP does not provide any means of authentication

Attacks

- Racing against the queried host is possible
  - provide false IP address/link-level address mapping
- Fake ARP queries
  - used to store wrong ARP mappings in a host cache

=> result in a redirection of traffic to another host

ARP messages are sent continuously to have caches keep the faked entries

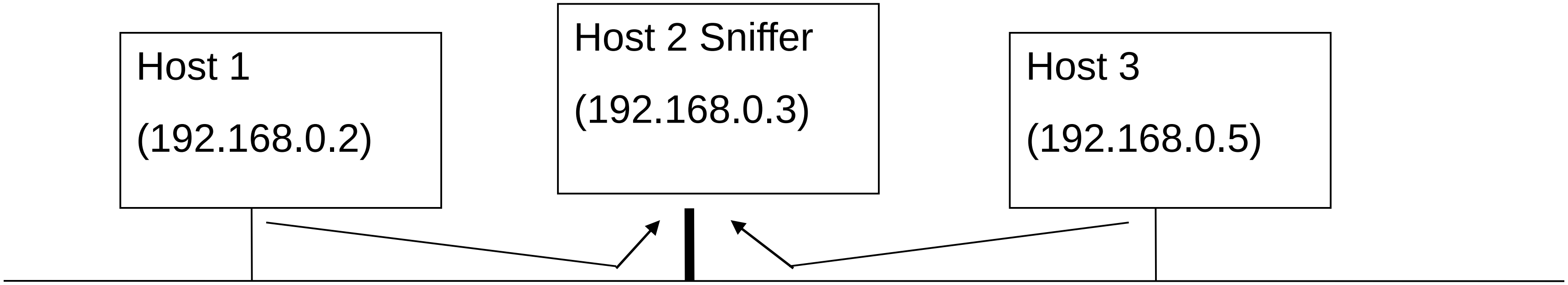
- can be used to impersonate the gateway and filter ALL the traffic
- OR: use ARP to map gateway IP to non-existent MAC address (denial-of-service)

Network sniffing

- Is the base for many attacks
  - attacker sets computer‘s NIC into **promiscuous mode**
  - NIC delivers all arriving packets to IP layer
  - can access all the traffic on the segment
- Many protocols transfer authentication information in

cleartext => collect username/password etc.

- Many tools available: tcpdump -x, dsniff etc.



Switch and Hub

- HUB works on Physical layer whereas SWITCH works on data link layer.
- A, B, C are connected to a HUP, C can capture traffic exchanged between A and B.
- A, B, C are connected to a SWITCH, C can not capture traffic exchanged between A and B.
- Switch also maintains MAC address tables.

Switch attack

Is sniffing also possible at switched Ethernet, where the switch only forwards the right packets to your host? YES!

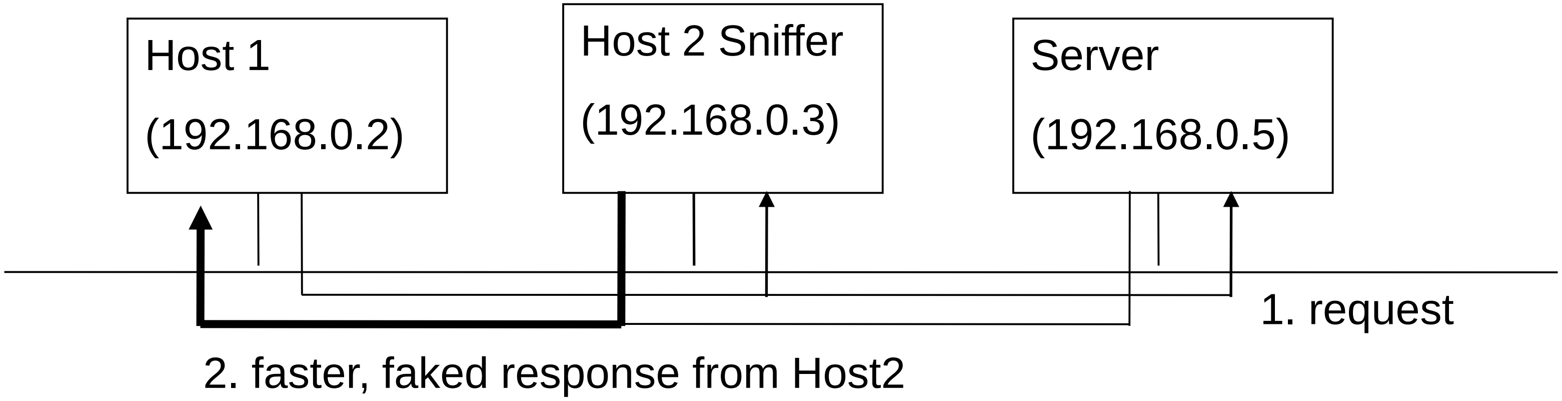
- MAC flooding
  - Switch maintains table with MAC address/port mappings
  - flooding switch with bogus MAC addresses will overflow table
  - switch will revert to hub mode
- MAC duplicating/cloning
  - you can buy NICs with reconfigurable MAC addresses
  - switch will record this in table and sends traffic to you

Attacks on network layer

IP Spoofing

= impersonating another host by sending a datagram with a faked IP-address

- used to impersonate sources of security critical info
- explicit address-based authentication
  - RPC, DNS
  - “r-“ commands (rsh, rcp, etc).



IP Spoofing

How can you do it on your own?

- open a RAW socket
  - socket(AF_INET, SOCK_RAW, IPPROTO_RAW)
- craft the packet
  - with faked IP address
  - including all headers with all attributes set correctly
  - including data
  - including checksums (TCP: required, UDP: recommended)
- send the packet using the RAW socket

Indirect Delivery: Routing

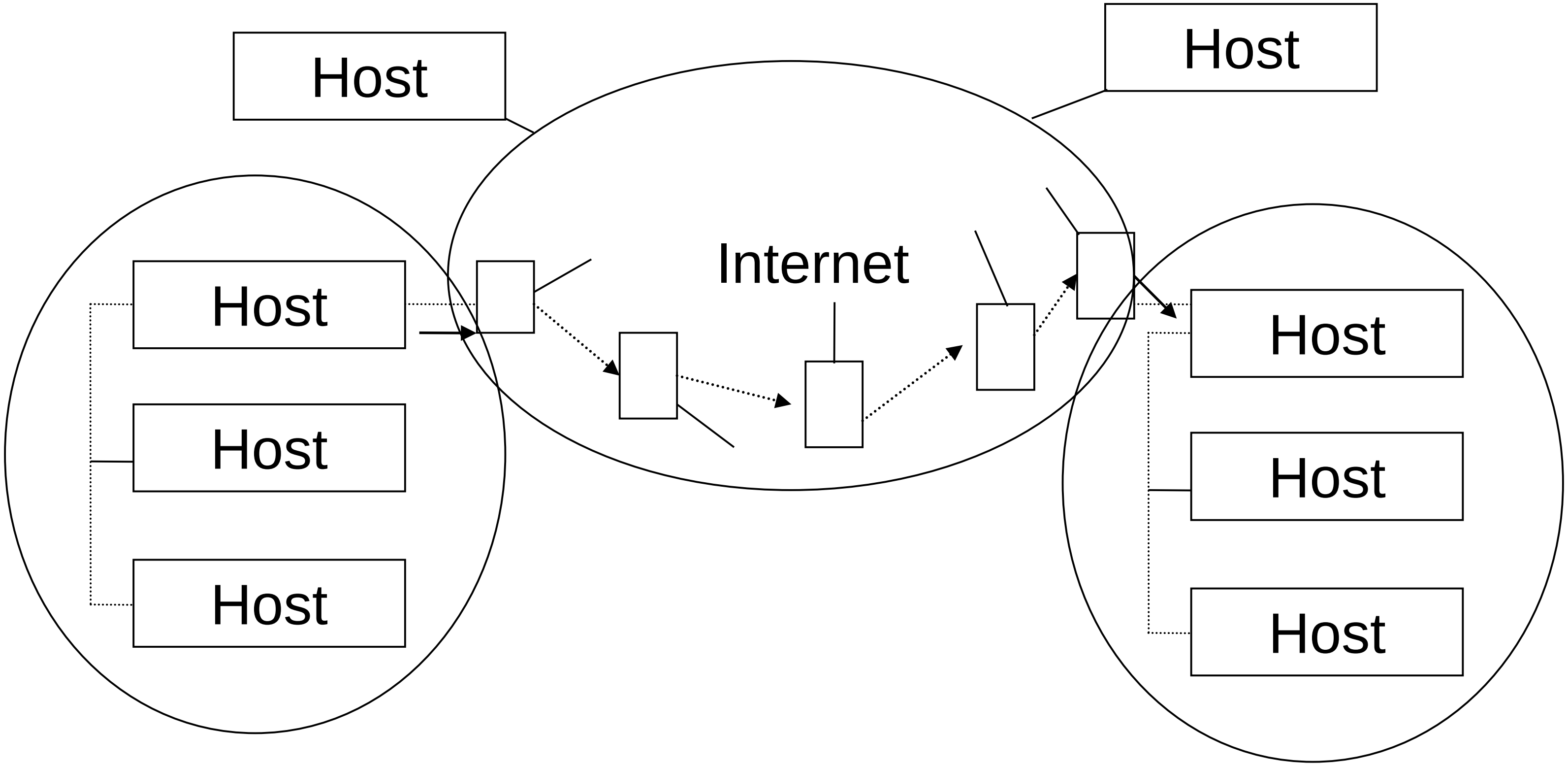
If hosts are in different physical networks packet can‘t be delivered directly

Packet is forwarded to a **gateway**

- has access to other network(s)
- decides upon destination where to send the packet next
- this is repeated until packet arrives at network with target host
- then direct delivery is performed
- link level addresses change at every step, also TTL field

Indirect Delivery: Routing

Store and forward communication

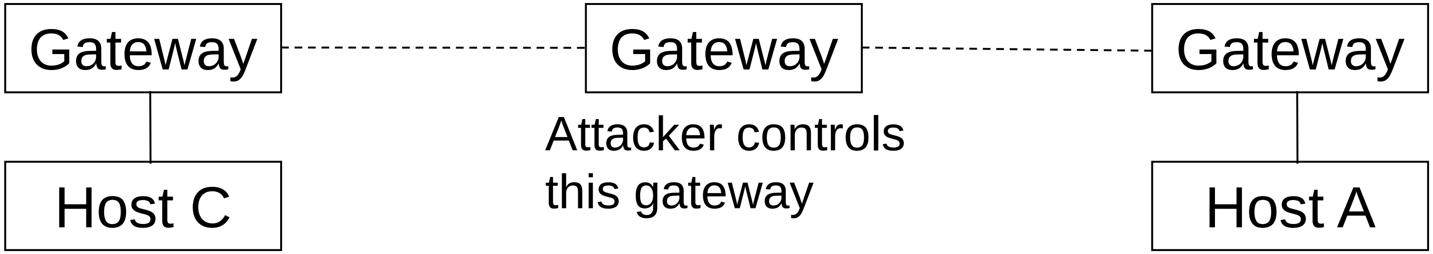


Man-in-the-Middle Attack

Attacker controls a gateway that is used in the delivery process can

- sniff the traffic
- intercept/block/delay traffic
- modify traffic

works only properly if attacker is on “best“ route



Man-in-the-Middle Attack

not easy in the Internet because of hop-by-hop routing

unless you control one of the backbone hosts

or source routing is used

- The IP source routing option can be used to specify the route to be used in the delivery process independent of the “normal“

delivery mechanisms

Man-in-the-Middle Attack

- the traffic can be forced through specific routes (=specific hosts)

- if the reverse route is used to reply to traffic, a host on the route can easily impersonate another host

- can be used to abuse a trust relationship

RIP Attacks

A host can send spoofed RIP packets in order to

“inject“ routes into a host (requires only IP/UDP spoofing)

a route with a smaller hop count would be used

This attack can be used for

- Hijacking
- DOS

On a LAN with RIPv2 passwords have to be used for updating routes, but plaintext passwords are used

- can be sniffed

Attack on transport layer

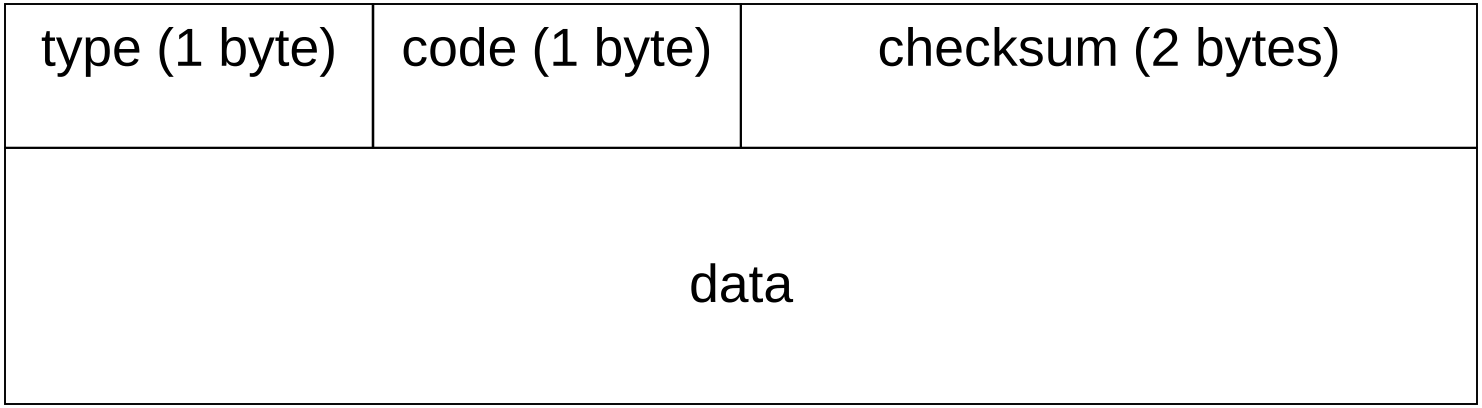
ICMP

ICMP (Internet Control Message Protocol)

- is used to exchange control/error messages about the delivery of IP datagrams
- ICMP messages are encapsulated inside IP datagrams

- ICMP messages can be:
  - Requests
  - Responses
  - Error messages
    - includes header and first 8 bytes of offending IP datagram

ICMP Message Format



type (1 byte)

code (1 byte)

checksum (2 bytes)

data

type field: specifies the class of the ICMP message

code field: specifies the exact type of the message

ICMP Messages

- Echo request/reply
  - used to test connectivity (ping)
- Time exceeded
  - used to report expired datagrams (TTL=0)
- Redirect
  - used to inform hosts about better routes (gateways)
- Destination unreachable
  - used to inform a host that it is impossible to deliver traffic to a specific destination

ICMP Messages

- Address mask request/reply
  - used by diskless systems to obtain the network mask at boot time
- Timestamp request/reply
  - used to synchronize clocks
- Source quench
  - used to inform about traffic overloads
- Parameter problem
  - used for inform about errors in the IP datagram fields

ICMP Echo

- Used by the ping program

type (1 byte)	code (1 byte)	checksum (2 bytes)
identifier (2 bytes) = Process ID		sequence number (2 bytes)
data		

identifier is used by “ping” to deliver back the packet to the right process (allowing more than one ping to run concurrently)

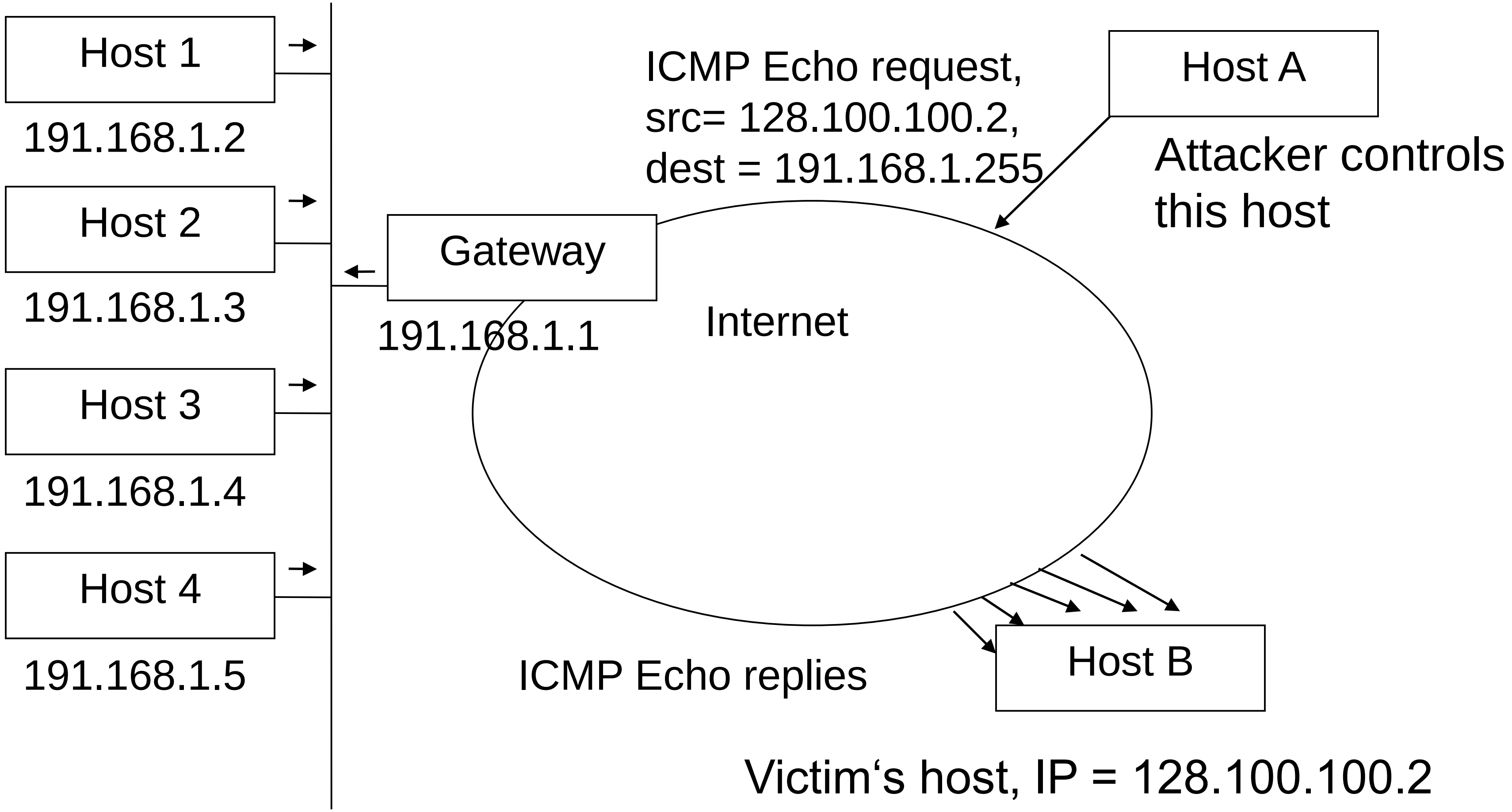
remember: in ICMP (based on IP) there are no ports

ICMP Echo Attacks

- map the hosts of a network
  - ICMP echo datagrams are sent to all the hosts in a subnet
  - attacker collects the replies and determines which hosts are alive

- denial of service attack (SMURF attack)
  - send spoofed (with victim‘s IP address) ICMP Echo Requests to subnets
  - victim will get ICMP Echo Replies from every machine

Smurf Attack



Host 1

191.168.1.2

Host 2

191.168.1.3

Host 3

191.168.1.4

Host 4

191.168.1.5

Gateway

191.168.1.1

ICMP Echo request,
src= 128.100.100.2,
dest = 191.168.1.255

Host A

Attacker controls this host

Internet

ICMP Echo replies

Host B

Victim's host, IP = 128.100.100.2

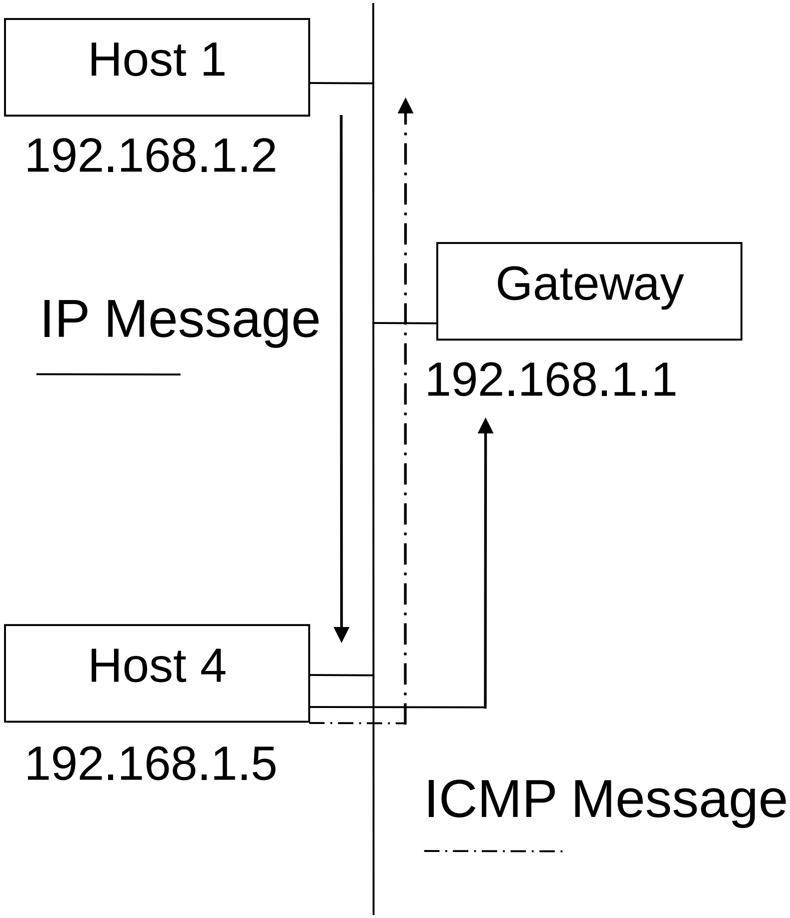
ICMP Redirect

- is used for stating that there is a better route to a host/net
- is sent by a router that routes a packet over the same interface that was used for receiving this packet

type (=5)	code	checksum (2 bytes)
IP address of the router that should be used		

IP header + first 8 bytes of the original datagram

ICMP Redirect - Example



1) In Host1's configuration, it is stated to use Host4 as a gateway. So when Host1 sends a packet outside the subnet, this is forwarded to Host4.

2) Host4 gets the packet, but has to forward the packet to Gateway.

3) Additionally Host4 sends Host1 an ICMP redirect message. "The net xxx can be reached better via Gateway yyy.

ICMP Redirect

- A host that receives an ICMP redirect message checks:
  - whether the new router is directly connected to the network
  - the redirect must be from the current router for this destination
  - the redirect can‘t tell the host to use itself as the router
  - the route that is being modified has to be an indirect route

- What is not checked
  - is message really sent by the current router?
  - is the target host (the new router) a router?

ICMP Redirect Attacks

- ICMP redirect messages can be used to re-route traffic on specific routes or to a specific host that is not a router at all
- The attack is very simple: just send a spoofed ICMP redirect message that appears to come from the host‘s default gateway
- Can be used to
  - Hijack traffic
  - Perform a denial of service attack

ICMP Dest. Unreachable

- ICMP message used by gateways to state that the datagram cannot be delivered
- Many subtypes
  - Network unreachable
  - Host unreachable
  - Protocol unreachable
  - Port unreachable
  - Fragmentation needed but don‘t fragment bit set
  - Destination host unknown
  - Destination network unknown etc.

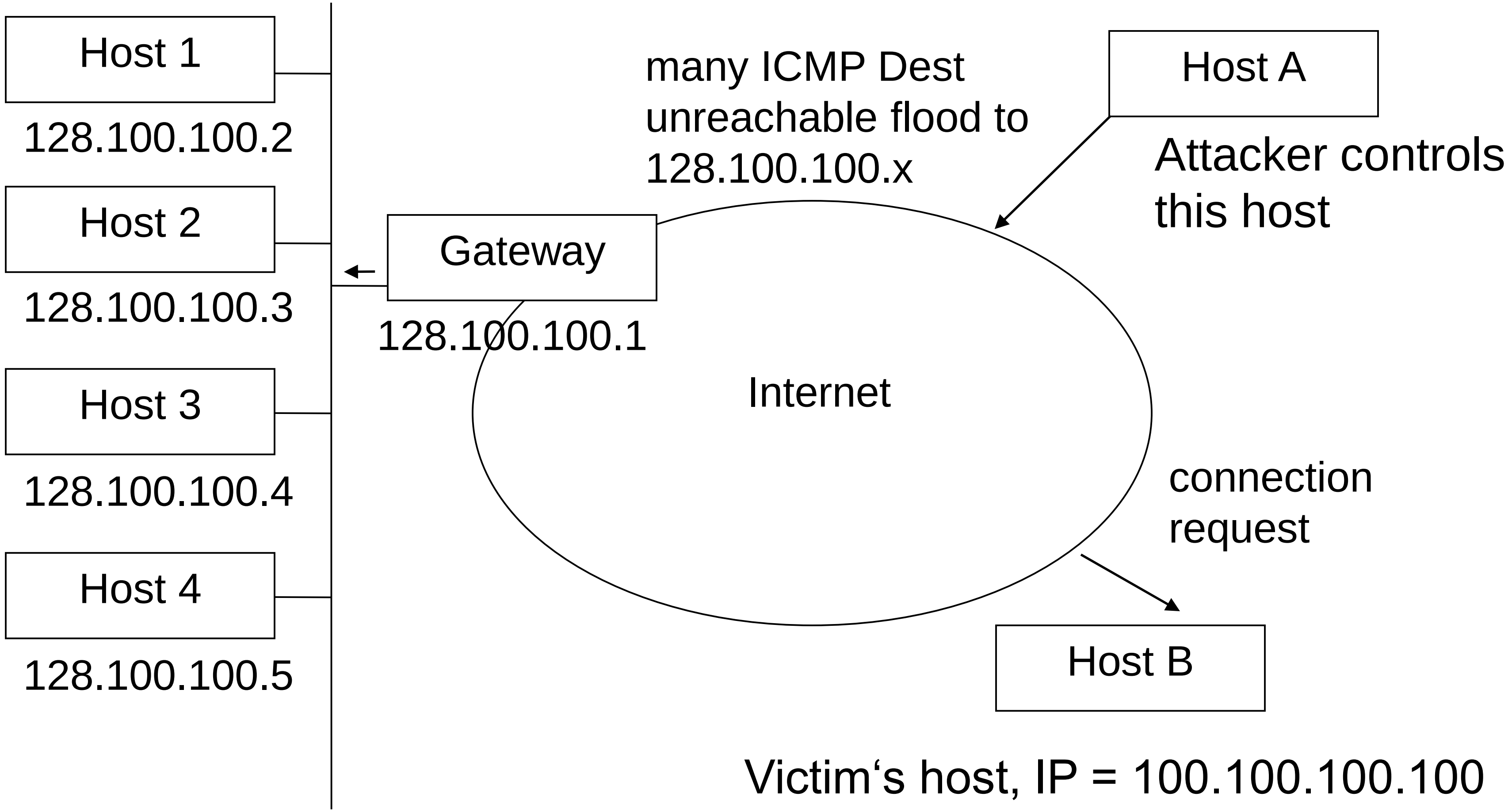
Dest. Unreachable Attack

- Can be used to “cut” out nodes from the network
- is a denial of service attack (DOS)

Example

An attacker injects many forged destination unreachable messages stating that 100.100.100.100 is unreachable into a subnet (e.g. 128.100.100.). If 128.100.100.2 net tries to connect to 100.100.100.100, he will immediately get an ICMP dest. unreachable from the attacker‘s host. For 128.100.100.2, this means that there is no way to contact 100.100.100.100, and therefore communication fails.

Dest. Unreachable Attack



Victim‘s host, IP = 100.100.100.100

ICMP Time Exceeded

Used when

- TTL becomes zero (code =0)

- The reassembling of a fragmented datagram times out (code=1)

type (=11)	code (0 or 1)	checksum (2 bytes)
unused (4 bytes)		
IP header + first 8 bytes of the original datagram		

Traceroute

- Program to determine the path to a specific host/net by evaluating ICMP Time Exceeded messages
- Does this by
  - sending a series of IP datagrams to the destination node
  - each datagram has an increasing TTL field (start=1)
  - gets back ICMP Time Exceeded messages by the intermediate gateways
  - so the full path can be reconstructed by Traceroute
- Traceroute also allows to use loose source routing
- Useful tool for topology mapping

Layer 4 Protocols

Many protocols use IP as the underlying network layer

- Important ones are

  ICMP (Internet Control Message Protocol)

  UDP (User Datagram Protocol)

  TCP (Transmission Control Protocol)

User Datagram Protocol

UDP (User Datagram Protocol)

- relies on IP
- connectionless
- unreliable (checksum optional)
- best-effort
- datagram delivery service

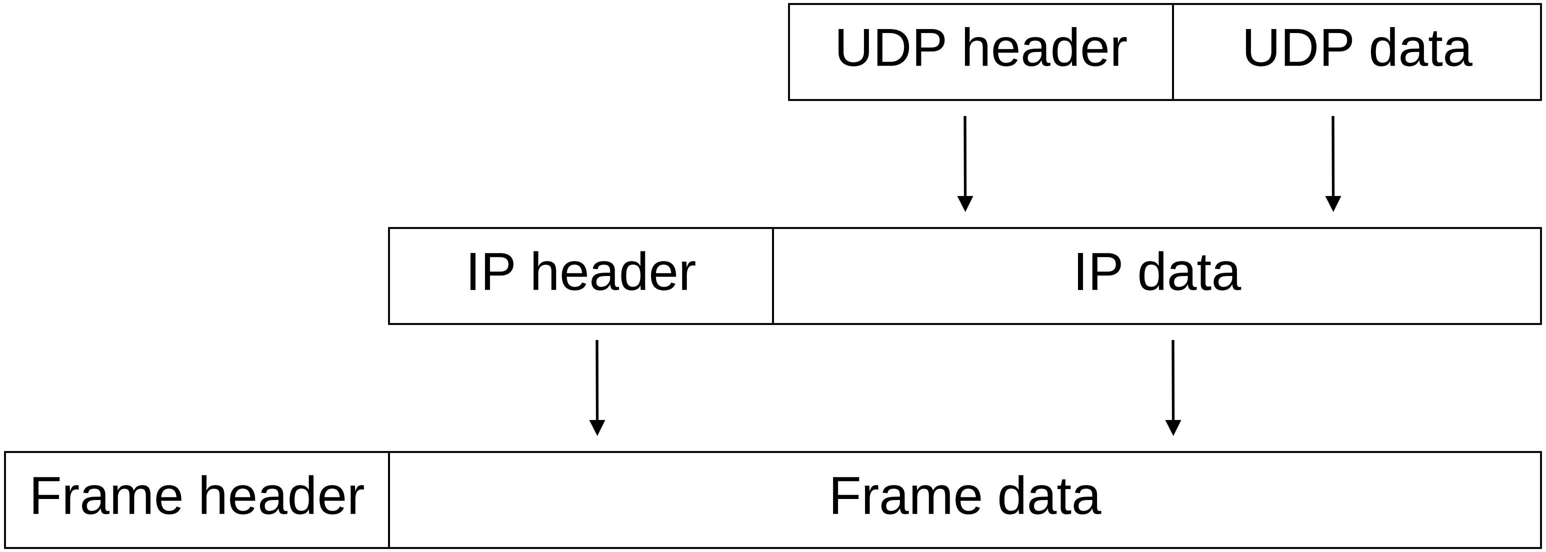
➢ delivery, integrity, non-duplication and ordering are not guaranteed

UDP Message

- Port abstraction
  - allows addressing different destinations for the same IP
- Often used for multimedia
  - and for services based on request/reply schema (DNS, RPC, NFS)
  - more efficient than TCP

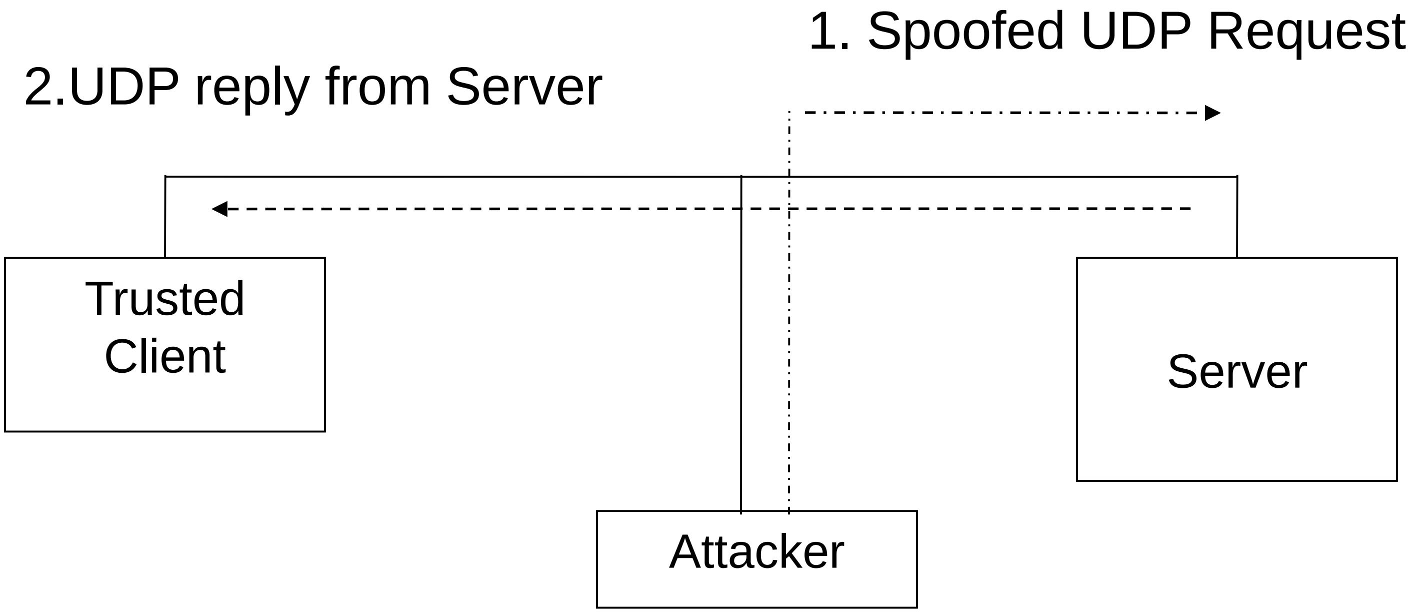
UDP source port (2 bytes)	UDP destination port (2)
UDP message length (2)	Checksum (2)
Data	

UDP Encapsulation



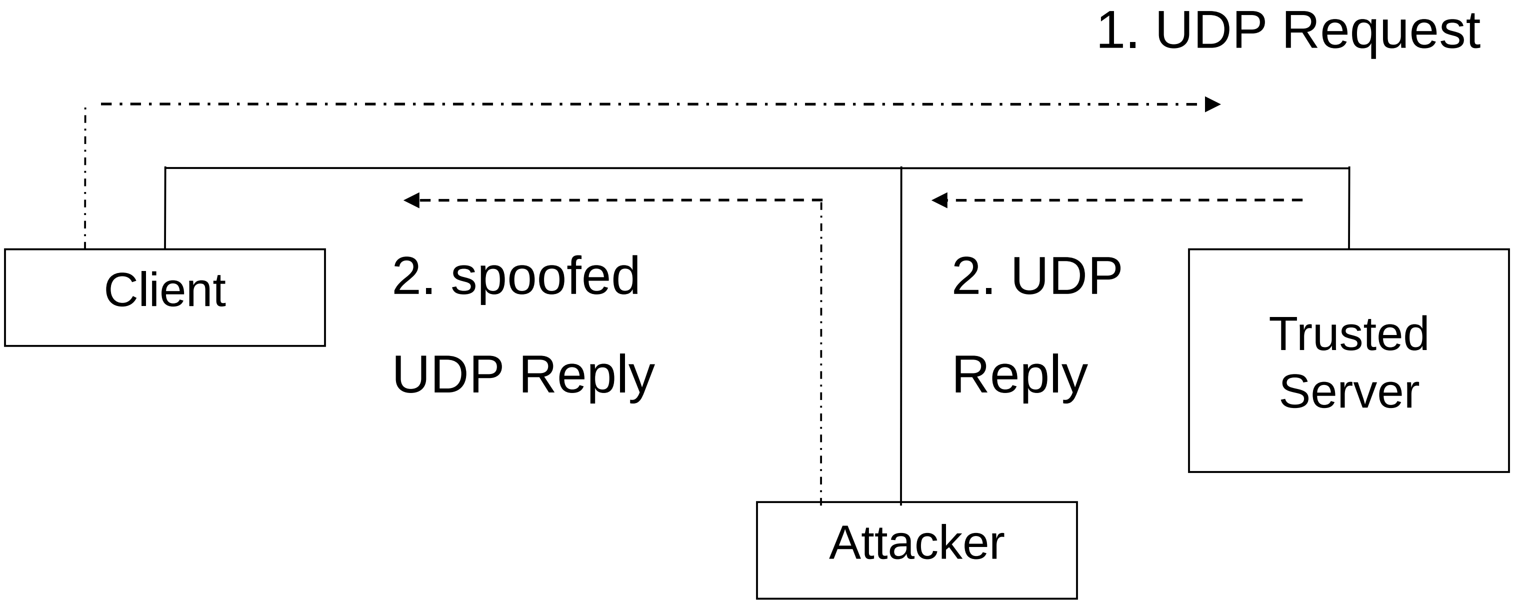
UDP Spoofing

- Basically IP spoofing, as easy to perform



UDP Hijacking

- Variation of the UDP spoofing attack
- Racing against the legitimate server



UDP Portscan

- Used to determine which UDP services are available
- Zero-length UDP packet is sent to each port
- If an ICMP error message “port unreachable“ is received, the service is assumed to be unavailable
- Many TCP/IP stack implementations implement a limit on the error message rate, therefore this type of scan can be slow (e.g. Linux limit is 80 messages every 4 seconds)

UDP Portscan

How to do a UDP portscan?

- by hand (with packet filter and RAW-socket)
- use netcat (http://netcat.sourceforge.net/) and tcpdump
- or use e.g. nmap -sU <address> (http://www.insecure.org/nmap/)

TCP

TCP (Transmission Control Protocol)

- relies on IP to provide
- connection-oriented
- reliable
- stream delivery service

➢ no loss, no duplication, no transmission errors, correct data ordering

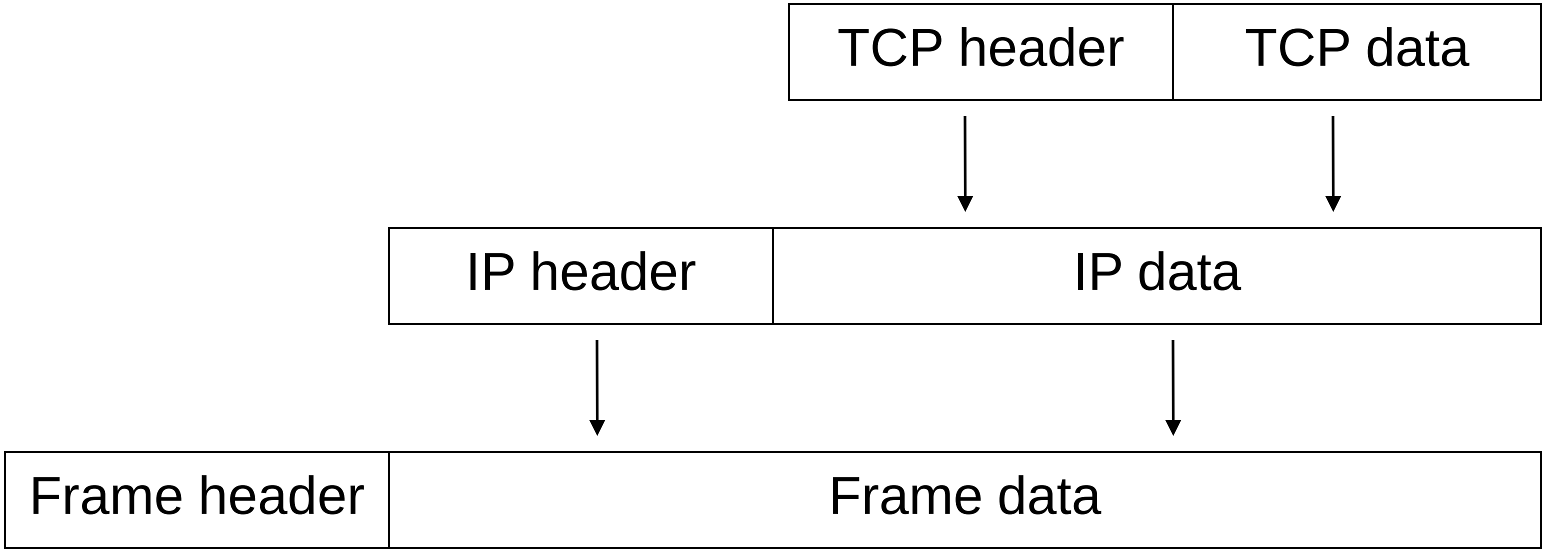
TCP

- Provides (like UDP) the port abstraction

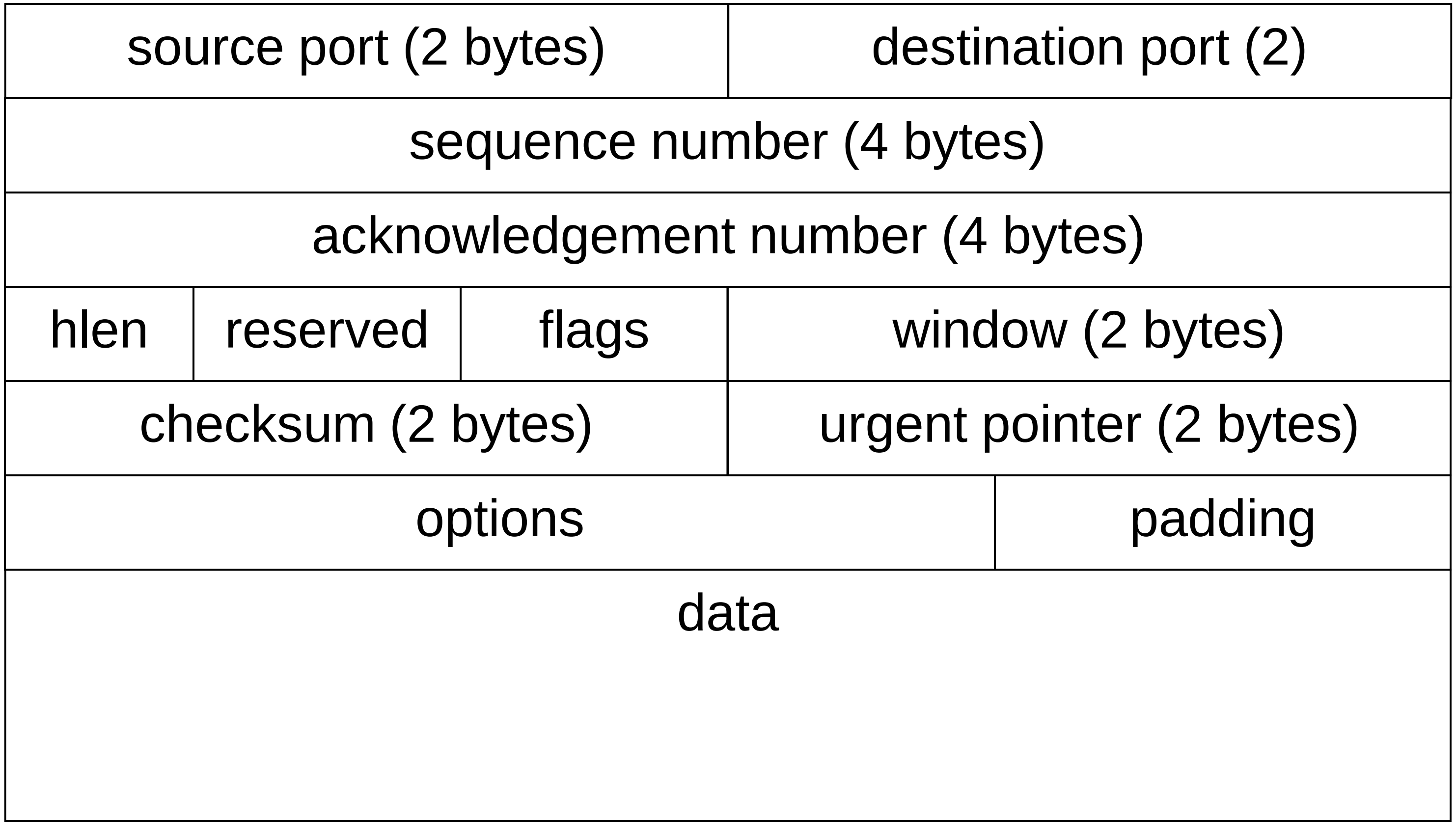
- Allows two nodes to establish a virtual circuit
  - identified with quadruples
  - <src_ip, src_port, dst_ip, dst_port>
  - virtual circuit is composed of two streams (full duplex)

- The pair <IP address, port> is called a *socket*

TCP Encapsulation



TCP Segment



<table>
<tr><td colspan="3">source port (2 bytes)</td><td colspan="2">destination port (2)</td></tr>
<tr><td colspan="5">sequence number (4 bytes)</td></tr>
<tr><td colspan="5">acknowledgement number (4 bytes)</td></tr>
<tr><td>hlen</td><td>reserved</td><td>flags</td><td colspan="2">window (2 bytes)</td></tr>
<tr><td colspan="3">checksum (2 bytes)</td><td colspan="2">urgent pointer (2 bytes)</td></tr>
<tr><td colspan="4">options</td><td>padding</td></tr>
<tr><td colspan="5">data</td></tr>
</table>

TCP Seq/Ack Numbers

- Sequence number (seq)
  - specifies the position of the segment data in the communication stream
  - seq = 1234 means:

  The payload of this segment contains data starting from 1234

- Acknowledgement number (ack)
  - specifies the position of the *next expected byte* from the communication partner
  - ack = 12345 means:

  I have received the bytes correctly to 12344, I expect the next byte to be 12345

- Both are used to manage error control
  - retransmission, duplicate filtering

TCP Window

- Used to perform flow control

- Segment will be accepted only if the sequence number has a value between
  - last ack number sent and
  - last ack number sent + window size

- The window size changes dynamically to adjust the amount of information that can be sent by the sender
  - set by the receiver to announce how much it can take
  - window size = amount of data the client can handle now

TCP Flags

- Flags are used to manage the establishment and shutdown of a virtual circuit

- SYN: request for synchronization of seq/ack numbers (used during connection setup)

- ACK: the acknowledgement number is valid (all segments in a virtual circuit have this flag set, except the first)

- FIN: request to shutdown a virtual circuit

- RST: request to immediately reset the virtual circuit

- URG: states that the urgent pointer is valid

- PSH request a “push“ operation on the stream (pass the data to the application (interactive) as soon as possible)

TCP Security

- Scanning
- OS Fingerprinting
- TCP Spoofing
- TCP Hijacking
- Denial of Service
  - SYN flooding
  - Process Table Attack
  - Land Attack

TCP Scanning

- Used to check whether a port is open on a host
  - `/etc/services` lists standard port/service mappings
- Should be done without letting monitored host know that it is scanned (OS/tools can log connections)
- Used to get some extra information about the host

In the simplest form a TCP connection is opened to a port

- if this succeeds a service is assumed to be available

TCP SYN Scanning

- Also known as “half open“ scanning
- The attacker sends a SYN packet (packet with SYN flag)
  - If the server answers with a SYN/ACK packet, then the port is open (or with a RST packet: the port is closed)
- The attacker sends a RST packet instead of an ACK

➢ Therefore the connection is never opened and the event is not logged by the operating system / monitor application

TCP FIN Scanning

- The attacker sends a FIN-marked packet
- In most TCP/IP implementations (not Windows)
  - if the port is closed, a RST packet is sent back
  - if the port is open, the FIN packet is ignored
- Variations of this type of scanning technique
  - XMAS Scan: FIN + PSH + URG set
  - NULL Scan: no flags set

OS Fingerprinting

- OS Fingerprinting
  - allows to determine the operating system of a host by examining the reaction to carefully crafted packets
  - use of reserved flags in the TCP header
  - use of weird combination of flags in the TCP header
  - check the selection of TCP initial sequence numbers
  - analysis of response to particular ICMP messages
  - server response at a special port (Login)

NMAP

- Is a tool for performing portscans and for OS fingerprinting
- http://www.insecure.org/nmap/
- supports
  - IP scans
  - UDP portscans
  - TCP portscans (SYN, FIN scanning)
  - OS fingerprinting

TCP Spoofing

Attack aimed at impersonating another host

mostly during the TCP connection establishment phase

- Node A trusts node B (e.g. login with no password)
- Node C wants to impersonate B with respect to A in opening a TCP connection
- C kills B (flooding, redirecting, crashing)
- C sends A a TCP segment in a spoofed IP packet with B‘s address as the source IP and an initial sequence number T

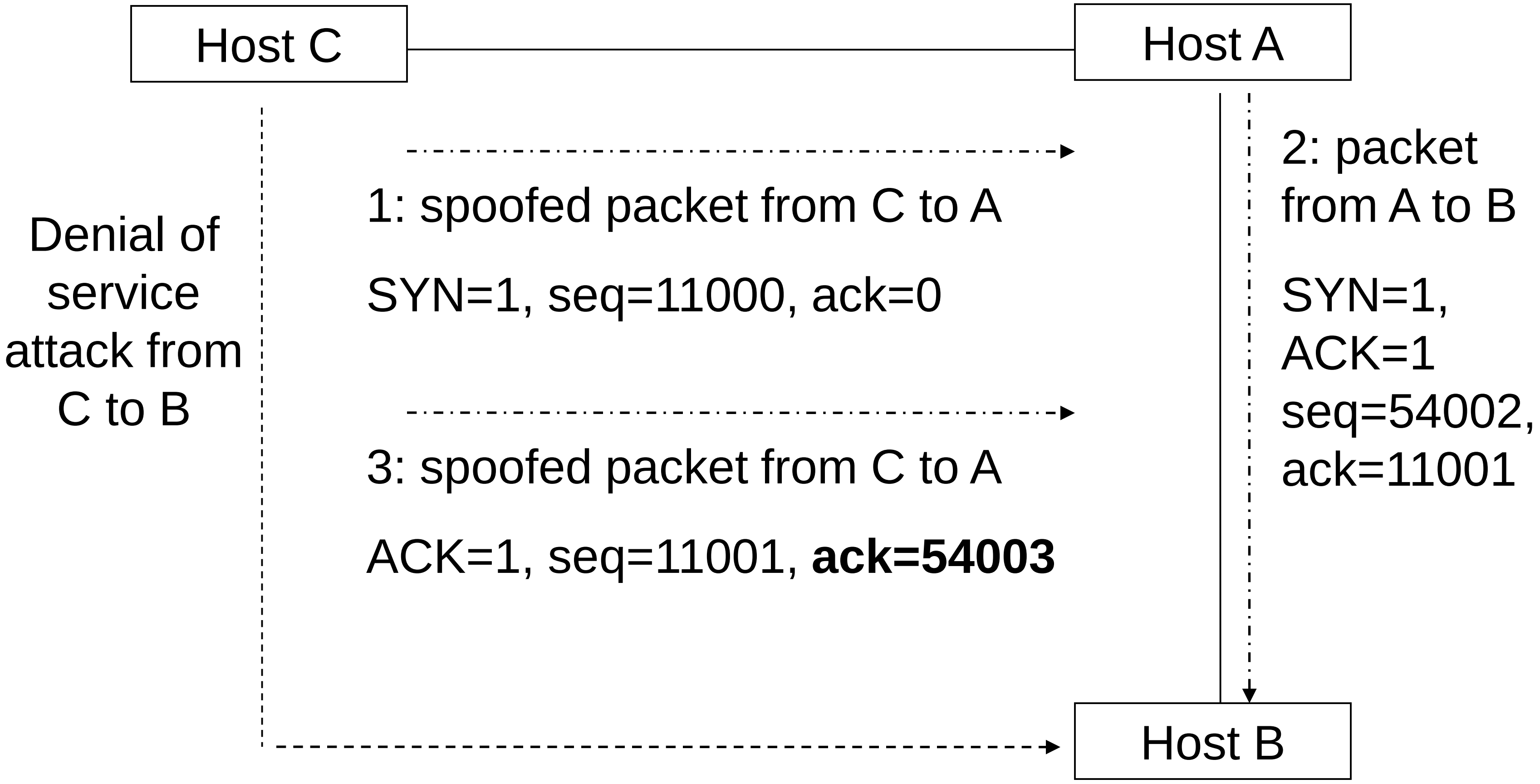
TCP Spoofing

- A replies with a TCP SYN/ACK segment to B with S as the sequence number and T+1 as acknowledge number
- C does not receive the segment from A to B, but in order to finish the handshake it has to send an ACK segment with S+1 as the acknowledge number to A

for this two possibilities exist:

- C eavesdrops the SYN/ACK segment and calculates the number
- C guesses the correct sequence number

TCP Spoofing



Host C

Host A

Host B

Denial of service attack from C to B

1: spoofed packet from C to A

SYN=1, seq=11000, ack=0

2: packet from A to B

SYN=1, ACK=1 seq=54002, ack=11001

3: spoofed packet from C to A

ACK=1, seq=11001, **ack=54003**

TCP Hijacking

- Technique to take control of an existing TCP connection

- The attacker uses spoofed TCP segments to
  - insert data into the streams
  - reset existing connections & reopen them

- But the correct sequence/acknowledgement numbers must be used (guessed or eavesdropped by the attacker)

DOS TCP Attacks

SYN flooding attack (known as Neptune attack)

- very common denial-of-service attack
- attacker starts handshake with SYN marked segment
- victim replies with SYN-ACK segment
- attacker‘s host stays silent
- a host can only keep a limited number of TCP connections in half-open state. After that limit, connections are not accepted.

- Current solution
  - drop half open connections in FIFO manner
  - SYN cookies

DOS TCP Attacks

Process Table Attack

- Daemons are programs that listen on a particular port for connection requests
- When a new connection is established the daemon
  - forks a new process that will handle the connection
  - waits for the next connection

Many daemons run with root privileges (no restrictions)

A huge number of connections fill up the process table and

no new processes can be created

DOS TCP Attacks

Land Attack

- A TCP segment with the SYN flag set is sent to an open port
- The source address and port are the same as the destination address/port
- The host starts an „internal“ ACK storm, which is very CPU intensive
  - tries to open connections to a port that is already in use
  - sequence numbers (a new one is generated when the SYN segment arrives) don‘t match which results in an ACK-storm