# IT Compact Course

Hardware and Software

**Internet and Web**

Cryptography

Kaspar Etter, October 2011

2. Internet and Web

# Outline

2.1. Introduction

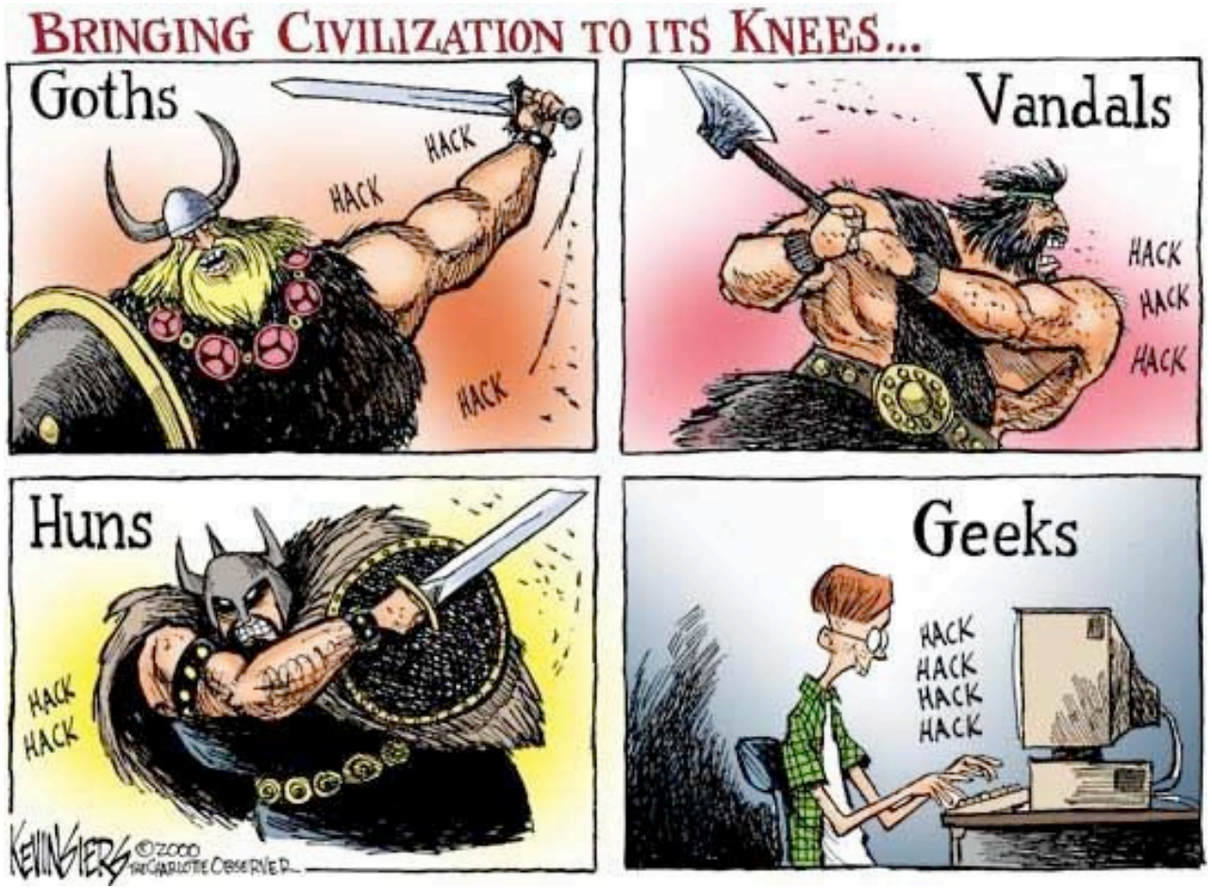
2.2. Internet Layer

2.3. Transport Layer

2.4. Application Layer

2.5. World Wide Web

2.6. Semantic Web




General reference and source for further reading: www.wikipedia.org

2. Internet and Web

# 2.1. Introduction

- Network: Nodes connected by physical medium called link
- Switched network: Scalability by nodes that redirect traffic
- Circuit switching: Reserved resources, guaranteed perform.
- Packet switching: Forwarding chunks of information, delays

- Internet: Network of networks with hierarchical structure
- Based on packet switching with only best effort delivery
- Internet service providers (ISPs) provide access to Internet
- ISPs are interconnected by international backbone providers

# History of the Internet

- 1969: ARPANET (Advanced Research Projects Agency)
- 1974: Unify different network methods by an internetwork protocol that makes hosts responsible for reliability
- 1982: Simple Mail Transfer Protocol (SMTP) defined
- 1983: TCP/IP becomes only approved protocol on ARPANET
- 1983: Introduction of the Domain Name System (DNS)
- 1990: Creation of the World Wide Web (WWW)
- Late 1990's: Commercialization of the World Wide Web
- Now: Ever increasing number of devices with Internet access

# Protocols

- Communication in networks is controlled by protocols
- Protocols define message format, message order, actions on message receipt and problem handling (incl. congestion)
- Standardization of protocols by various organizations
- Names often abbreviated to TLAs (three-letter acronyms)
- Protocols have to compensate for deficiencies of network:
  - Packet loss: Detect by sequence numbers, retransmission
  - Bit-error: Detect by checksums, perform error-correction

# Performance

- Two fundamental measures: Bandwidth and latency
- Bandwidth (a.k.a. throughput): Bits transferred per time
- Latency (a.k.a. delay): Time for message to traverse network
- Memory measured in bytes, bandwith in bits per second
- Round trip times (RTTs) can be measured with “ping” (CLI)
- Protocol design impacts performance: Avoid needless RTTs
- More bandwidth cannot reduce the latency (= bottleneck)

# Layers

- Dealing with complex systems (for design and discussion)
- Modularization eases maintenance and updating of systems
- Each layer provides functionality in a transparent manner
- Layers take data from above and add header information
- Internet protocol stack:
  - Application Layer
  - Transport Layer
  - Internet Layer
  - Link Layer

# 2.2. Internet Layer

- Responsible for transporting packets between end systems
- Internet Protocol (IP) provides unreliable communication
- Relatively simple inside the network, complex at the “edge”
  - Series of local routing decisions without connection state: Packets between the same hosts may take different paths
  - Arbitrary network technologies (like Ethernet or Wi-Fi) can be used on the Link Layer, combined by the common IP
  - Networks may require the fragmentation of data packets, which get reassembled at the receiving host

# IP Address

- IP(v4) address: 32-bit identifier for host or router interface
- IP adresses reflect the Internet's hierarchical organization: Network (high order bits) and host part (low order bits)
- Local network: IP addresses of device interfaces, which can physically reach each other, have the same network part
- IP address is either hard-coded or assigned by the Dynamic Host Configuration Protocol (DHCP) ("plug-and-play")
- Today, 32-bit address space almost completely allocated
- Difficult transition to IPv6 that uses 128-bit addresses

# Routing

- Routing is the process of selecting a path for (data) delivery
- Each IP packet contains its source and destination address
- Routers forward packets based on internal routing tables, which store the fastest link to various network destinations
- Internet is a very dynamic network: Links are added and fail
- Network topology discovered by special routing protocols, e.g. propagate changes in distance to all neighboring nodes
- Packets have a limited time to live (TTL) to prevent loops
- Routing happens within (intra) and across (inter) networks

2.2. Internet Layer

# Local Area Network (LAN)

- Classification of computer networks by geographical scope
- Unicast delivers a message to a single specific node
- Broadcast delivers a message to all nodes in the network
- Broadcasting only possible in LANs but not in the Internet
- Media Access Control (MAC) addresses used on Link Layer
- Address Resolution Protocol (ARP) resolves an IP address to an interface's hard-wired MAC address by broadcasting
- Extension of networks by switches (packet forwarding on the Link Layer) and routers (connecting several networks)

# 2.3. Transport Layer

- Responsible for transporting packets between processes
- Two protocols with different goals: Transmission Control Protocol (TCP, p. 14) and User Datagram Protocol (UDP)

- UDP adds source and destination port numbers to packets
- UDP is connectionless (no setup of transmission channels)
- UDP packets can be lost, corrupted or arrive out of order: Implement the desired features in the Application Layer
- UDP supports broadcast, often used for streaming of media (dropping packets is preferable to waiting for delayed ones)

# Port Numbers

- The port number designates the process to which a host's operating system delivers incoming packets
- Register with OS to send and receive packets from a port
- It is a 16-bit number allowing 65'536 possible values
- First 1'024 port numbers reserved for widely-used services
- Use the associated port number to access a specific service
- Static port numbers on server side, dynamic on client side

2.3. Transport Layer

# Transmission Control Protocol

- TCP is a connection-oriented protocol that requires hand-shaking to set up a channel for end-to-end communication
- TCP provides reliable, in-order byte-stream data transfer (acknowledgements and retransmissions using sequence #)
- Packets are buffered at both sending and receiving process
- Includes flow control: Sender does not overwhelm receiver
- Includes congestion control: Sender slows down its sending rate when network is overloaded (too many packets lost)
- Denial of service attacks by spoofing requests for a server

# Network Address Translation

- NAT is an effort to alleviate the IPv4 address exhaustion
- Widely deployed in routers for home and small-office use
- All packets leaving the local network have same IP address as source; mapping maintained by translating port numbers
- Devices inside local network are no longer addressable and visible from the outside world (acts as a primitive firewall)
- Port forwarding allows to have servers behind NAT router
- NAT violates end-to-end connectivity (must be considered)

# Firewalls

- A firewall permits or denies network traffic based on rules
- Goal is to protect a local network from the outside world
- Attacks need vulnerabilities in both firewall and application
- Set of rules designed to distinguish between unauthorized access and legitimate communication (in both directions)
- Firewalls either software (part of OS) or hardware-based
- Filtering done on Internet, Transport or Application Layer
- Abuse of approved protocol for other purposes (tunneling)

# 2.4. Application Layer

- Application Layer protocols designed for specific purposes
- Use either TCP or UDP for host-to-host communication
- Many Application Layer protocols are text-based

- Two different architectures for applications:
- Client-server: Clients request services provided by servers
- Peer-to-peer: Equal peers are both suppliers and consumers

- Some applications employ a mixture of both (e.g. e-mail)
- Clients can be thin (terminal) or fat (independent functions)

# Domain Name System (DNS)

- Routing with IP addresses that are not suitable for humans: Readable domain names are easier to remember
- DNS acts as a distributed "phone book" to look up the IP addresses of domain names (a binary protocol over UDP)
- Scaling achieved by its hierarchical structure and caching
- A domain name is a sequence of labels separated by dots: www.ethz.ch is a top level domain with two subdomains
- Replies can include various IP addresses for load balancing
- Not only IP addresses, also other types of resource records

# E-Mail

- Mails delivered with Simple Mail Transfer Protocol (SMTP)
- Access to mailbox with other protocols (e.g. POP or IMAP)
- Text-based command/response interaction (TCP, port 25)
- Target host determined by mail exchange record of DNS
- Authentication is optional: Problem of spam messages
- Checking with reverse DNS lookup authenticity of sender
- Blacklisting or whitelisting of bad resp. good IP blocks
- Encryption of transmission only if server supports this
- Privacy legally protected by secrecy of correspondance

2.4. Application Layer

# Authentication and Encryption

- Spoofing: Forging of origin to impersonate someone else
- Sniffing: Intercepting and logging of traffic passing over a net
- The former requires authentication, the latter encryption

- Many protocols do not authenticate the source of a message
- Examples: IP/ARP spoofing, DNS cache poisoning, e-mail etc.
- Countermeasures: Use a protocol variant with Transport Layer Security (TLS) enabled and establish a Virtual Private Network (VPN) to an organization in untrusted networks
- (Asymmetric) cryptography is the topic of the third evening

# 2.5. World Wide Web

- The World Wide Web (WWW or the Web) is a system of interlinked hypertext documents accessed via the Internet
- A web browser renders web pages for display on monitors
- Proposed by Sir Tim Berners-Lee in 1989 at CERN in CH
- Non-proprietary technology using URI, HTTP and HTML
- A Uniform Resource Identifier (URI) identifies a resource
- A Uniform Resource Locator (URL) is a URI that specifies where and how a resource can be accessed on the Internet
  Syntax: scheme://domain:port/path?query_string#fragment
- A Unif. Res. Name (URN) is a URI that denotes a resource

# Hypertext Transfer Protocol

- More commonly known as HTTP (most URLs start with it)
- Web's Application Layer protocol (for document retrieval)
- A client initiates a TCP connection to a server on port 80
- HTTP is "stateless" (server maintains no information about past client requests), therefore no inconsistency problems
- Basic authentication supported (with name and password)
- Exchange of HTTP messages: "GET /index.html HTTP/1.1"
- Requests and responses contain headers about the host, user agent, accepted languages, time, cookies, caching etc.

# Hypertext Markup Language

- HTML is the predominant markup language for web pages
- HTML describes the content and structure but not layout
- HTML is text-based and uses tags like `<p>` for structuring
- Content between opening (`<p>`) and closing (`</p>`) tags
- Tags can have attributes: `<a href="target.html">`
- Various tags for headings, paragraphs, lists, links, quotes etc.
- Browsers interpret these tags and display pages accordingly
- Inspect pages with Inspector (Safari) or Firebug (Firefox)
- Pages rendered on your machine: Extensions like AdBlock

# HTML Example

```
<html>
  <head>
    <title>Title of web page</title>
  </head>
  <body>
    <h1>Heading</h1>
    <p>Paragraph with an example of a
    <a href="index.html">link</a>.</p>
    <p><img src="image.jpg"/></p>
  </body>
</html>
```

# Session Management

- HTTP is stateless, keeping track of a user's activity like login or shopping cart to support state is the task of the web site
- Three options for session management:
  - Cookies: Store small pieces of information (e.g. SessionID) at the client, which adds data to requests at same web site
  - URL encoding: Add a token to the query string of the URL (e.g. "page.html?sid=1234") and rewrite all internal links
  - Hidden form fields: Treat page requests like submissions of HTML forms and include the session state in hidden fields

# Cascading Style Sheets (CSS)

- CSS used to describe the layout of pages written in HTML
- Goal is to separate document content from its presentation
- Stored in the HTML header or a separate document, which enables multiple web pages to share the same formatting
- Allows different styles for different rendering (screen/print)
- Matching of style rules defining properties against elements:

```
p {
   font-family: "Helvetica";
   font-size: 12px;
   color: blue;
}
```

# JavaScript

- JavaScript is a scripting language that enables dynamic pages
- Executed on the client-side by the browser (as interpreter)
- Syntax similar to C and Java but nothing to do with them
- Introduced by Netscape (old browser) in 1995 (after www)
- JavaScript can modify a page's content, structure and layout
- Possibility of loading or submitting content after initial load
- Poses new vulnerabilities like cross-site scripting (XSS): JS code injection on foreign site giving the attacker full power

# Bookmarklets

- Possibility of storing JavaScript in a bookmark (not a hack)
- Bookmarklets add one-click functionality to the browser
- Example 1: Looking up the selected word (replace [url]):

```
javascript:window.open([url]+document.getSelection());
```

For dictionary with: `"http://www.dict.cc/?s="`

For Google with: `"http://www.google.ch/search?q="`

- Example 2: Editing the content of a page

```
javascript:with(document.body){void(contentEditable=
contentEditable==='true'?'false':'true')}
```

- More examples: https://www.squarefree.com/bookmarklets/

# 2.6. Semantic Web

- The Semantic Web is the vision that facilitates machines to understand the meaning of information on the WWW
- HTML standardizes syntax but not semantics of web pages
- Extend human-readable web pages with machine-readable metadata about the content and its relation to other pages
- Enable automated agents to perform tasks on users' behalf
- Though already proposed in 1999, its realization is nowhere
- Personal opinion: Too much emphasis on formats and ontologies instead of interactive protocols and user identities

# Implications

The implications of a Semantic Web would be as fundamental and transforming as those of the World Wide Web itself:

- No need for agencies between content providers and users
- Hence emancipation of users from proprietary platforms
- Intelligent aggregation of information for decision-making
- This kind of maturity might democratize whole economies
- Logic implemented at user-side and not delivered with data
- Difficulty to find adequate business models (no advertising)

2. Internet and Web

# Concepts Learned Today

- Caching (revisited)
- Client and Server
- Content and Style
- Identifiers and Locators
- Protocols and Layers
- Syntax vs. Semantics
- Transparency (again)

2. Internet and Web

# Clip of Today

Kevin Slavin: How algorithms shape our world (15:23)


www.ted.com/talks/kevin_slavin_how_algorithms_shape_our_world.html