PA197 Secure Network Design
1. Introduction

Eva Hladká, Luděk Matyska

Fakulty of Informatics

February 20, 2018

Content

Course Organization

- attending the lectures
- the knowledge acquired course materials will be published on the course webpage
- assessment methodology:
- course literature:
    - slides, RFCs, ...
    - literature being announced in relevant course parts

Course Overview

- the course goal:
  - to provide basic network architectures and functions
    - data transmission
    - end to end argument
    - routing
    - switching
    - ...
  - general requirements on the security and reliability
    - implication towards the architecture design
  - Network architectures from the point of secure
    - reliable design also in
    - ad-hoc/sensor networks
    - vehicular and/or mobile networks

Data Transmissions - Introduction

- **the main goal:** to ensure a transmission of bits (= the content of passed frames) between sender and receiver
- several standards (RS-232-C, CCITT V.24, CCITT X.21, *IEEE 802.x*) defining electrical, mechanical, functional, and procedural characteristics of interfaces used for connecting various transmission media and devices, e.g.:
    - parameters of the transmitted signals, their meaning and timing
    - mutual relationships of control and state signals
    - connectors' wiring
    - and many many others

- *Bit-to-Signal Transformation*
  - representing the bits by a signal – electromagnetic energy that can propagate through medium
- *Bit-Rate Control*
  - the number of bits sent per second
- *Bit Synchronization*
  - the timing of the bit transfer (synchronization of the bits by providing clocking mechanisms that control both sender and receiver)
- *Multiplexing*
  - the process of dividing a link (physical medium) into logical channels for better efficiency
- *Circuit Switching*
  - circuit switching is usually a function of the physical layer
  - (packet switching is an issue of the data link layer)

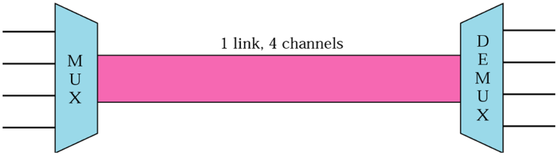
Signals

- data is transferred (via transmission media) in the form of (electromagnetic) *signals*
  - the data have to be converted into the signals
- *signal* = a function of time representing changes of physical (electromagnetic) characteristics of the transmission media
- data that have to be transferred (0s and 1s) – *digital* (binary)
- signals spread through the transmission media – *analog* or *digital*
  - some media suitable for both analog and digital transmission – wired media (coaxial cable, twisted pair), optical fibre
  - some media suitable just for analog transmission – ether (air)

Transmission Media

- provide an environment for the functionality of physical layer
- basic distinction:
  - *guided (wired) media*
    - provide a conduit from one device to another
    - twisted pair (LANs, up to 10 Gbps), coaxial cable, optical fibre (backbones, hundreds of Gbps), etc.
  - *unguided (wire-less) media*
    - transfer an electromagnetic wave without the use of physical conductor
    - the signals are broadcasted (spread) via ether (air, vacuum, water, etc.)
    - radio signals, microwave signals, infrared signals, etc.

- *multiplexing* – a technique of sharing an available bandwidth by concurrent communication channels
  - the goal is to maximize the utilization of the media
  - applied especially for optical fibres and non-wired media



- for analog signals:
  - *Frequency-Division Multiplexing (FDM)*
  - *Wave-Division Multiplexing (WDM)*
- for digital signals:
  - *Time-Division Multiplexing (TDM)*

End to End (E2E) argument

How to provide demanded functionality in computer networks?

- **End-to-End (E2E)** argument
  - application demanded functionality is possible to provide wit knowledge and by application
    - $\Rightarrow$ if it is possible, communication protocol operations have to be defined by realization only in communication system end nodes or in the closest distance
    - in lower system levels protocol function should be implemented only if performance increases.
  - suitable for applications demanding higher degree fidelity transported data and some latency is tolerated.
- **Hop-by-Hop (HbH)**
  - repeating specific functionality on the each two–point connection is possible to obtain increasing performance
  - it requires storing state informations on inside network nodes $\Rightarrow$ limited scalability
  - useful for applications,where minimize latency is more important then transported data fidelity, (e.g. real-time applications)

Routing

- the main goal of routing is:
  - to find optimal paths
    - the optimality criterion is a *metric* – a cost assigned for passing through a network
  - to deliver a data packet to its receiver
- the routing *usually* does not deal with the whole packet path
  - the router deals with just a single step – to whom should be the particular packet forwarded
    - somebody "closer" to the recipient
    - so-called *hop-by-hop* principle
  - the next router then decides, what to further do with the received packet

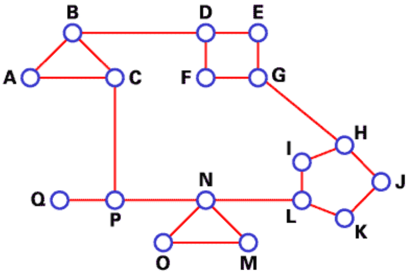
Routing – basic approaches

The basic approaches divide based on the routing table creation/maintenance:

- *static (non-adaptive)*
  - manually (by hand) edited records
  - suitable for a static topology and smaller networks
- *dynamic (adaptive)* – these respond to network changes
  - complex (usually distributed) algorithms
  - e.g.:
    - *centralized* – a centre controls the whole routing
    - *isolated* – every node on its own
    - *distributed* – nodes' cooperation

Routing – mathematical view

- the routing can be seen as a problem of graph theory
- a network can be represented by a graph, where:
  - nodes represent routers (identified by their IP addresses)
  - edges represent routers' interconnection (a data link)
  - edges' value = the communication cost
  - *the goal:* to find paths having minimal costs between any two nodes in the network



Routing – routing algorithms' required features

Required features of any routing algorithm:

- accuracy
- simplicity
- effectivness and scalability
  - to minimize an amount of control information ($\approx$ 5% of the whole traffic!)
  - to minimize routing tables' sizes
- robustness and stability
  - a distributed algorithm is necessary
- fairness
- optimality
  - *"What should be treated as the best path?"*

Basic approaches to distributed routing:

- *Distance Vector (DV)* – Bellman-Ford algorithm
    - the neighboring routers periodically (or when the topology changes) exchange complete copies of their routing tables
    - based on the content of received updates, a router updates its information and increments its *distance vector number*
        - a metric indicating the number of hops in the network
    - i.e., *"all pieces of information about the network just to my neighbors"*
- *Link State (LS)*
    - the routers periodically exchange information about states of the links, to which they are directly connected
    - they maintain complete information about the network topology – every router is aware of all the other routers in the network
    - once acquired, the Dijkstra algorithm is used for shortest paths computation
    - i.e., *"information about just my neighbors to everyone"*

Packet Switching

- Packet switching refers to protocols in which messages are divided into packets before sending and each packet is transmitted individually. Once all packets forming a message arrive at the destination, they are recompiled into the original message.
- Packet switching operation
  - Data are transmitted in short packets, typically an upper bound on packet size is 1000 bytes.
  - Each packet contains part of the user's data and some control information.
  - The control information should at least contain
    - destination address
    - source address
  - Store and forward - Packets are received, stored briefly and past on the next node.
- Advantages
  - Line efficiency – single node to node link can be shared by many packets over time and packets que and and transmitted as fast as possible

Switching Technique

- Virtual Circuits
  - Pre–planned route is established before any packets sent
  - Call setup before the exchange (handshake)
  - all packets follow the same route and arrive in sequence
  - each packet contains a virtual circuit identifier instead of destination address
  - no routing decision required for each packet
  - clear request to drop circuit
- Datagrams
  - Each packet is treated independently with no reference to packets that have gone before.
  - Packets may arrive out of order
  - Packets may go missing
  - Up to receiver to re-order packets and recover from missing packets
  - More processing time per packet node
  - Robust in the face of link or node failures.

Circuit vs. Packet Switching

- Performance
  - propagation delay
  - transmission time
  - node delay
- Packet switching evolution
  - X.25 packet–switched network
  - router–based networking
  - switching vs. routing
  - frame relay network
  - ATM network

- Switching
  - path set up at connection time
  - simple table look up
  - table maintenance via signaling
  - no out of sequence delivery
  - lost path may lost connection
  - much faster than pure routing
  - link decision made ahead of time, and resources allocate then
- Routing
  - can work as connectionless
  - complex routing algorithm
  - table maintenance via protocol
  - out of sequence delivery likely
  - robust: no connections lost
  - significant processing delay
  - output link decision based on packet header contents – at every node

Dual network basis

- Physical and software base
  - Physical base: links and physical equipment
    - Not a subject of this lecture
  - Software base: protocols and applications
    - Subject of this lecture

Network (Communication) Protocols I.

- motivated by the need to communicate among several entities (at least two)
  - *entity* = anything capable of sending or receiving information
- the form/method of the communication must be known to all the participating entities
  - they have to **agree on a protocol**
- human analogy:

Network (Communication) Protocols II.

- the **protocol** defines *"What"* the subject of communication is, *"How"* the communication has to behave and *"When"* does it behave
- they define:
  - *syntax* = structure/format of data (the order in which they are presented)
  - *semantics* = refers to the meaning of each section of bits (how should a particular pattern to be interpreted)
  - *timing* = when data should be sent and how fast they can be sent
- examples of network protocols:
  - UDP, TCP, IP, IPv6, SSL, TLS, SNMP, HTTP, FTP, SSH, Aloha, CSMA/CD, ...

Network Protocol

Network Protocol is a set of rules that defines the format and the order of messages exchanged among two or more communicating entities, as well as the actions performed during sending/receiving that messages.

Standardization

- definition of norms/standards describing various actions, activities, forms/methods of communication, etc. (not only in IT)
- main goals:
  - quality
  - security
  - compatibility
  - interoperability
  - portability
- standards fall into two categories:
  - *de facto* – standards that have not been approved by an organized body but have been adopted as standards through widespread use (they are often established originally by manufacturers)
  - *de jure* – standards legislated by an officially recognized body
- standard IT organizations:
  - ISO, ITU-T, ANSI, IEEE, IETF (*RFCs*), IEC, etc.

ISO/OSI Model I.

- **7-layer model** proposed by OSI organization in order to ensure compatibility and interoperability of communication systems developed by various vendors
- the purpose of layered architecture:
  - each layer is **responsible for particular functionality**
    - it adds some control information to the data in order to do its job
  - each layer **communicates just with its neighbours**
    - each layer uses the services provided by the lower layer and provides its services to the higher layer
    - the functionality is **isolated** in the particular layer (once a layer changes, just the neighbouring layers have to adapt to such a change)
  - logically, the communication is performed just between peer layers; physically, the communication traverses all the lower layers
  - the layers are just an abstraction – the real implementations are more or less different
- 7 layers not widely accepted ⇒ TCP/IP model

ISO/OSI Model vs. TCP/IP Model

ISO / OSI

- **Aplikační vrstva**
  síťové aplikace
- **Prezentační vrstva**
  datová reprezentace
- **Relační vrstva**
  relace, správa relací
- **Transportní vrstva**
  process-process komunikace, spolehlivost
- **Síťová vrstva**
  síťové adresování (logické), směrování
- **Vrstva datového spoje**
  MAC a LLC (fyzická adresace)
- **Fyzická vrstva**
  přenosová média, signály, bitová reprezentace

TCP/IP

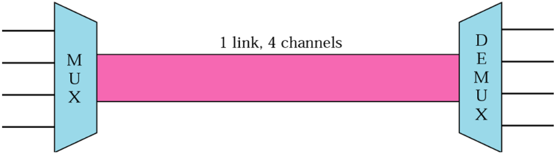
- **Aplikační vrstva**
- **Transportní vrstva**
- **Síťová (Internetová) vrstva**
- **Vrstva přístupu k síti/médiu**

- **Physical Layer:**
  - provides the functionality for an interaction with transmission media
  - provides services for the *Data Link Layer*
    - the Data Link Layer passes/obtains data to/from the Physical Layer in the form of 0s and 1s organized into *frames*
    - the Physical Layer transforms the streams of bits (from frames) into *signals* spread through the transmission media
  - controls the transmission media; for example, decides about:
    - sending/receiving the data (signals)
    - data transformation (coding) into signals
    - the number of logical channels simultaneously transferring data from various sources

- *Bit-to-Signal Transformation*
  - representing the bits by a signal – electromagnetic energy that can propagate through medium
- *Bit-Rate Control*
  - the number of bits sent per second
- *Bit Synchronization*
  - the timing of the bit transfer (synchronization of the bits by providing clocking mechanisms that control both sender and receiver)
- *Multiplexing*
  - the process of dividing a link (physical medium) into logical channels for better efficiency
- *Circuit Switching*
  - circuit switching is usually a function of the physical layer
  - (packet switching is an issue of the data link layer)

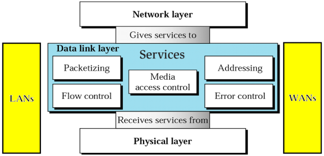
- data is transferred (via transmission media) in the form of (electromagnetic) *signals*
  - the data have to be converted into the signals
- *signal* = a function of time representing changes of physical (electromagnetic) characteristics of the transmission media
- data that have to be transferred (0s and 1s) – *digital* (binary)
- signals spread through the transmission media – *analog* or *digital*
  - some media suitable for both analog and digital transmission – wired media (coaxial cable, twisted pair), optical fibre
  - some media suitable just for analog transmission – ether (air)

- *multiplexing* – a technique of sharing an available bandwidth by concurrent communication channels
  - the goal is to maximize the utilization of the media
  - applied especially for optical fibres and non-wired media



- for analog signals:
  - *Frequency-Division Multiplexing (FDM)*
  - *Wave-Division Multiplexing (WDM)*
- for digital signals:
  - *Time-Division Multiplexing (TDM)*

- **Data Link Layer:**
  - receives *packets* (being passed from the Network Layer) and transforms them into *frames*
  - in cooperation with the Physical layer ensures the transmission of frames between communicating devices interconnected with a *(shared) transmission media*
    - i.e., just the local (inside a segment) delivery (LAN)
  - ensures the transmission reliability between these devices
  - ensures the flow control in order to avoid receiver congestion
  - controls the access of the devices to shared media (Medium Access Control)



- *Framing*
  - the incoming packets (being passed from the Network Layer) are encapsulated into *frames*
- *Addressing*
  - provides the addresses of physical layer entities – *physical/MAC addresses*
  - frames contain source and destination addresses of communicating entities
- *Error Control*
  - it's not possible to eliminate the errors occurring on the physical layer
  - L2 layer ensures the required level of reliability of the data link (error detection and correction)
- *Flow Control*
  - prevents the receiver congestion
  - *stop-and-wait* mechanism, *sliding-window* mechanism, . . .
- *Medium Access Control – MAC*
  - necessary in environments, where the transmission media is shared by several entities
  - eliminates collisions caused by multiple (concurrent) transmissions

- a concept of redundancy is used
  - sender adds bits whose value is a function of transmitted data
  - receiver calculates the same function and if the values differ, it detects (tries to repair) an error
  - when using error detection only (or if the error is unrepairable), the receiver requests the sender to repeat the transmission
- *Error Detection, Automatic Request for Retransmission (ARQ)*
  - error detection and transmission repetition ensurance
  - suitable for little-lossy transmission media
  - even/odd parity, *Cyclic Redundancy Check (CRC)*, etc.
- *Forward Error Correction (FEC)*
  - error detection and attempts to data correction (using redundant data)
  - suitable for lossy transmission media (especially with high transmission latency)
  - e.g., *Hamming code*
  - for details see *PV169: Communication Systems Basics*

- the functionality responsible for coordination of multiple devices' access to shared transmission media
- *The goal:* the elimination of collisions caused by concurrent transmissions (emissions)
  - i.e., concurrent transmissions to a shared transmission environment
- medium access protocols:
  - *random-access protocols* – Aloha, CSMA/CD, CSMA/CA
  - *controlled-access protocols* – based on reservations, polling, tokens, etc.
  - *channelization protocols (multiplex-oriented access)* – FDMA, TDMA, etc.

- **Network Layer:**
  - provides services for the *Transport Layer*:
    - receives *segments* from the Transport Layer and transforms them into *packets*
    - in cooperation with the Data Link Layer ensures the packets' transmission between communicating nodes *(even between different LANs)*
  - logically joins independent LAN networks
    - the upper layers are provided with an illusion of just a single wide-area network (*WAN*)
  - allows unique identification (addressing) of every host/device on the Internet
  - ensures *routing* of passing packets
  - in cooperation with the Data Link Layer associates the L3-addresses with the L2/MAC-addresses (and vice versa)
  - further services: multicast

- *Internetworking*
  - logical gluing of heterogeneous physical networks together to look like a single network (from the upper layers' point of view)
    - by such an interconnection, an *internetwork* (shortly *internet*) is created
  - an illusion of a uniform environment provided by a single wide-area network
- *Packetizing*
  - received segments are transformed into packets
- *Fragmenting*
  - a technique to solve the problem of heterogeneous MTUs – when a datagram is larger than the MTU of the network over which it must be sent, it is divided into smaller fragments which are each sent separately
- *Addressing*
  - the entity addresses used on the network layer – so-called *IP addresses*, unique throughout the whole network
  - packets contain source and destination addresses of

- *Address Resolution*
  - ARP, RARP protocols
- *Routing*
  - the process of selecting paths in a network along which to send network traffic from a source to a particular destination
- *Control Messaging*
  - providing basic information about unavailability to deliver a packet, about a network/host state, etc. – ICMP protocol

Transport Layer:

- provides its services to the *Application Layer*:
  - obtains data coming from sending application and transforms them into *segments*
  - received segments delivers to the destination application
- in cooperation with the network layer ensures data (segments) delivery between communicating *applications/processes*
  - providing transmission reliability, if required
  - provides them with a logical communication channel
    - an illusion of direct physical interconnection
  - so-called *process-to-process delivery*
- the lowest layer providing so-called *end-to-end* services
  - the headers generated on the sender's side are interpreted "only" on the receiver's side
  - the transport layer data are seen by routers as a payload of transmitted packets

- *Packetizing*
    - the data provided by an application are transformed into packets (having a transport header added)
- *Connection Control*
    - *connection-oriented* and *connectionless* services
- *Addressing*
    - the addresses of transport layer entities (= network applications/services) – so-called *ports*
    - the packets contain source and destination ports (an identification of source and destination application)
        - an application is uniquely identified in the network by the pair *IP_address:port*
- *Connection Reliability*
    - *Flow Control* and *Error Control*
        - provided on the node-to-node principle by lower layers, L4 provides it on the *end-to-end* principle
    - ensures a reliability over *best-effort* service (IP)
- *Congestion Control and Quality of Service (QoS) ensurance*

Application Layer:

- provides services to *users*:
  - application programs specific for a particular purpose
    - e.g., electronic mail, WWW, DNS, etc. etc.
  - applications = the main reason for computer networks existence
- comprises *network applications/programs* and *application protocols*
  - application protocols (HTTP, SMTP, etc.) are **parts of** network applications (web, email)
    - they are not applications on their own
    - the protocols define a form of communication between communicating applications
  - application protocols define:
    - types of messages, which the applications exchange (*request/response*)
    - messages' syntax
    - messages' semantics (a semantics of particular fields)
    - rules, when and how the messages are exchanged

Redundancy principle in network design

- Basic principle in nature
  - duplication important viscus in animal's bodies – kidneys
- Basic principle in networks
  - topology (see topology of CESNET2 network)
  - parts of protocols (CRC on several layers)

Wireless Ad-hoc Network

- a collection of autonomous nodes that communicate with each other by forming a multihop radio network and maintaining connectivity in a decentralized manner
  - each node functions as both a *host* and a *router*
  - the control of the network is distributed among the nodes
  - the network topology is (in general) dynamic
    - the connectivity among the nodes may vary in time due to node departures, new node arrivals, and the nodes' mobility
    - $\Rightarrow$ a need for efficient routing protocols that allow the nodes to communicate over multihop paths in an efficient way
- these networks pose many complex issues $\Rightarrow$ there are many open problems for research
  - without a central infrastructure, things become much more difficult

- very fast construction
  - no need to establish wired connections
- resilient
  - no single point of failure, such as a base station
- spectrally more efficient than cellular networks
  - every node can communicate with any other node (sometimes even simultaneously), so nodes can make better use of the channel

- problems arise due to:
  - lack of a central entity for network organization
    - the participating nodes must organize themselves into a network
    - *self-organization* is a must
  - limited range of wireless communication
    - data have to be delivered over a path involving multiple nodes
    - $\Rightarrow$ mechanisms for dynamic path identification and management are required
  - mobility of participants
    - the network nodes may be allowed to move in time and space
    - the network quality depends on the speed to adapt to new topologies
    - $\Rightarrow$ **Mobile Ad-hoc Networks (MANETs)**
- among others, the following issues have to be addressed:
  - *medium access control* – no base station can assign transmission resources (it must be decided in a distributed fashion)
  - *routing* – finding a route from one participant to another

- often (but not always), the participants in an ad-hoc network (not only sensor network) draw energy from batteries
- it is desirable to sustain a long run time for:
  - individual nodes/devices
  - the network as a whole
    - usually, application demands do not bother with individual nodes, as long as the global application-dependent objective can still be fulfilled
- employed networking protocols have to take the limited energy into account and behave in an energy-efficient way
  - e.g., use routes with low energy consumption (energy/bit)
  - e.g., take available battery capacity of devices into account
  - How to resolve conflicts between different optimizations?
- some form of recharging or energy scavenging from the environment is often used in order to increase the available energy

- *Available energy*
  - sensor nodes are operated by batteries that provide limited energy for the node
- *Processing power*
  - employed micro controllers usually provide very limited processing performance (due to size and energy restrictions)
- *Memory and storage*
  - the characteristics of the available memory usually correlate with the size of the micro controller
- *Bandwidth and throughput*
  - wireless radio transceivers are optimized for low-energy operation ⇒ they provide a relatively small bandwidth to the application
- *Reliability*
  - depending on the application scenario, the demands for the reliability (both communication reliability and error-proneness of the hardware) can strongly differ
- *Addressing*
  - typically, off-the-shelf sensor nodes do not have a globally unique address pre-programmed ⇒ networking mechanisms must either dynamically allocate unique addresses or even abandon address-based techniques
- *Scalability*
  - a primary constraint – the scalability of employed methods and algorithms

Recapitulation

- *Course organization*
- *Course overview*
  - basic network functions
    - data transmission, E2E argument, routing and switching
  - general requirements on the security and reliability
    - implications towards the architecture design, ISO/OSI and TCP/IP models
  - reliable design of selected networks
    - senzor, mobile