UNIVERSITY OF TRENTO - Italy

eit Digital MASTER SCHOOL

Security Engineering

Lecture 16 – Network Security

Fabio Massacci

(with the courtesy of W. Stallings)

Lecture Outline

- ***Network Attacks***
  - Attive Attacks
  - Passive Attacks
  - TCP Attacks
- ***Contermeasures***
  - IPSec
  - SSL/TLS
  - Firewalls
  - Intrusion Detection Systems
  - Honeypots

Network Security

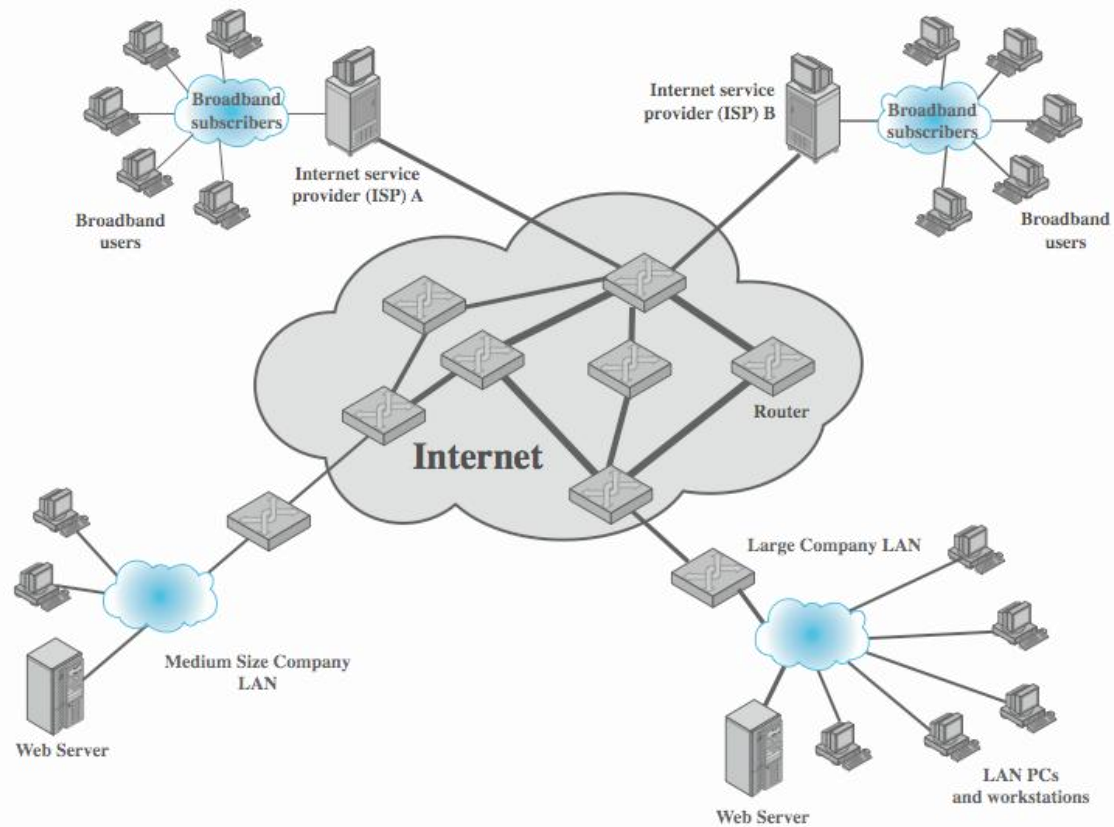
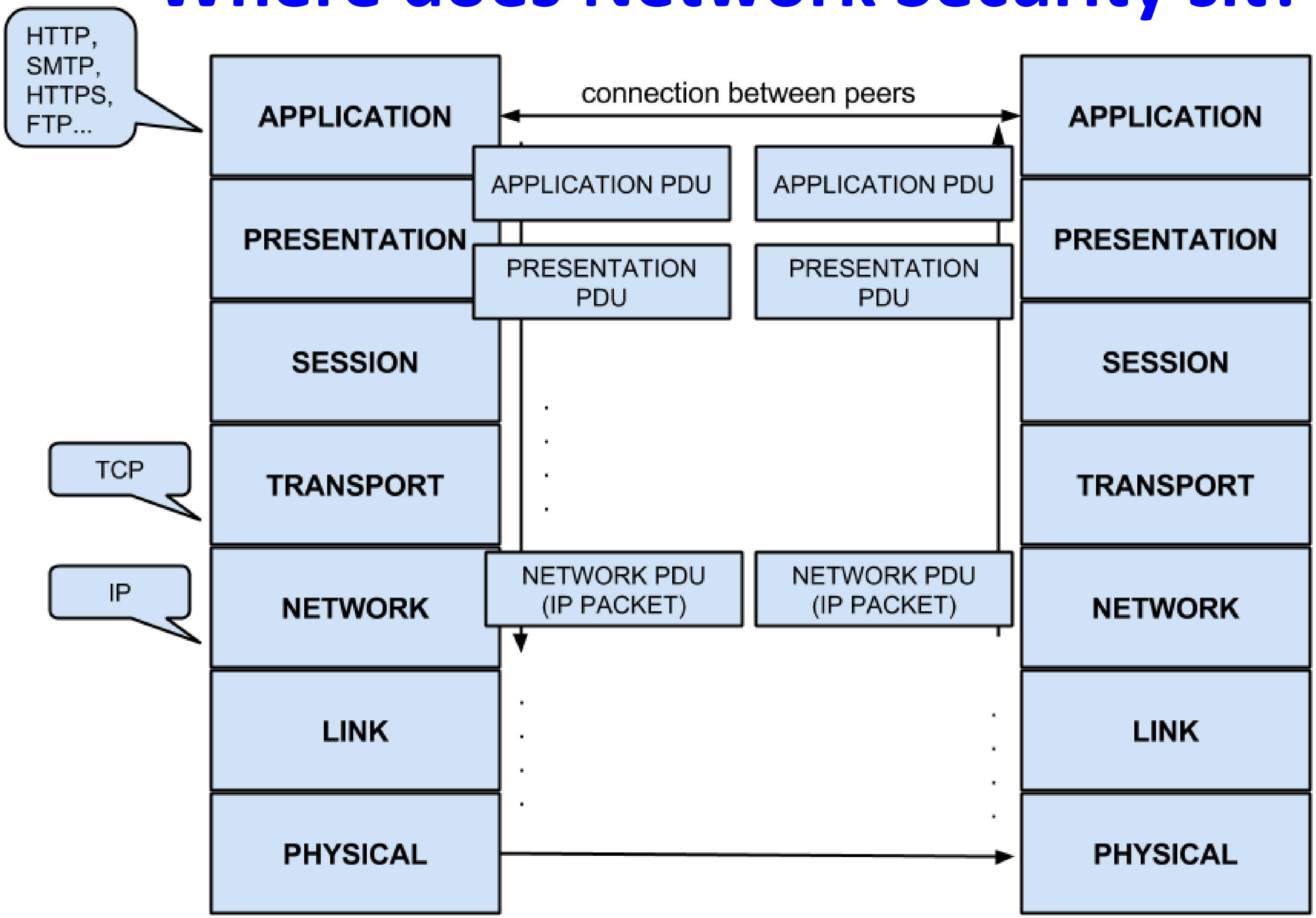


Figure 8.1 Example Network to Illustrate DoS Attacks

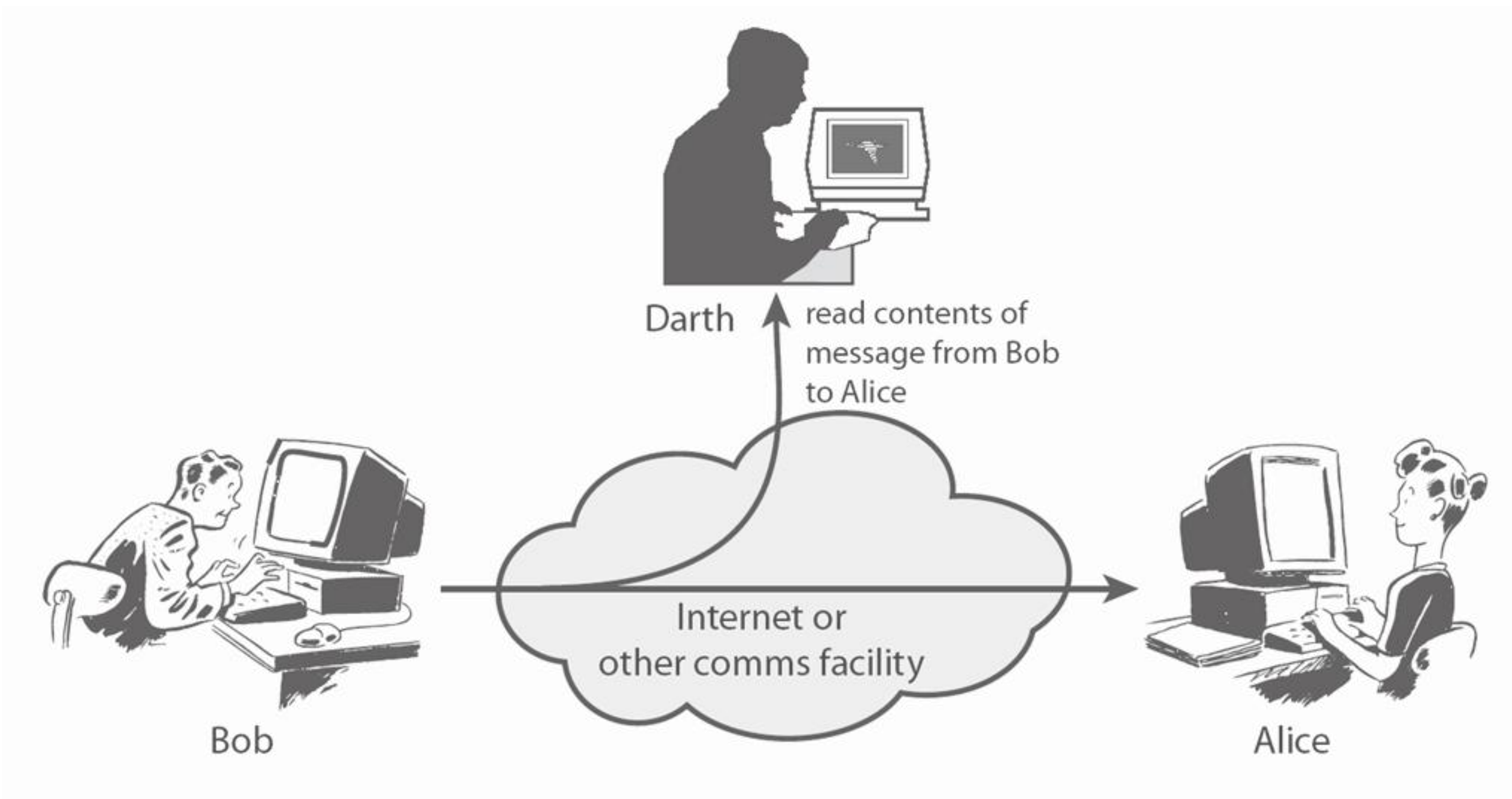
Where does Network Security sit?



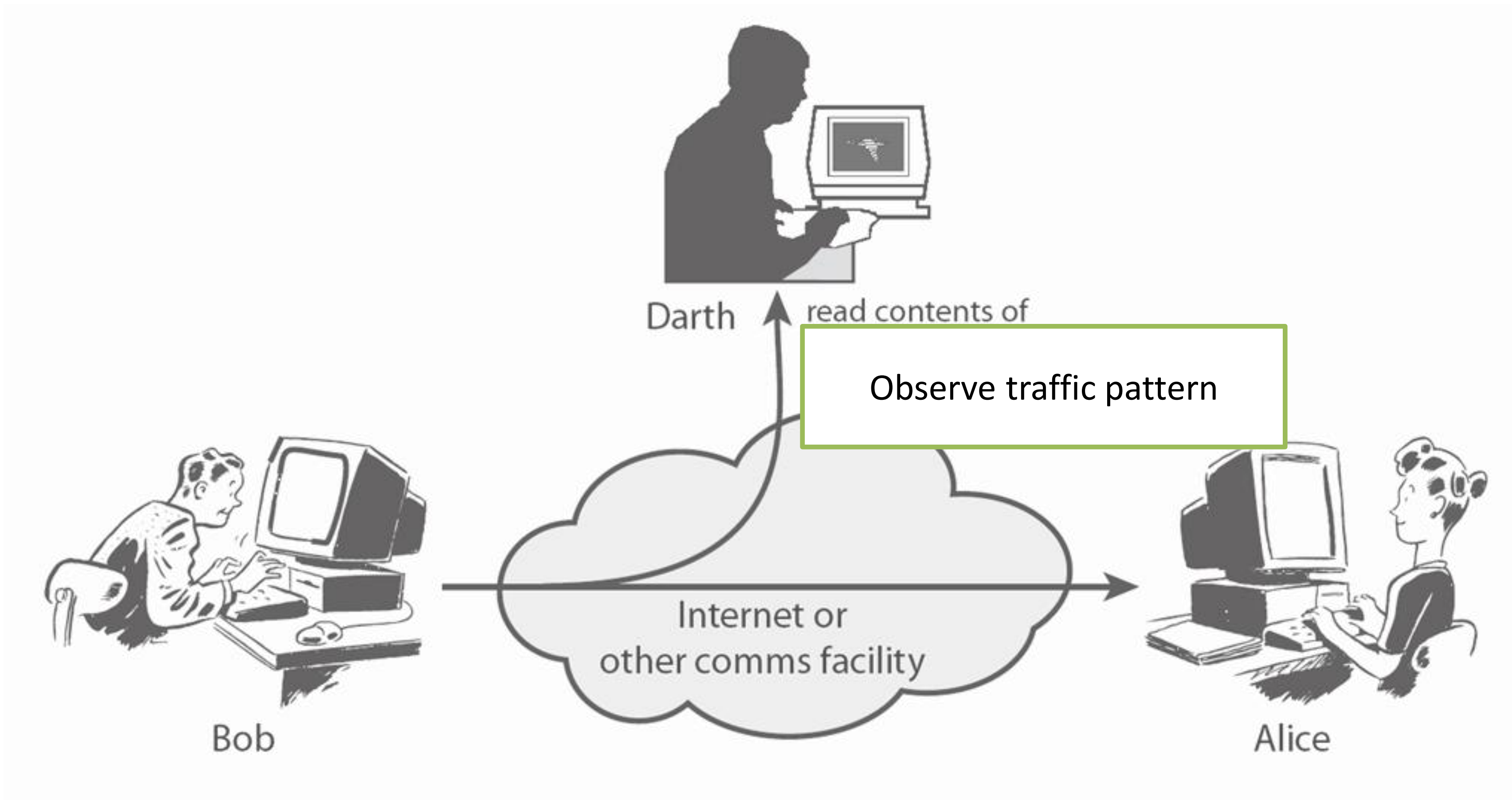
Passive and Active attacks

- ***Passive attacks***
  - **GOAL :** obtain information
  - No modification of content or fabrication
    - Release of message contents
    - Traffic analysis
- ***Active attacks***
  - **GOAL :** modification of content and/or participation in communication to
    - Impersonate legitimate parties (Masquerade)
    - Replay or retransmit
    - Modify the content in transit
    - Launch denial of service attacks

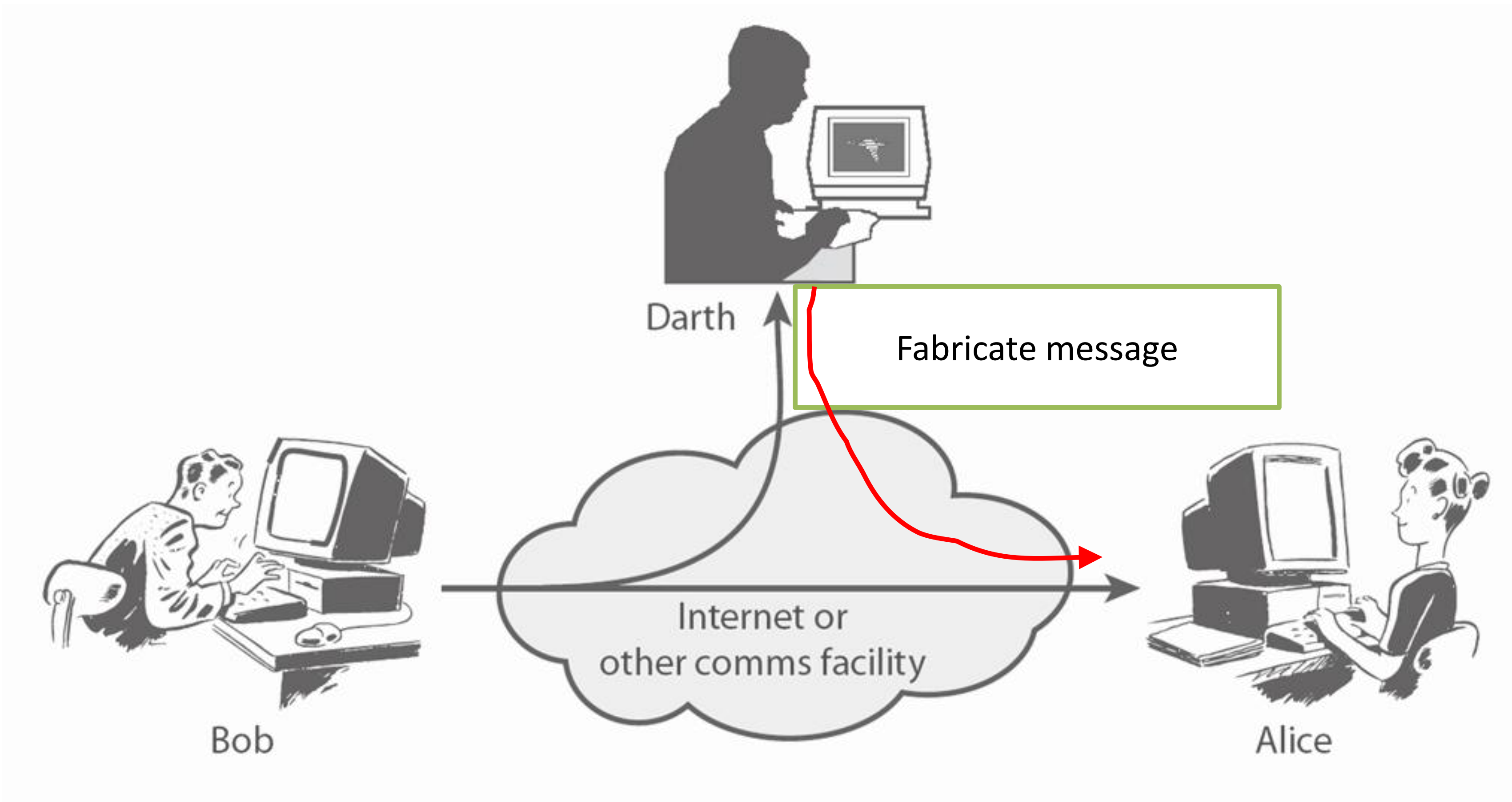
Passive Attack - Interception



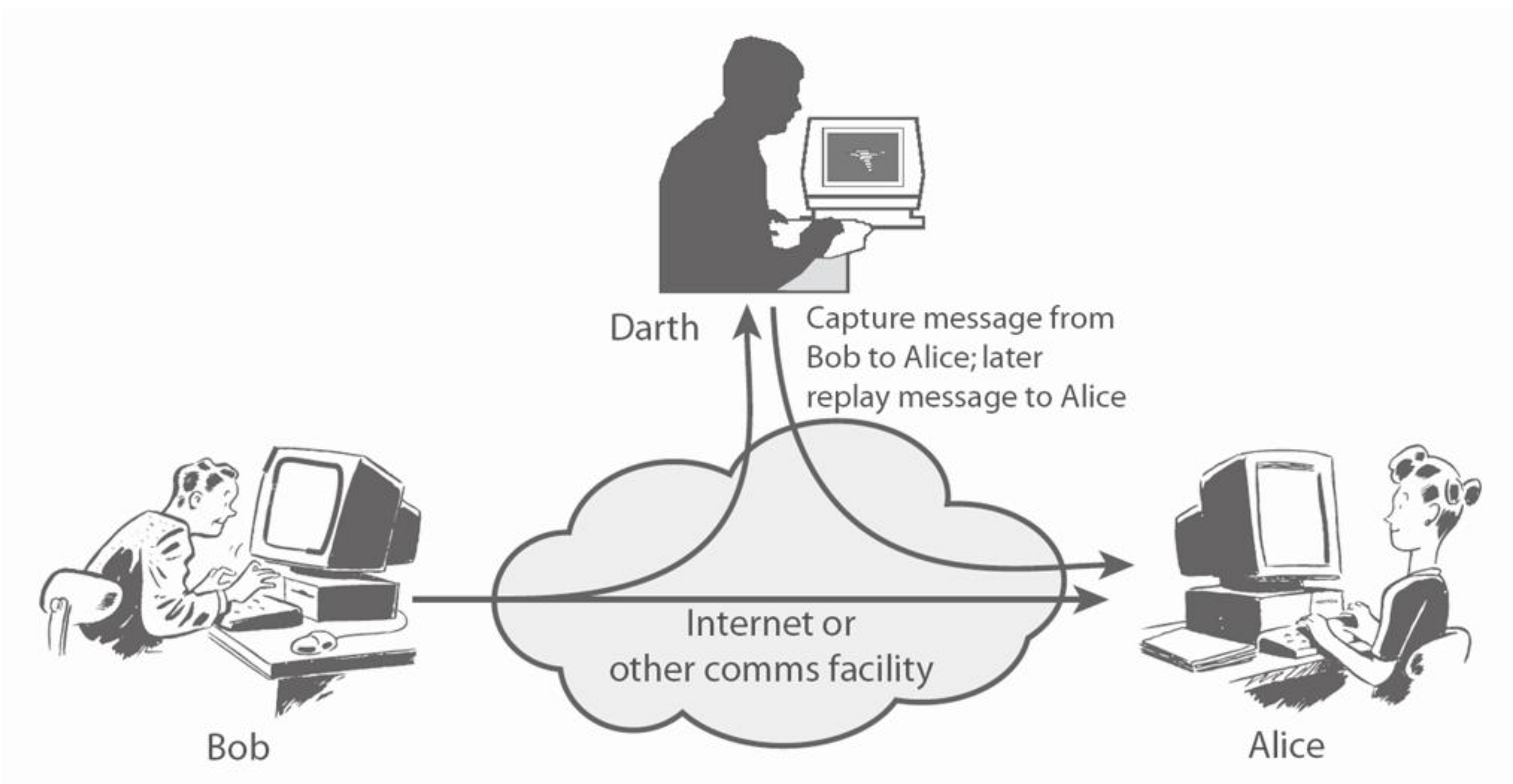
Passive Attack: Traffic Analysis



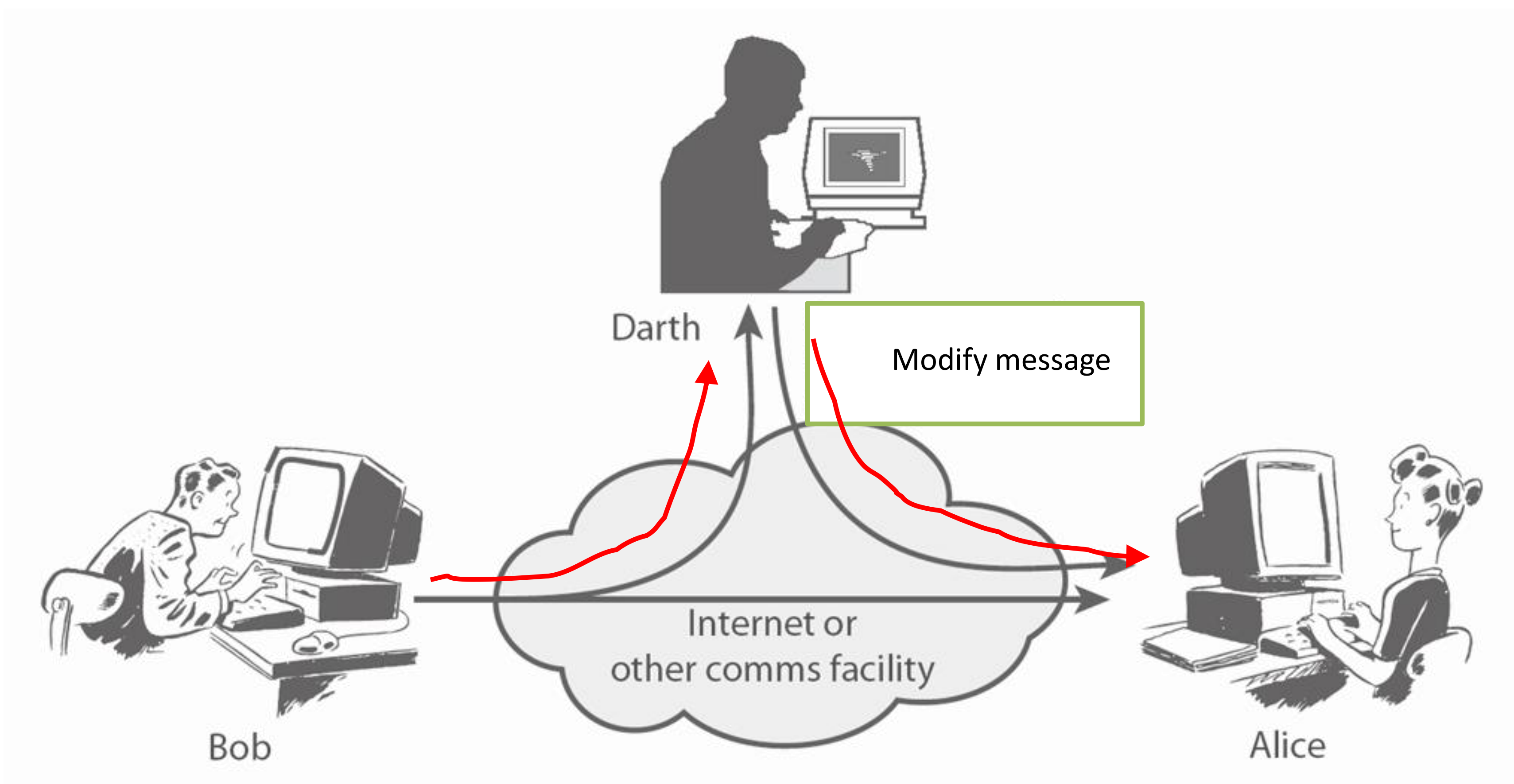
Active Attack: Masquerade



Active Attack: Message Replay



Active Attack: Modification



Key issues

- ***Historical design assumptions***
  - Network protocols were designed to rely messages between TRUSTED partners (US Agencies and first league US Universities)
  - Remember: Trusted != Trustworthy
- ***Design un-intended consequences***
  - Addresses are forgeable
  - Content is forgeable
  - Content can be malicious
  - Rely operators can be malicious

Possible Countermeasures

- ***Cryptographic network security services aka Security protocols***
  - Kerberos,
  - IPSec
  - TLS/SSL
- ***Non-cryptographic network security services***
  - Firewalls
  - Intrusion Detection Systems
  - Honeypots

Active Attack: Denial of Service

- ***an action that prevents or impairs the authorized use of networks, systems, or applications***
- ***Attacks to***
  - network bandwidth
  - system resources (network stack)
  - application resources
- ***IP source spoofing***
- ***Syn flooding***
- ***Missing acks etc.***

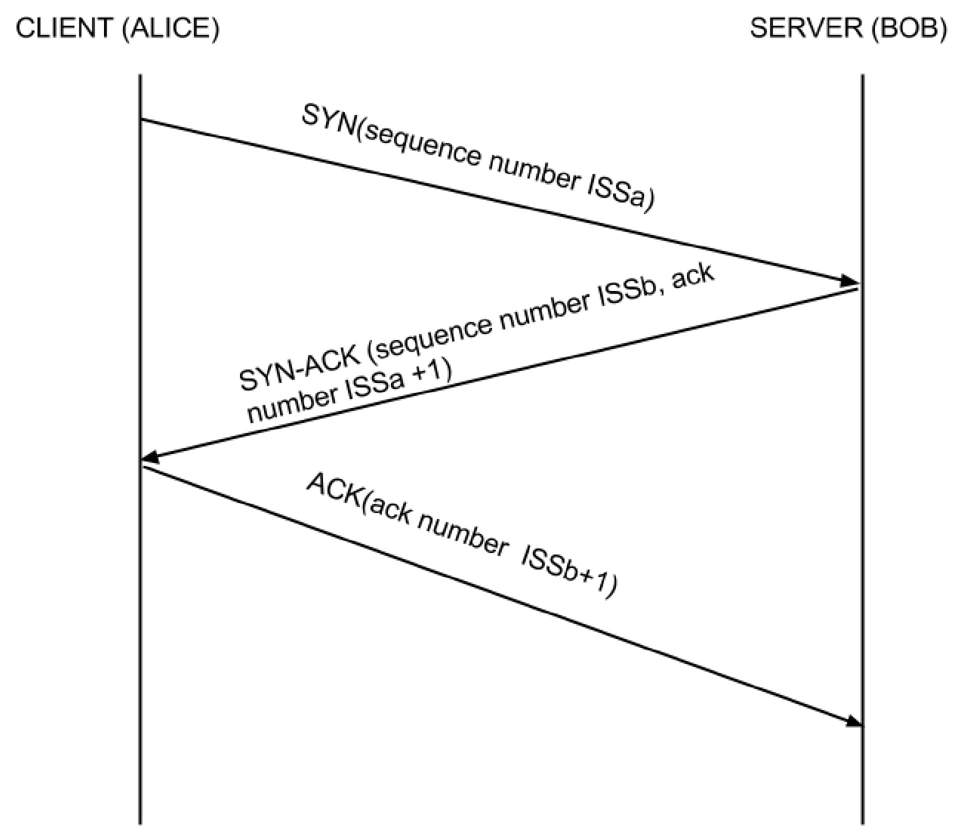
Example: Source Address Spoofing

- ***Use forged source addresses***
  - given sufficient privilege to “raw sockets”
  - easy to create
  - real source is much harder to identify
- ***What happens***
  - generate large volumes of packets with different, random, source addresses
  - cause some congestion when people respond to the innocent recipient (“sender” of the message)
- ***Why?***
  - Source in IP is used for identification not authentication (same as in real mail headers)

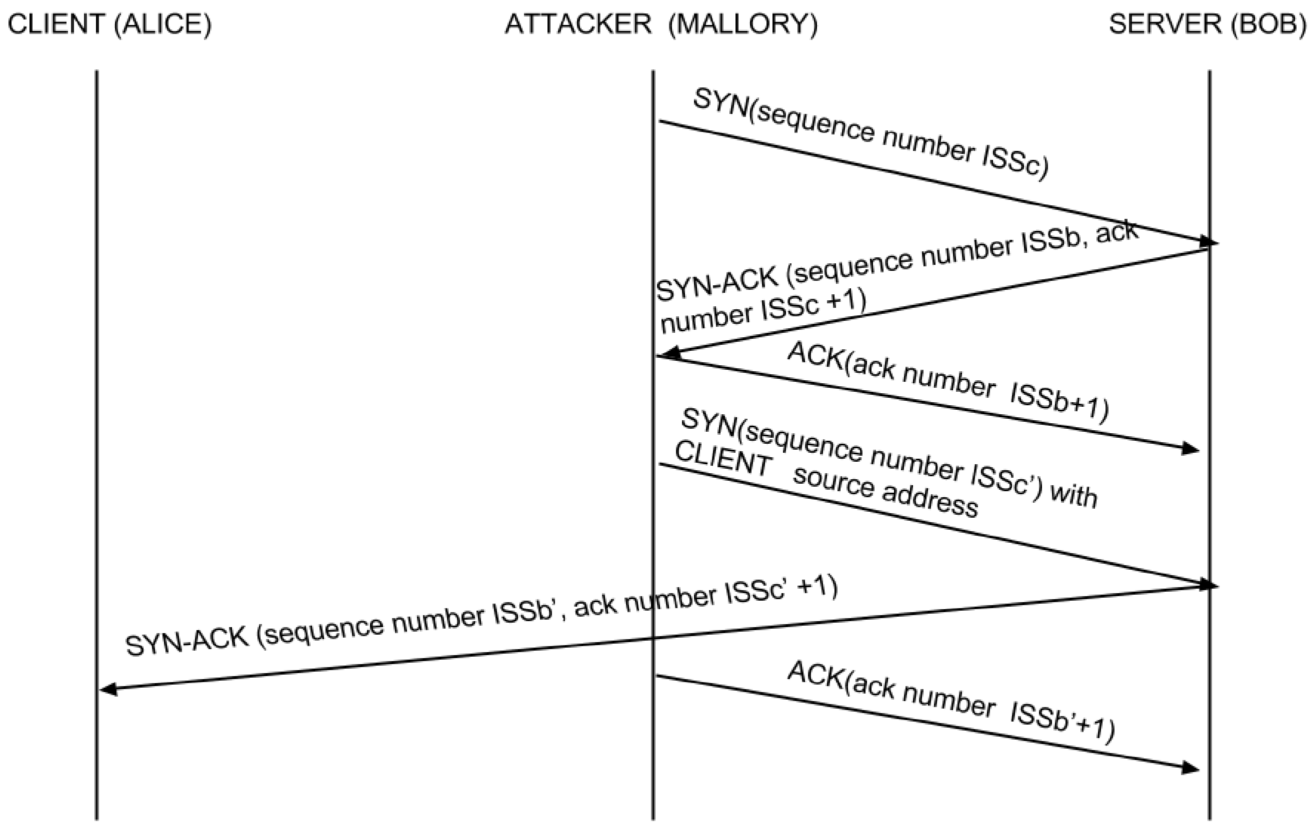
Example: TCP Attacks

- ***TCP connections have associated state***
  - Starting sequence numbers, port numbers
- ***Problem – what if an attacker learns these values?***
  - Port numbers are sometimes well known to begin with (ex. HTTP uses port 80)
  - Sequence numbers are sometimes chosen in very predictable ways

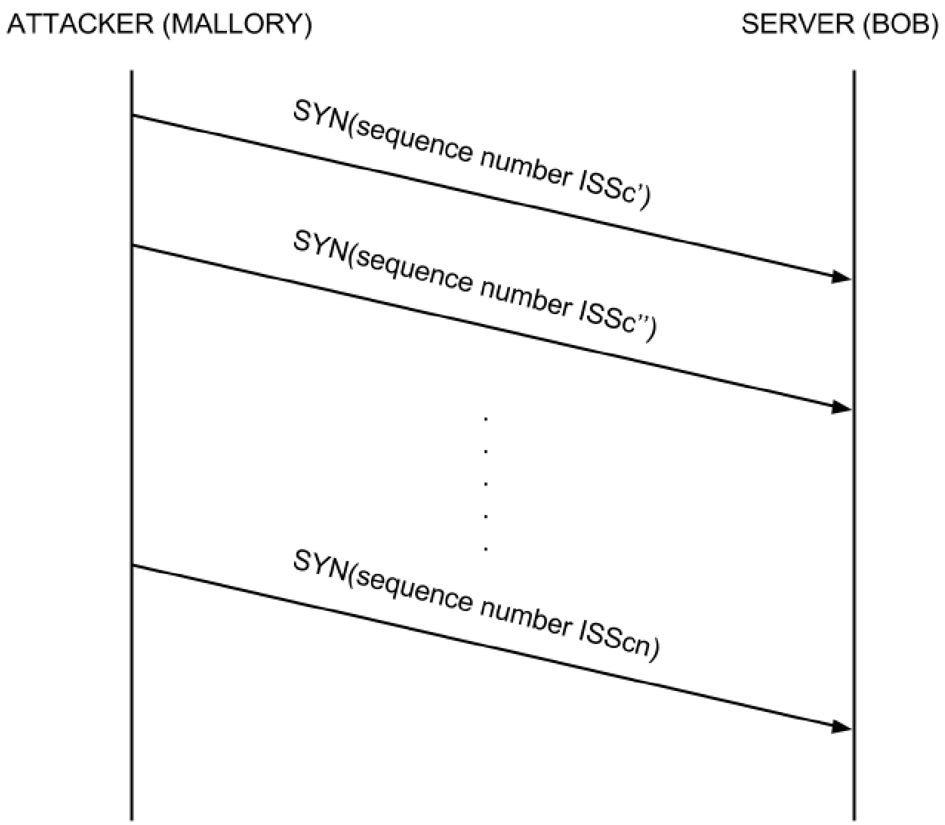
TCP Three-Way Handshake protocol



TCP Session Hijacking



TCP SYN Flooding Attacks



Respond to a Syn Flooding?

- ***X receives a "base traffic" of several GBs of initial TCP handshake***
  - Several scattered servers in China
- ***How to respond?***
  - Contacting ISP doesn't seem to work

Key ideas behind security protocols

- ***Vulnerability: headers and content are forgeable***
- ***Countermeasure: make headers unforgeable***
  - 1-way function of
  - headers AND
  - secret, but only known to sender AND
  - easily verifiable by recipient
- ***Problem 1: freshness***
  - Make sure that result of 1-way function is not always the same (as other wise people can reply it)
- ***Problem 2: bootstrapping***
  - Make sure that sender and recipient initially share some secret
- ***Same for content***

Building Secure Tunnels

- ***Logical connections between two endpoints that crosses an insecure network***
- ***Provide***
  - Data integrity, confidentiality and data origin authentication
- ***Built as follows***
  - Authenticated key establishment protocol
  - Key Derivation
  - Traffic Protection using Derived Keys

IPSec

- ***general IP Security mechanisms***
- ***provides***
  - authentication
  - confidentiality
  - key management
- ***applicable to use over LANs, across public & private WANs, & for the Internet***

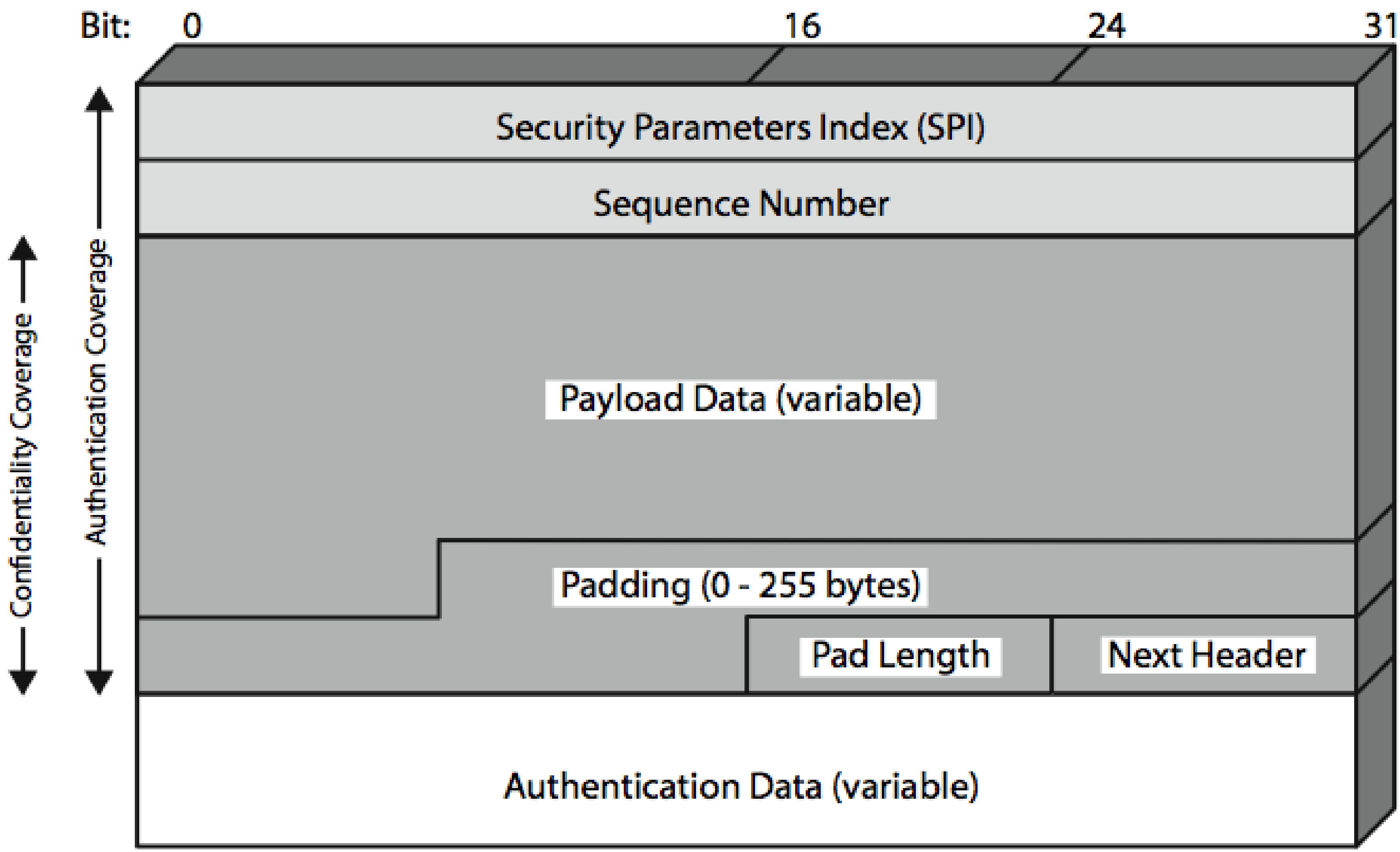
IP Security Architecture

- ***mandatory in IPv6, optional in IPv4***
- ***have two security header extensions:***
  - Authentication Header (AH) (RFC 4302)
  - Encapsulating Security Payload (ESP) (RFC 4303)
  - Key Exchange function
- ***VPNs want both authentication/encryption***
  - hence usually use ESP

Authentication Header (AH)

- ***provides support for data integrity & authentication of IP packets***
  – end system/router can authenticate user/app
  – prevents address spoofing attacks by tracking sequence numbers
- ***based on use of a MAC***
  – HMAC-MD5-96 or HMAC-SHA-1-96
- ***parties must share a secret key***

Encapsulating Security Payload (ESP)



Security Associations

- ***a one-way relationship between sender & receiver that affords security for traffic flow***
- ***defined by 3 parameters:***
  - Security Parameters Index (SPI)
  - IP Destination Address
  - Security Protocol Identifier
- ***has a number of other parameters***
  - seq no, AH & EH info, lifetime etc
- ***IP implementation has a database of Security Associations***

SSL / TLS

- ***Transport Layer Security protocol, ver 1.0***
  - De facto standard for Internet security
  - The primary goal of the TLS protocol is to provide confidentiality and data integrity between two communicating applications
  - In practice, used to protect information transmitted between browsers and Web servers
- ***Based on Secure Sockets Layers protocol, ver 3.0***
  - Same protocol design, different algorithms
- ***Deployed in nearly every web browser***

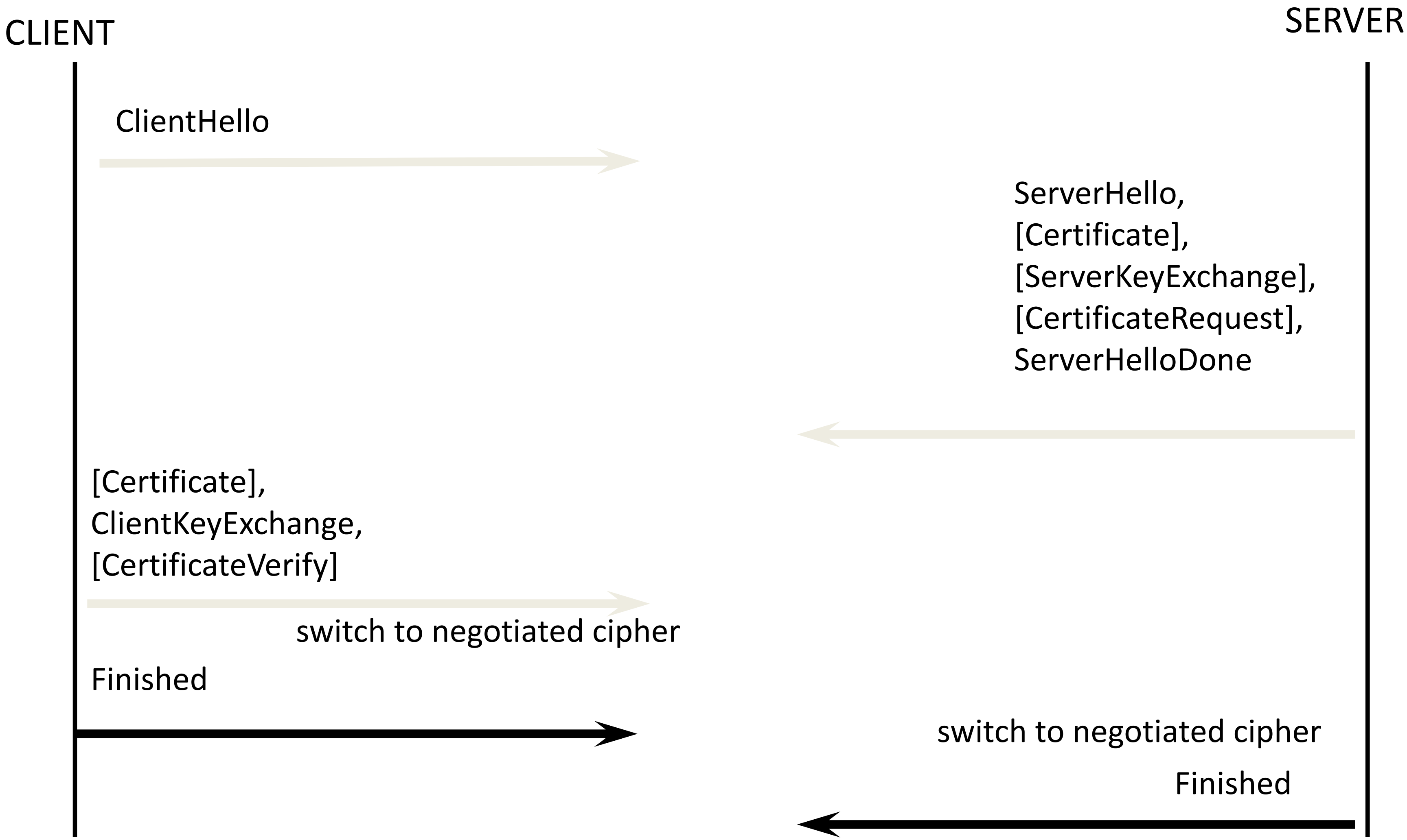
TLS Basics

- ***TLS consists of two protocols***
- ***Handshake protocol***
  – Use public-key cryptography to establish a shared secret key between the client and the server
- ***Record protocol***
  – Use the secret key established in the handshake protocol to protect communication between the client and the server
- ***We will focus on the handshake protocol***

TLS Handshake Protocol

- ***Two parties: client and server***
- ***Negotiate version of the protocol and the set of cryptographic algorithms to be used***
  - Interoperability between different implementations of the protocol
- ***Authenticate client and server (optional)***
  - Use digital certificates to learn each other's public keys and verify each other's identity
- ***Use public keys to establish a shared secret***

Handshake Protocol Structure



ClientHello

CLIENT

SERVER

ClientHello

Client Random [28]
Suggested Cipher Suites: TLS_RSA_WITH_IDEA_CBC_SHA TLS_RSA_WITH_3DES_EDE_CBC_SHA TLS_DH_DSS_WITH_AES_128_CBC_SHA
Suggested Compression Algorithm: None

ServerHello

CLIENT

Server Hello

SERVER

Server Random [28]

Use Cipher Suite:
TLS_RSA_WITH_3DES_EDE_CBC_SHA
Session ID: 0xa00372d4XS

Certificate Chain

SubjectAltName: SuperStoreOutlet
Public Key: 0c521aa593
Issuer:SuperStoreHQ

SubjectAltName: SuperStoreHQ
Public Key: 0x9f400862
Issuer:Verisign

ClientKeyExchange

CLIENT

SERVER

M3

A: Client Key Exchange	A: RSA_Encrypt(ServerPublic Key, Secret)
B: ChangeCipherSpec	NONE
C: Finished	MD5(M1 || M2 || M3A) SHA(M1 || M2 || M3A)

ServerKeyExchange

CLIENT

SERVER

M4

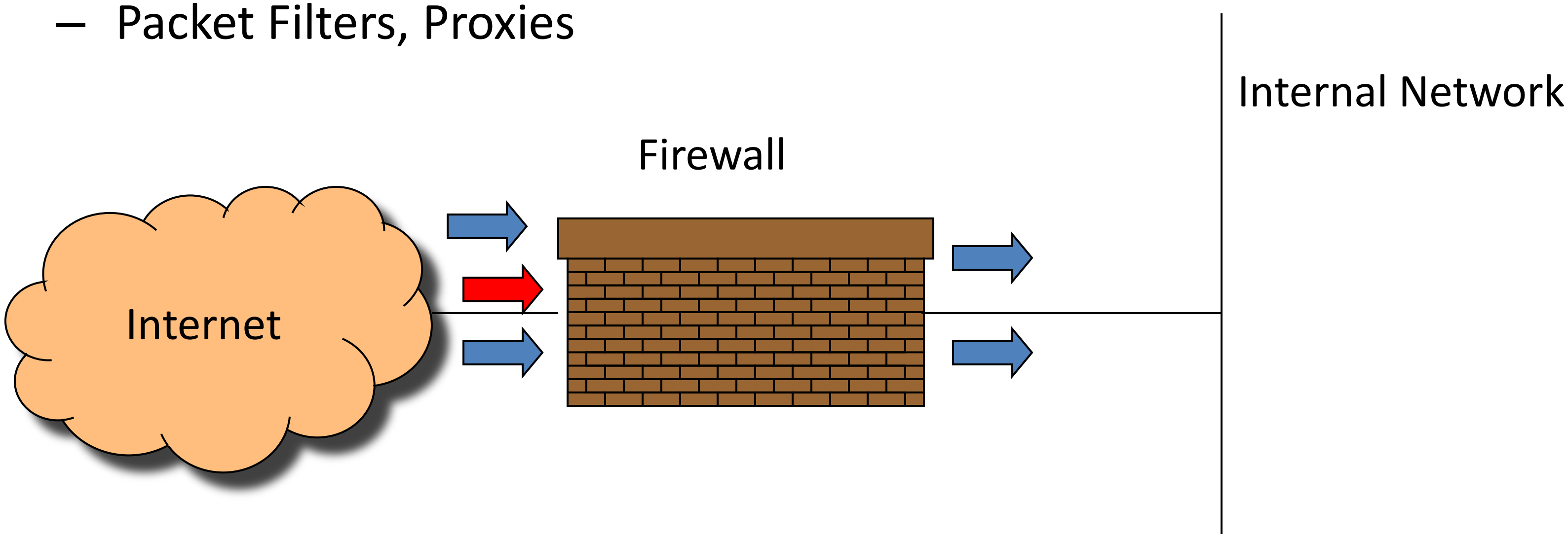
A: ChangeCipherSpec	NONE
B: Finished	MD5(M1 || M2 || M3A || M3C) SHA(M1 || M2 || M3A|| M3C)

Firewalls

- ***Lots of vulnerabilities on hosts in network***
- ***Users don't keep systems up to date***
  - Lots of patches
  - Lots of exploits in wild (no patch for them)
- ***Solution***
  - Limit access to the network
  - Put firewalls across the perimeter of the network
  - Try to identify "signature" of attack and stop it
    - At network or application level

Firewalls

- ***Firewall inspects traffic through it***
- ***Allows traffic specified in the policy***
- ***Drops everything else***
- ***Two Types***
  - Packet Filters, Proxies



Packet Filters

- ***Work at Network and Transport Layer***
- ***Packet filter selectively passes packets from one network interface to another***
- ***Usually done within a router between external and internal networks***
  - screening router

Packet Filters

- ***Data Available***
  - IP source and destination addresses
  - Transport protocol (TCP, UDP, or ICMP)
  - TCP/UDP source and destination ports
  - Packet options (Fragment Size etc.)
- ***Actions Available***
  - Allow the packet to go through
  - Drop the packet (Notify Sender/Drop Silently)
  - Alter the packet (NAT)
  - Log information about the packet

Application-Level Proxies

- ***Implements the server and client part of the protocol on the firewall***
- ***Proxy acts as a server for clients requests***
  - Validate client requests
- ***Proxy act as a client and connects to the destination server***

Firewall Rules

- ***Permissive Policies***
  - Allow all traffic but block certain dangerous services
- ***Restrictive Policies***
  - Block all traffic and allow only traffic know to meet a useful purpose such as HTTP, POP3, SMTP, SSH
- ***An example:***
  - Allow from internal network to Internet: HTTP, FTP, SSJ, DNS
  - Allow from anywhere to mail server: SMTP
  - Allow from mail server to Internet: SMTP, DNS
  - Allow from inside to mail server: SMTP, POP3
  - Allow reply packets
  - Block everything else

Firewall Limitations

- ***No protection against insider attacks***
- ***No "message content" based filtering***
  - Deep packet inspection only works if you have not an encrypted connection (and anyhow there are a lot of applications)
- ***No dection of protocol tunneling***
- ***No encrypted messages filtering***

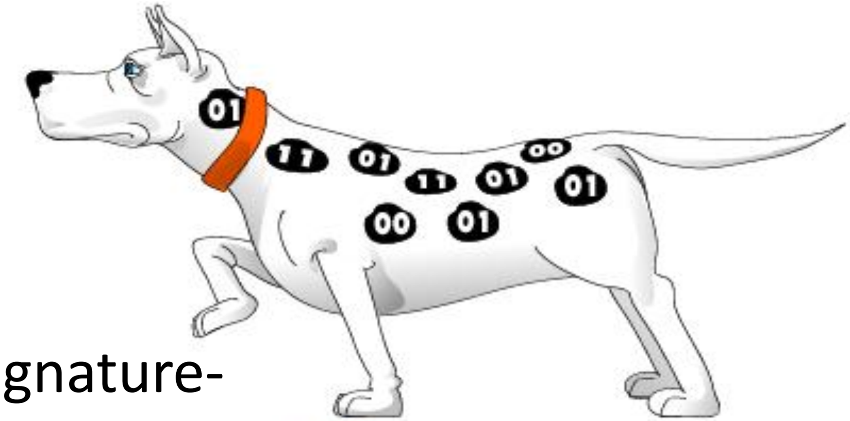
Intrusion Detection Systems

- ***Firewalls allow traffic only to legitimate hosts and services***
- ***Traffic to the legitimate hosts/services can have attacks***
- ***Solution***
  - Intrusion Detection Systems
  - Monitor data and behavior
  - Report when identify attacks

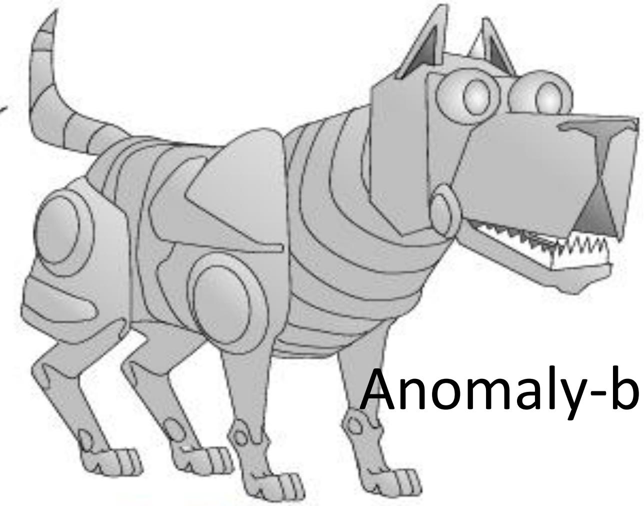
UNIVERSITY OF TRENTO - Italy

Types of IDS

eit Digital MASTER SCHOOL



Signature-based



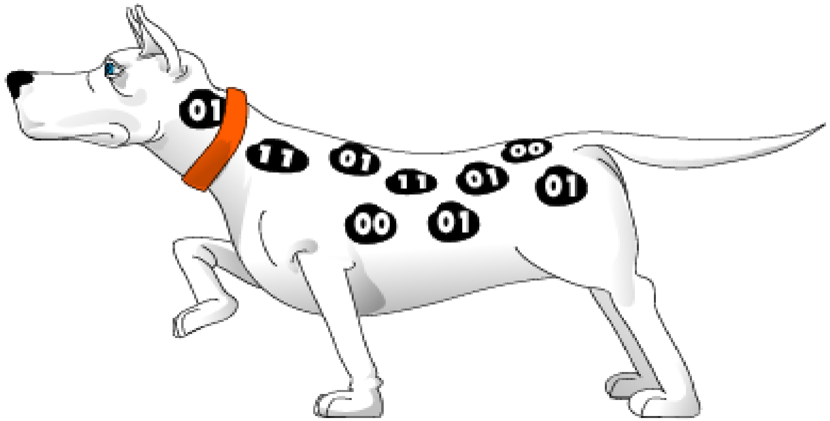
Anomaly-based

Host-based

Network-based

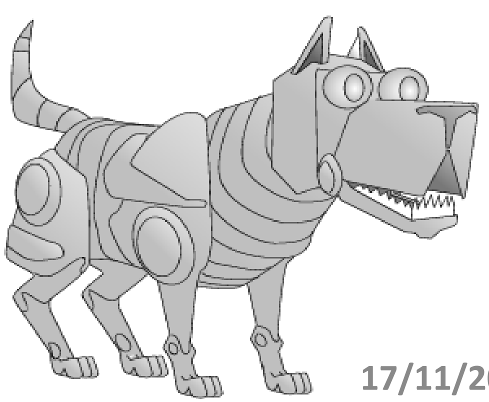
Signature-based IDS

- ***Characteristics***
  - Uses known pattern matching to signify attack
- ***Advantages***
  - Widely available
  - Fairly fast
  - Easy to implement
  - Easy to update
- ***Disadvantages***
  - Cannot detect attacks for which it has no signature



Anomaly-based IDS

- ***Characteristics***
  - Uses statistical model or machine learning engine to characterize normal usage behaviors
  - Recognizes departures from normal as potential intrusions
- ***Advantages***
  - Can detect attempts to exploit new and unforeseen vulnerabilities
  - Can recognize authorized usage that falls outside the normal pattern
- ***Disadvantages***
  - Generally slower, more resource intensive compared to signature-based IDS
  - Greater complexity, difficult to configure
  - Higher percentages of false alerts

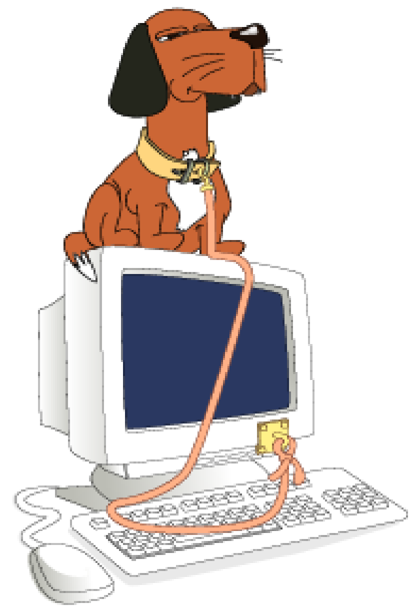


Network-based IDS

- ***Characteristics***
  - NIDS examine raw packets in the network passively and triggers alerts
- ***Advantages***
  - Easy deployment
  - Unobtrusive
  - Difficult to evade if done at low level of network operation
- ***Disadvantages***
  - Different hosts process packets differently
  - NIDS needs to create traffic seen at the end host
  - Need to have the complete network topology and complete host behavior

Host-based IDS

- ***Characteristics***
  - Runs on single host
  - Can analyze audit-trails, logs, integrity of files and directories, etc.
- ***Advantages***
  - More accurate than NIDS
  - Less volume of traffic so less overhead
- ***Disadvantages***
  - Deployment is expensive
  - What happens when host get compromised?



Honeypots

- ***Information system resources whose value lie in their ellicit use***
- ***Systems to track attackers and learn about new attack techniques***
- ***Low- interaction honeypots***
  - Limited collection of an attacker's activities logs
  - Easy to be detected by an attacker
- ***High-interaction honeypots***
  - Risk of being misused by the attacker

Network Security Standard

- ***ISO 27033:2009***
- ***Part 1***
  - Guidance on how to implement network security
  - Guidance and process on how to identify network security risks
  - Guidance on how to select security controls in ISO 27002
- ***Part 2***
  - Guidance on how to implement a security architecture
- ***Part 3***
  - Illustrates network specific security risks and threats

Reading Material

- ***Chapters 16 and 17. Dieter Gollman.Computer Security, Wiley.***
- ***Chapters 6, 8, 9, 21. William Stallings and Laurie Brown. Computer Security: Principles and Practice, 3rd edition, Prentice Hall.***
- ***Read this paper for ideas (Car=Drones)***
  – http://www.autosec.org/pubs/cars-usenixsec2011.pdf