Special Topics in Security
ECE 5698

Engin Kirda
ek@ccs.neu.edu

Northeastern University

Admin News

- Next week, I need to travel
  - On Tuesday, I am planning to post a recorded lecture
  - No physical class
- Next week on Friday
  - There is no class
- The week after that, second Challenge goes online

ARP

ARP (Address Resolution Protocol)

- Service at the link-level, RFC 826
- maps network-addresses to link-level addresses
- Host A wants to know the hardware address associated with IP address of host B
- A broadcasts ARP message on physical link
  - including its own mapping
- B answers A with ARP answer message
- Mappings are cached: arp -a shows mapping

RARP

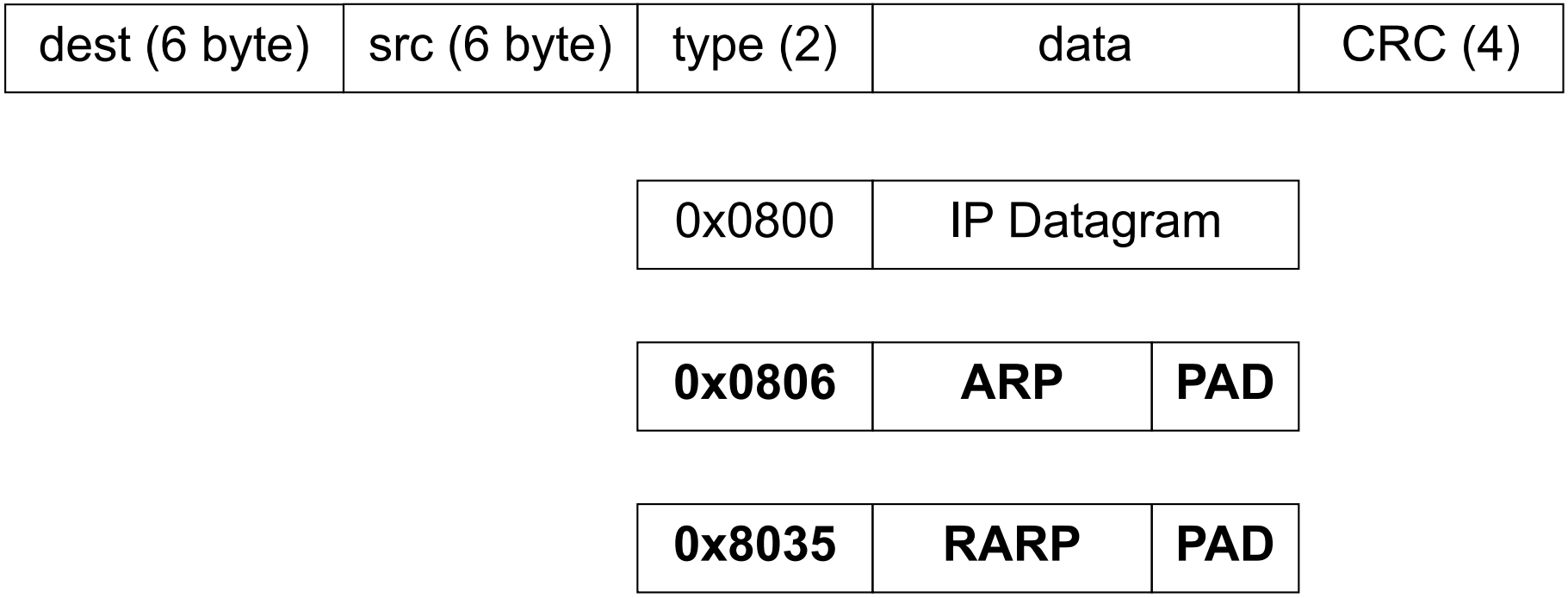
RARP (Reverse Address Resolution Protocol)

- maps link-level addresses to network-addresses
- for diskless stations to obtain their own IP address
- Service at the link-level, RFC 903

Host A wants to know its IP address (which is IP_A)

- A broadcasts RARP message on physical link
- RARP server answers with RARP answer
  - containing IP_A

(R)ARP Message



(R)ARP Message

hardware type (2 byte)		protocol type (2 byte)
hw.adr.size (1 byte)	prot. adr. size (1 byte)	opcode (2 byte)
sender Ethernet address (6 byte)		
sender IP address (4 byte)		
target Ethernet address (6 byte)		
target IP address (4 byte)		

(R)ARP Message

- use same message format
- contain:
  - types and address sizes of hardware and protocol
  - type of message (=opcode, (R)ARP request/reply)
  - link-level and network level addresses of sender and target.
- depending on type, different fields are empty
  - ARP: target link level address
  - RARP: everything except source link-level address

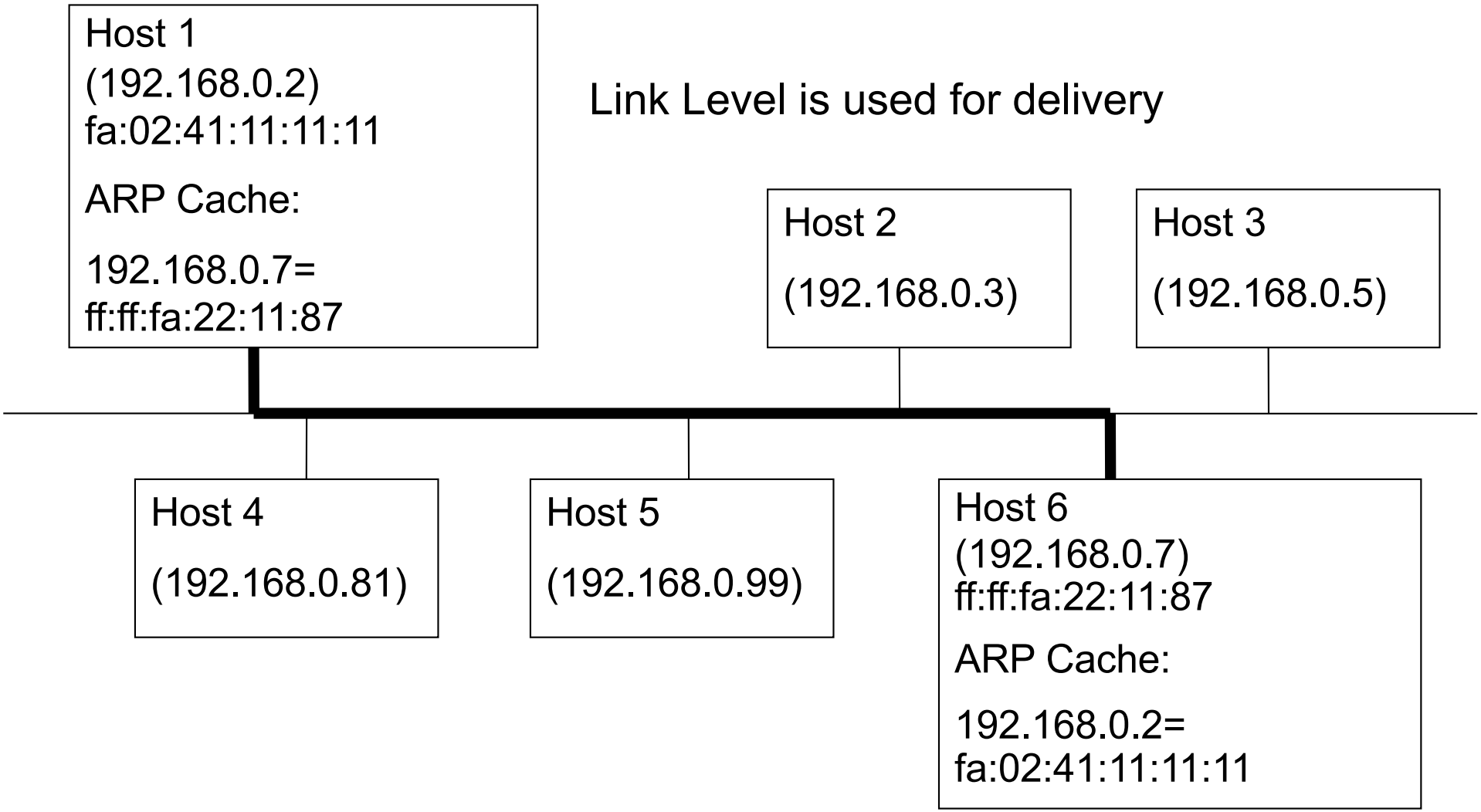
Sending an IP Packet

- Assume host A wants to send an IP packet to host B and that all ARP caches are empty

Procedure

- A sends ARP request for IP-B.
- B sends ARP answer to A
- ARP caches on A+B are filled
- A sends encapsulated IP datagram on link level to B
- datagram is delivered

Direct IP Delivery



LAN Attacks

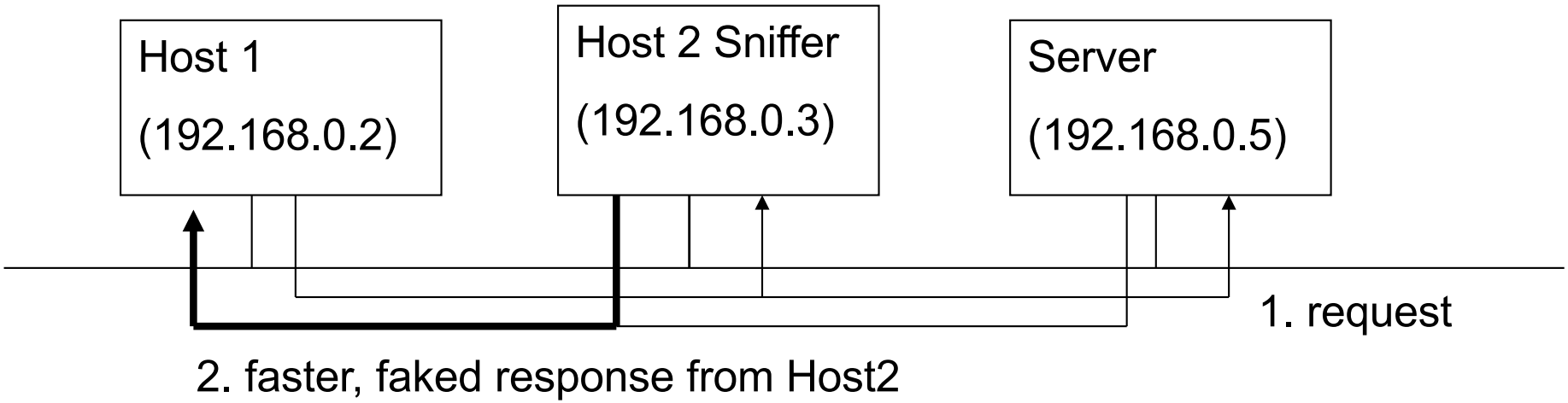
- Goals:
  - Information Recovery
  - Impersonate Host
  - Tamper with delivery mechanisms

- Methods:
  - Sniffing
  - IP Spoofing
  - ARP attacks

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

ARP Attack 1/2

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 the attacker

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

ARP Attack 2/2

- 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)

- Tools:
- e.g. WinARP: denial of service against Windows
  - requires the victim to click on many modal dialogs
  - or reboot the machine (1 dialog / ARP packet)

Some Tools

everyone should know (do `man` <toolname> on UNIX/Linux)

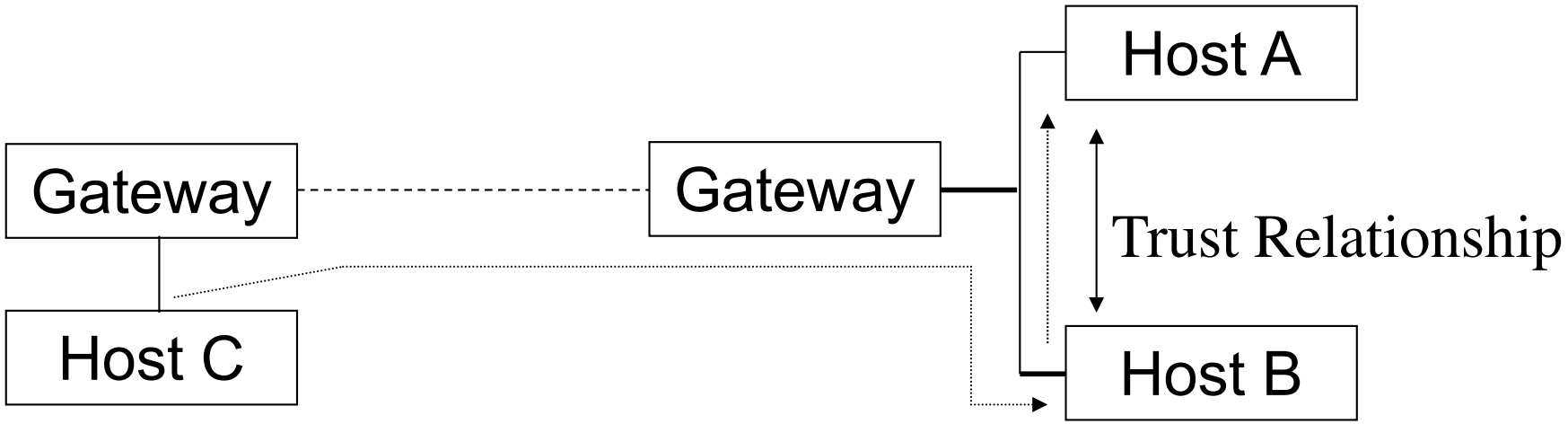
- arp
  - service program for the ARP service
- ping
  - check whether a host is alive
- tcpdump
  - check what is going on on the net down to the packet level
- nslookup (dig / host)
  - DNS resolving

Attacks involving Multiple Networks

- Blind IP spoofing
- Man-in-the-middle-attacks
- Attacks concerning the routing mechanism

Blind IP Spoofing

- usually the attacker does not have access to the reply, abuse trust relationship between hosts
  - e.g.
    - Host C sends an IP datagram with the address of some other host (Host A) as the source address to Host B
    - attacked host (B) replies to the legitimate host (A)

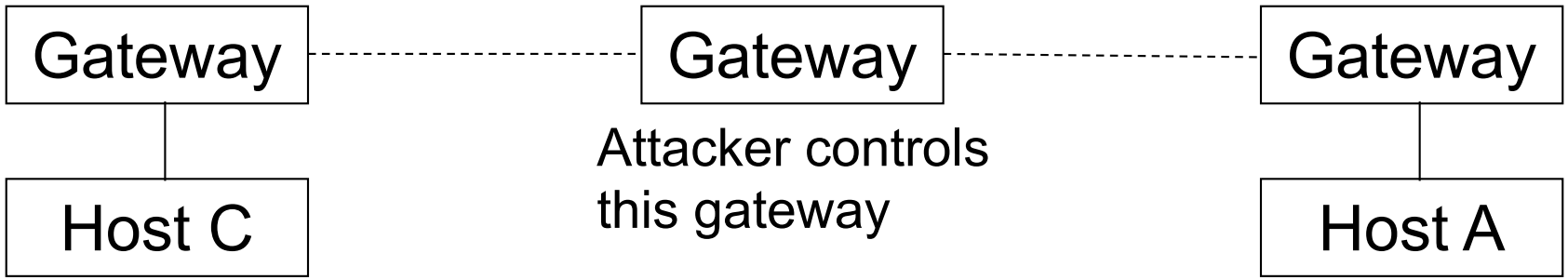


Man-in-the-Middle Attack

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



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)

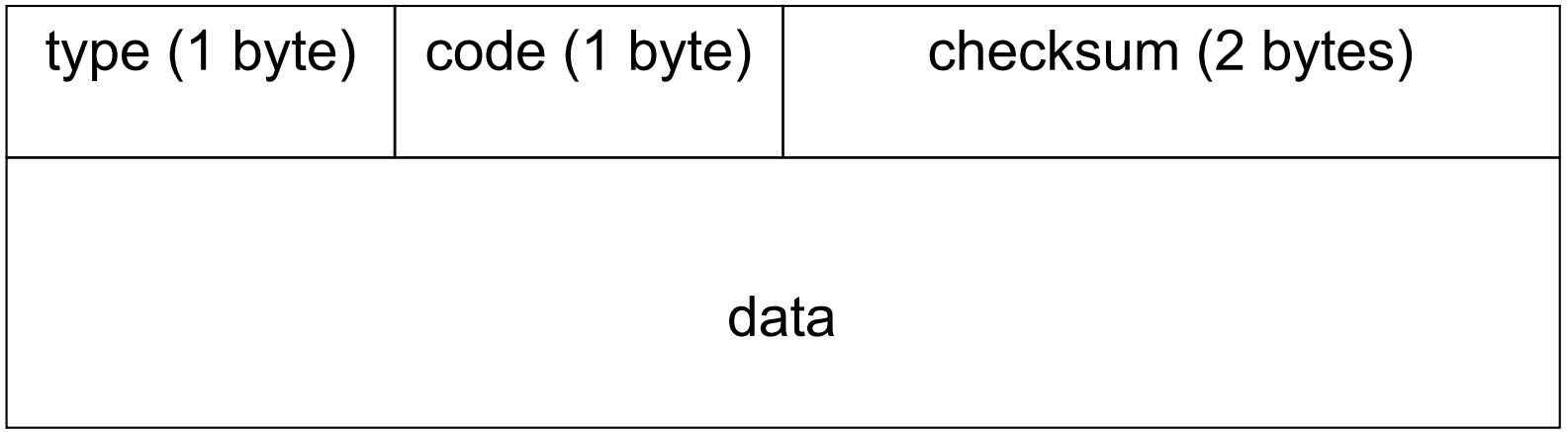
ICMP

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 field: specifies the class of the ICMP message

code field: specifies the exact type of the message

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 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 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		

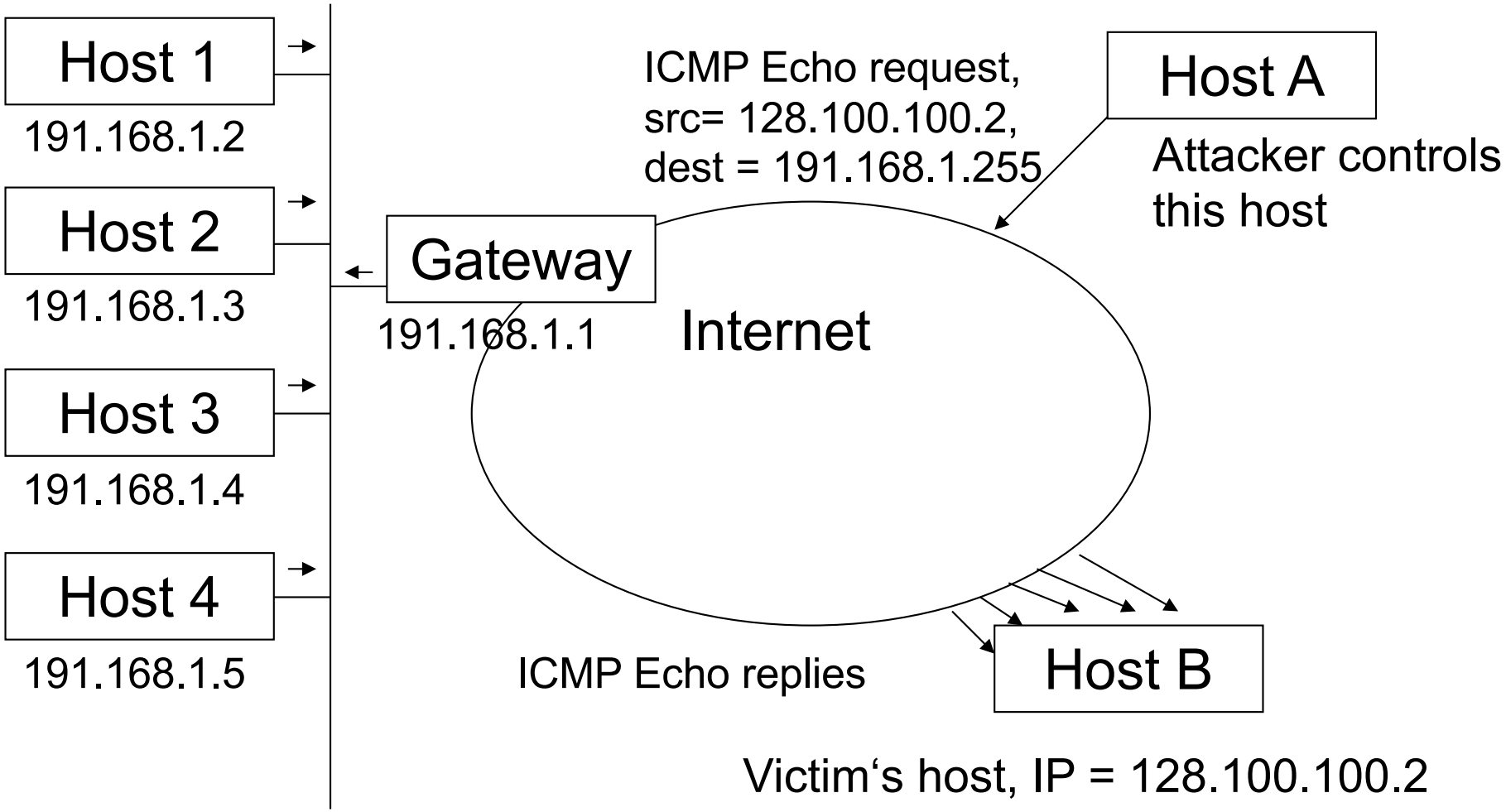
identifier is used by „ping“ to deliver back the packet to the right process (allowing more than one ping to run concurrently)

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

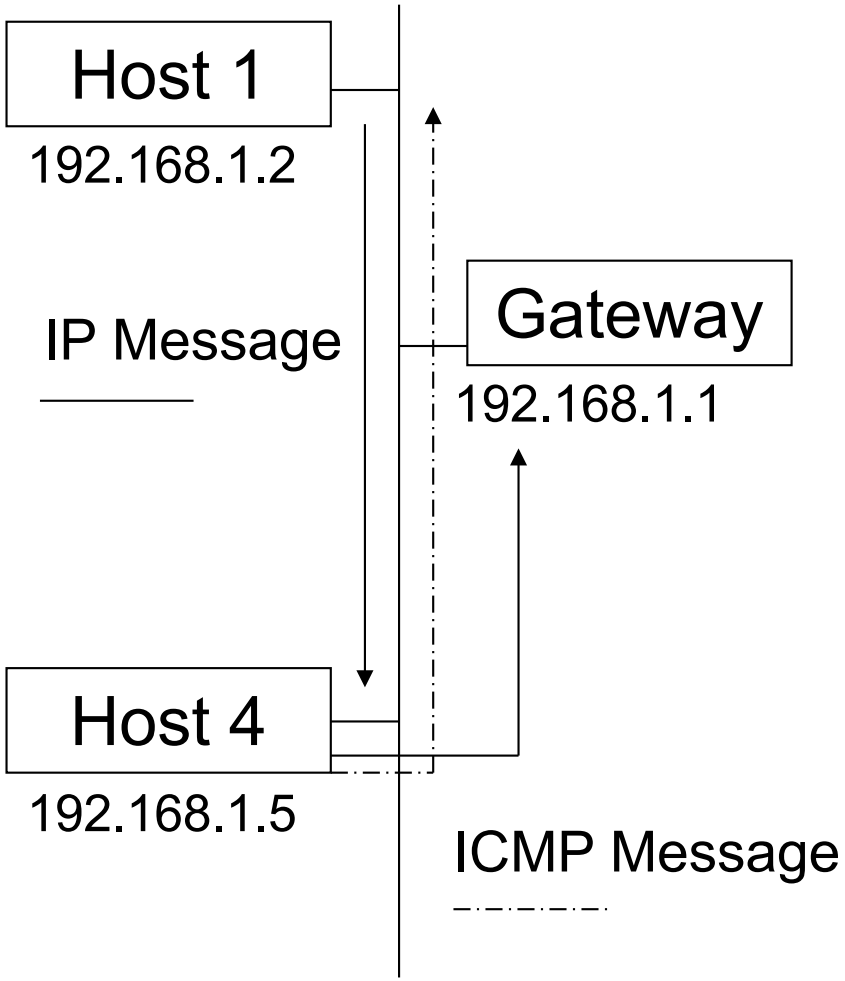


ICMP Redirect

- 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.

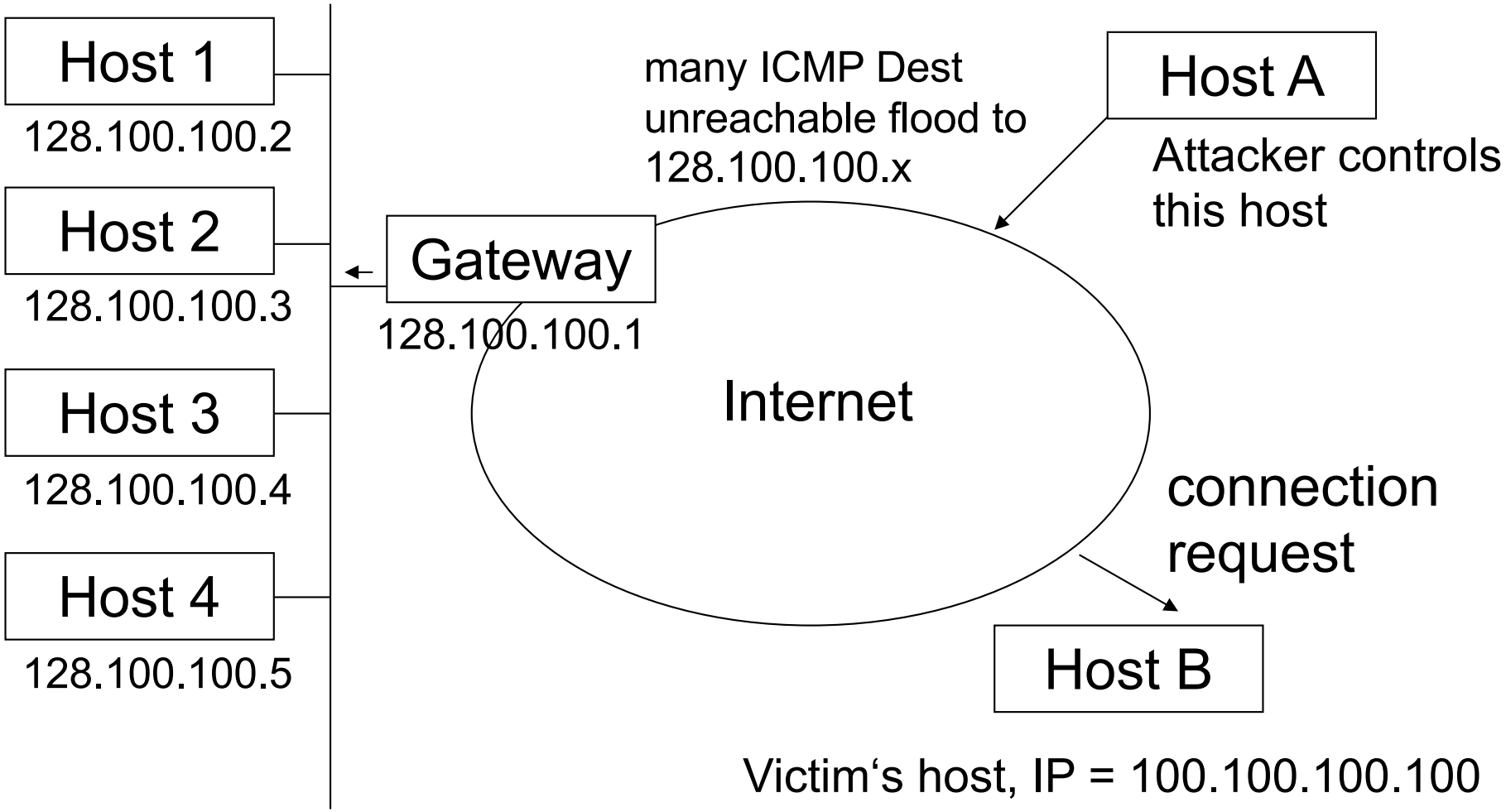
Dest. Unreachable Attack

- 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 Time Exceeded 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



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

User Datagram Protocol

UDP (User Datagram Protocol)

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

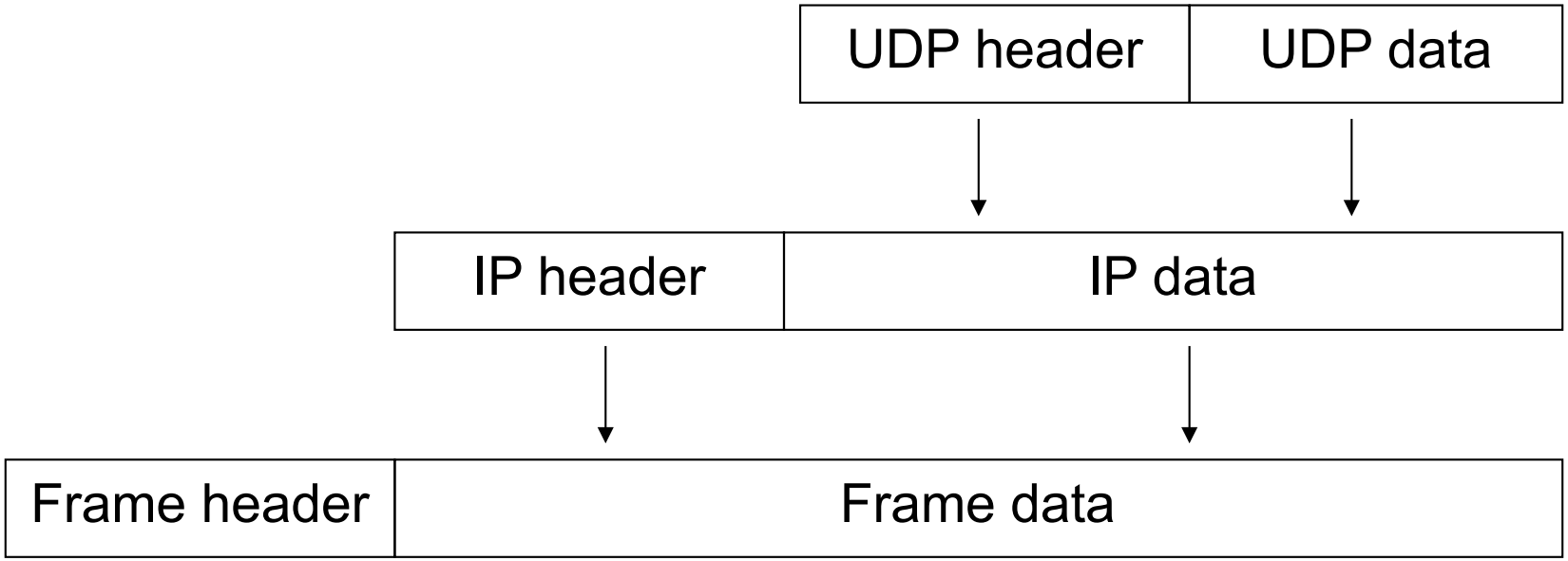
➢ delivery, integrity, non-duplication and ordering are not guaranteed

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

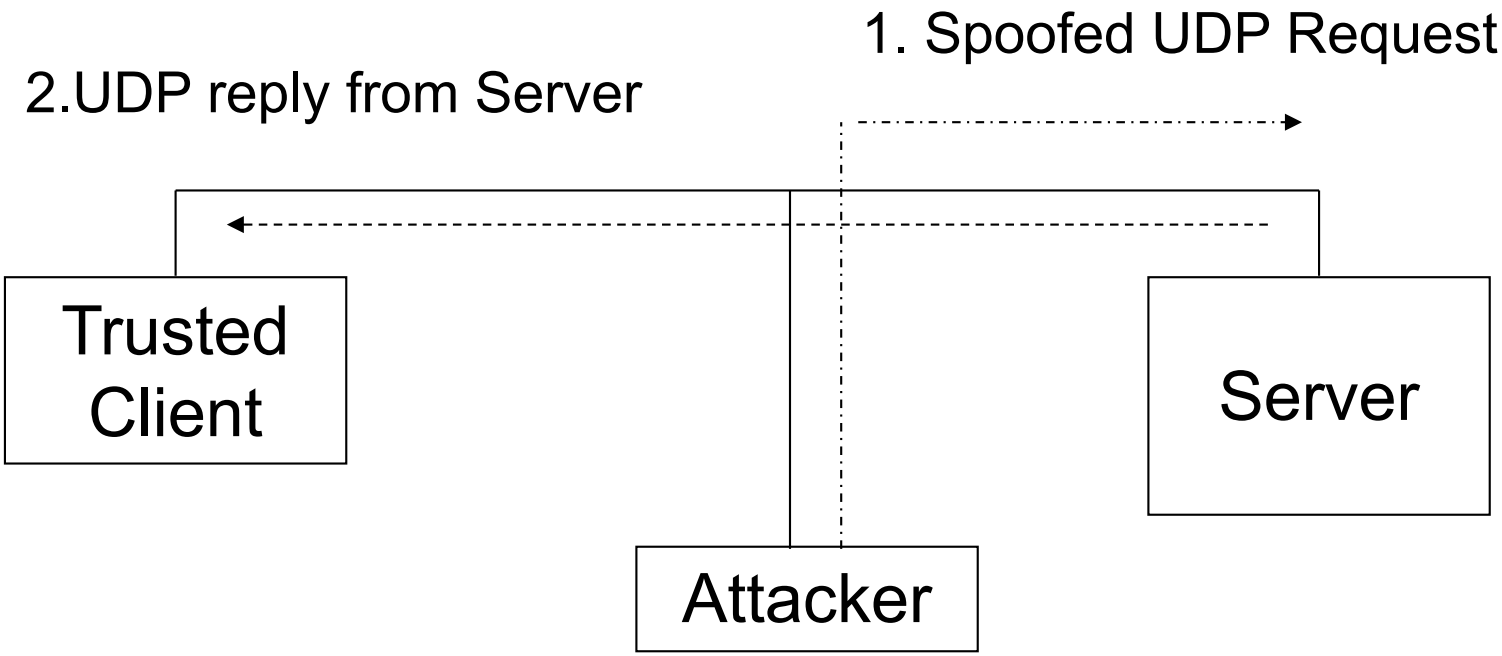
UDP source port (2 bytes)	UDP destination port (2)
UDP message length (2)	Checksum (2)
Data	

UDP Encapsulation



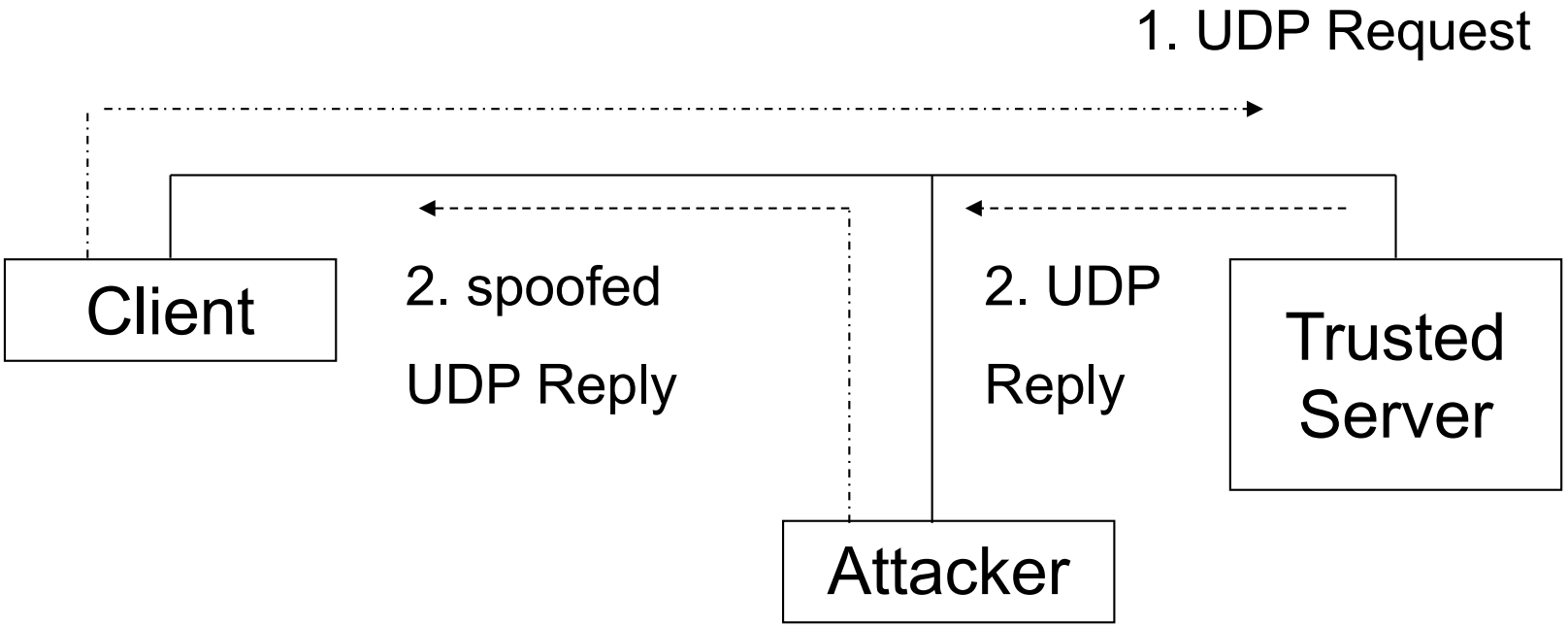
UDP Spoofing

- Basically IP spoofing, as easy to perform



UDP Hijacking

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



UDP Storms

- Spoofed UDP datagram is sent to the echo service (port 7)
- Source port is set to the chargen device (port 19)
- Reply of the echo service is interpreted as a request by the chargen service
- Reply of the chargen service is interpreted as a request by the echo service
- ... etc …
- Same attack can be carried out using two echo services

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)

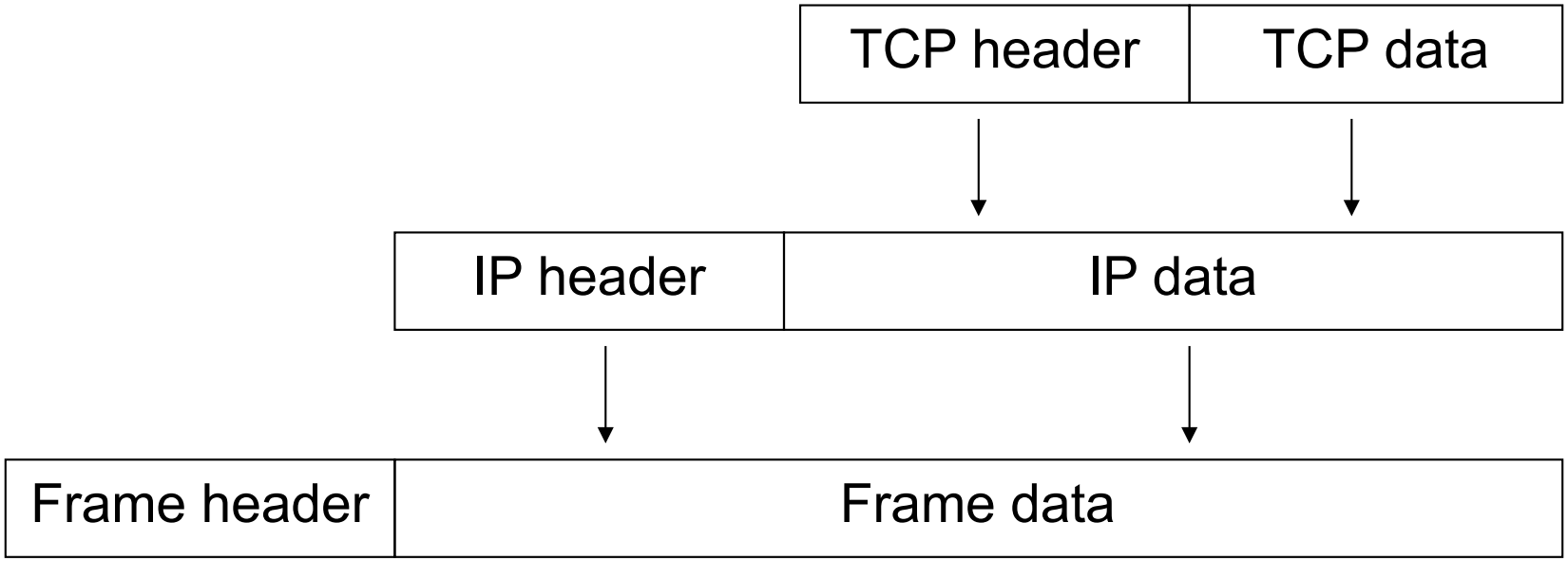
- relies on IP to provide
- connection-oriented
- reliable
- stream delivery service

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

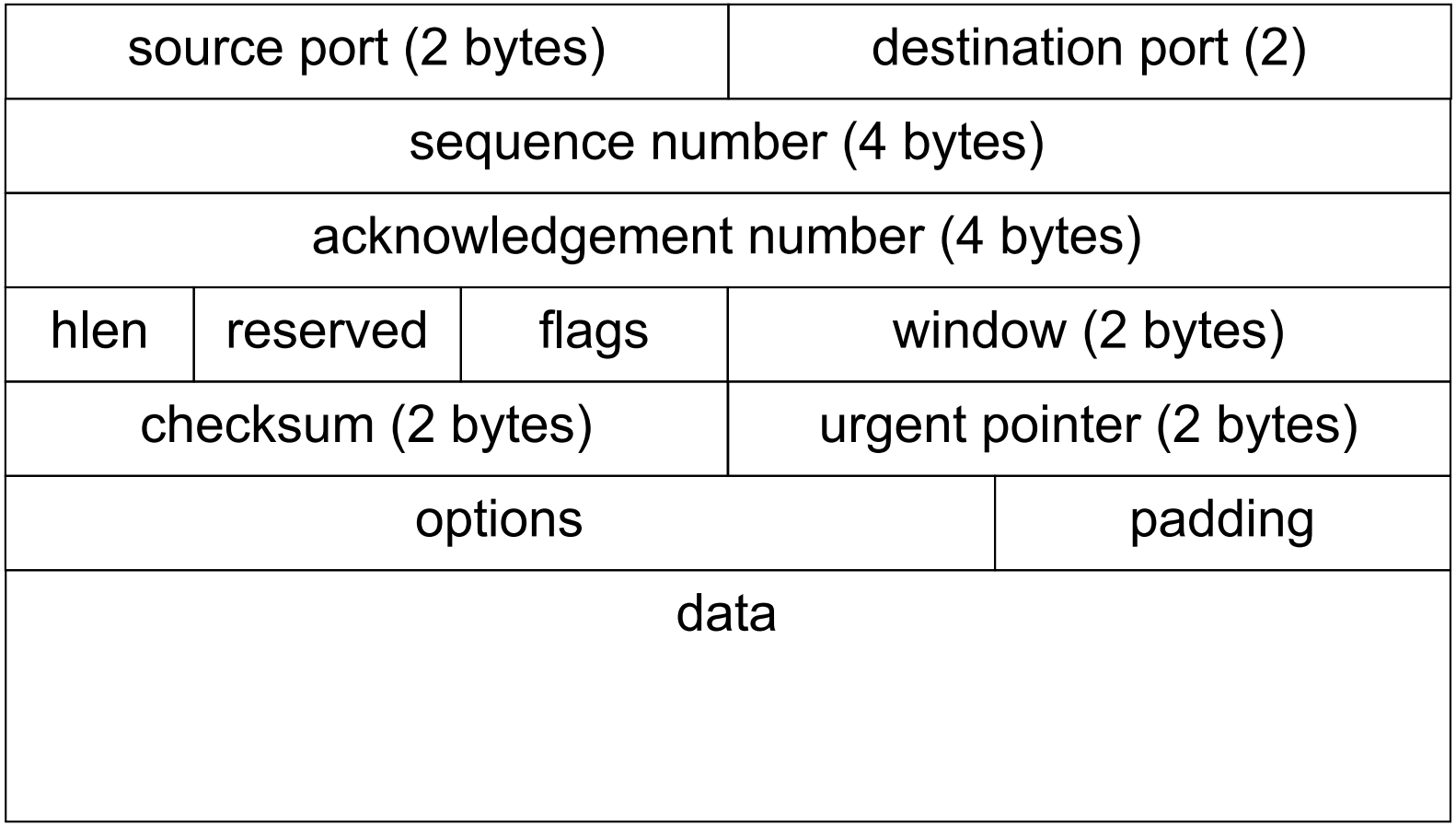
TCP

- 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


source port (2 bytes)
destination port (2)
sequence number (4 bytes)
acknowledgement number (4 bytes)
hlen
reserved
flags
window (2 bytes)
checksum (2 bytes)
urgent pointer (2 bytes)
options
padding
data

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 Options

Most important

- Maximum Segment Size (MSS)
  - the size of the largest packet a partner is able to receive
  - set during setup phase
- Window scale factor
  - allows to specify a larger window of TCP segments to accept
- Timestamp
  - for TCP-Echo requests + responses (similar to ICMP)

TCP Virtual Circuit :: Setup

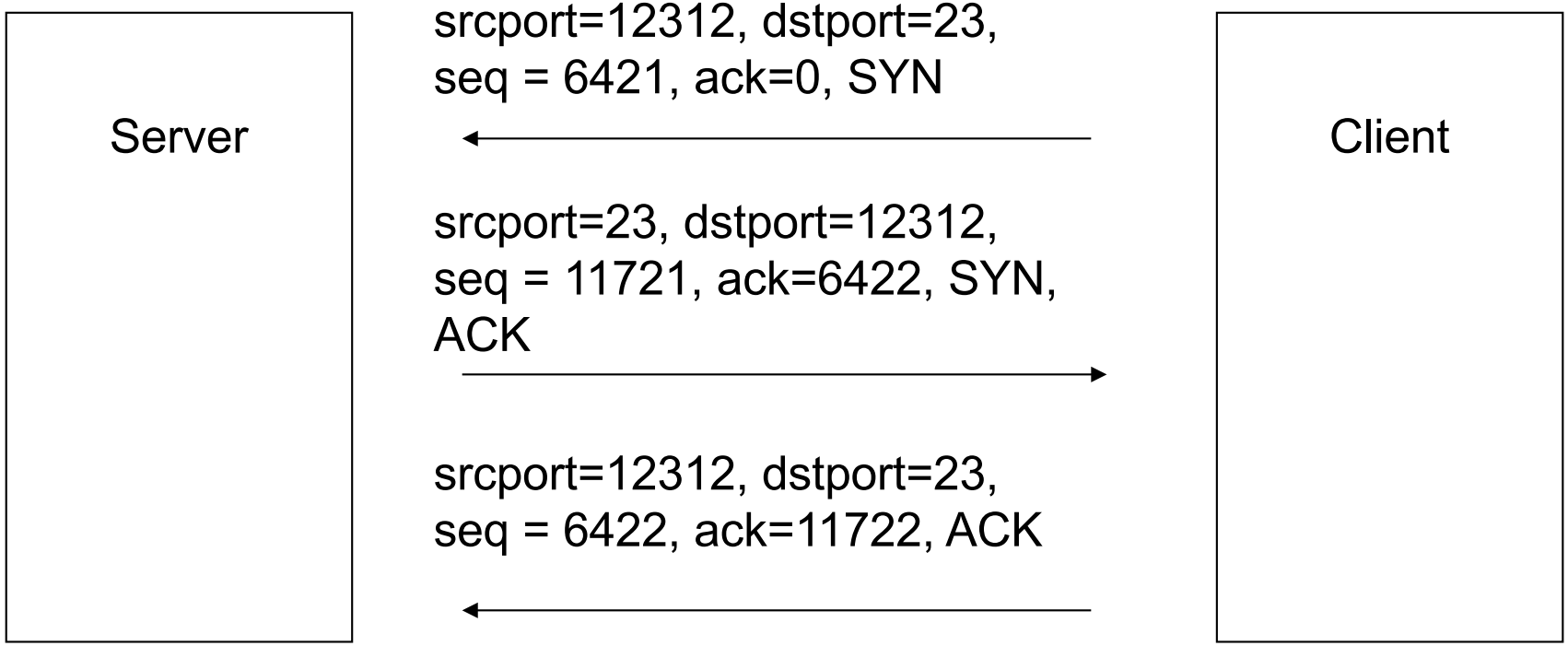
- A server listens to a specific port
- Client sends a connection request to the server, with SYN flag set and a random initial sequence number c
- The server answers with a segment marked with both the SYN and ACK flags and containing
  - an initial random sequence number s
  - c+1 as the acknowledge number
- The client sends a segment with the ACK flag set and with sequence number c+1 and ack number s+1

Admin News

- Tuesday, there is no class, Friday there is a Quiz, and a recorded class

Three Way Handshake

- Three way because 3 TCP segments are necessary to set up a virtual circuit



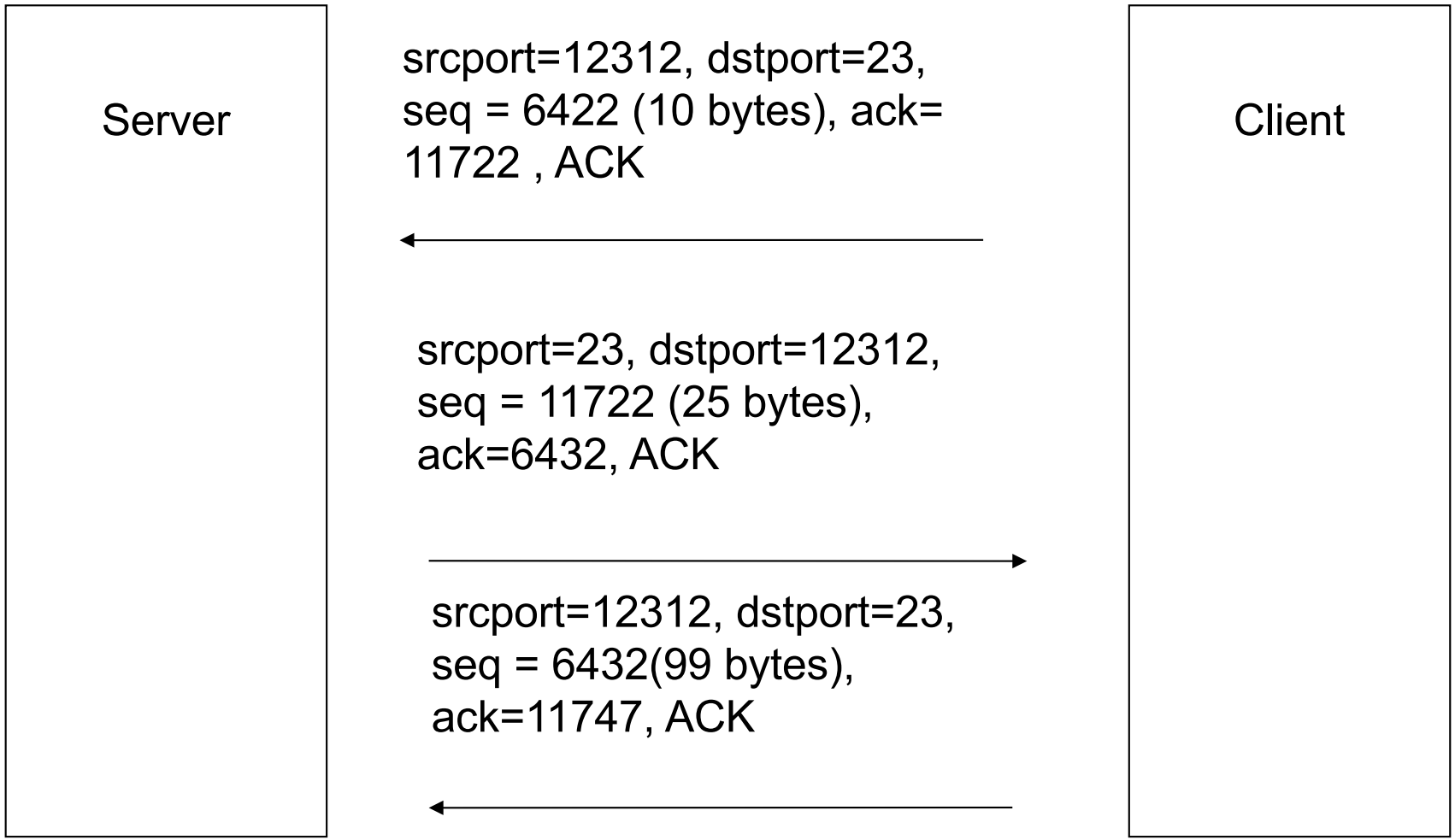
Initial Sequence Number

- The TCP standard (RFC 793) specifies that the sequence number should be incremented every 4 μs
- BSD UNIX systems initially used a number that is incremented by 64000 every half second and by 64000 each time a connection is established
- This number is important, because it adds protection against simple attacks.

TCP Data Exchange

- Each TCP segment contains
  - sequence nr = position of data in stream (often, last ack number)
  - ack nr = sequence number of last correctly received segment increased by the payload size of this segment
- A partner accepts a segment of the other partner only if the numbers are inside the transmission window
- An empty segment may be used to acknowledge the received data
- Packets with no payload and SYN or FIN flag consume one sequence number

TCP Data Exchange



Virtual Circuit :: Shutdown

- One of the partners, e.g. A, can terminate its stream
  - by sending a segment with the FIN flag set
- B answers with a segment with the ACK flag set
- From this point on A will not send any data to B: it will just acknowledge data sent by B
  - with empty segments
- When B shuts its stream down, the virtual circuit is considered closed

Sample TCP Connection

From	To	S	A	F	Seq-Nr	Ack-Nr	Payload
192.168.0.1	192.168.0.2	1	0	0	4711	0	0
192.168.0.2	192.168.0.1	1	1	0	38001	4712	0
192.168.0.1	192.168.0.2	0	1	0	4712	38002	0
192.168.0.2	192.168.0.1	0	1	0	38002	4712	‚Login:\n‘ 7
192.168.0.1	192.168.0.2	0	1	0	4712	38009	‚s‘ 1
192.168.0.1	192.168.0.2	0	1	0	4713	38009	‚e‘ 1
192.168.0.1	192.168.0.2	0	1	0	4714	38009	‚c‘ 1
192.168.0.1	192.168.0.2	0	1	0	4715	38009	‚\n‘ 1
192.168.0.2	192.168.0.1	0	1	0	38009	4716	0
192.168.0.1	192.168.0.2	0	0	1	4716	38009	0
192.168.0.2	192.168.0.1	0	1	0	38009	4717	0

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