Security in communications and networks

PA018

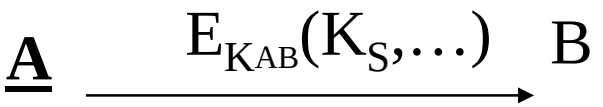
Vašek Matyáš

Major security enablers – critical (infrastructure) applications

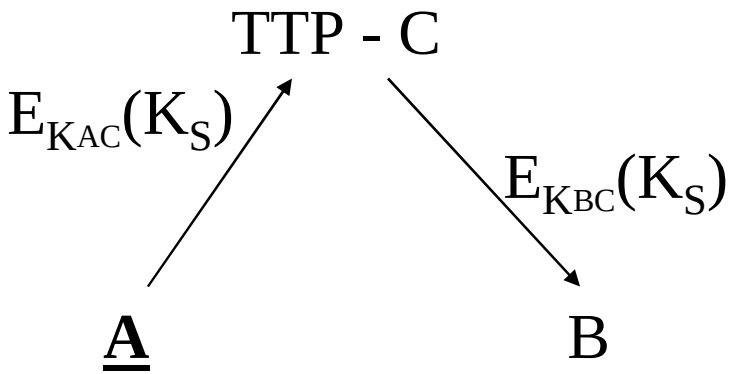
- Kerberos
- Public-key crypto based – certificates, typically X.509 – SSH, SSL/TLS
- Shared-key crypto based – symmetric key ciphers, hash functions

Key distribution (with indirect authentication)

- Direct distribution

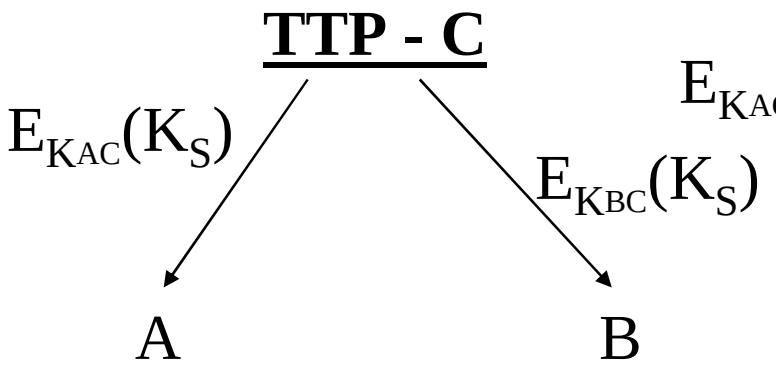


- Key distribution center (also generates the key – following slide)
- Key transport center

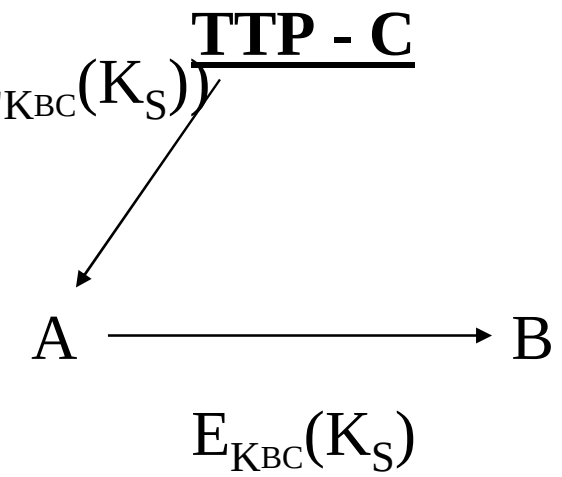


Indirect authentication – key distribution topologies.

TTP-managed



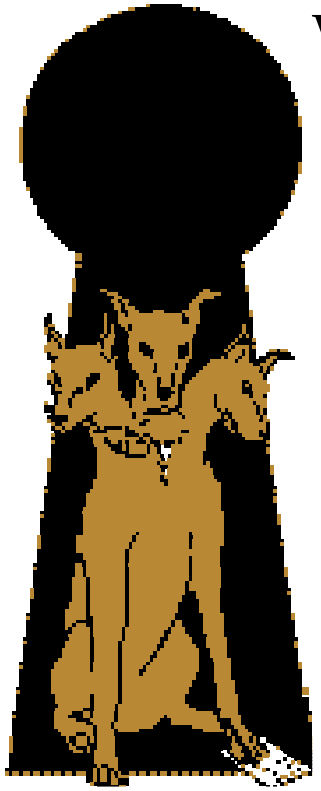
Direct (pull/push)



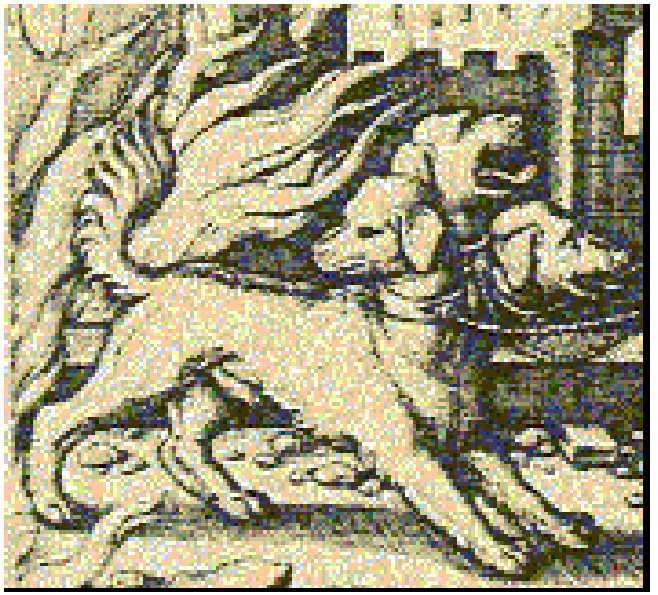
Kerberos

• Greek mythology – guardian to the entrance of Hades (master of the Underground)

• MIT project Athena – MIT's UNIX-based campus-wide academic computing facility



M12.1 Kerberos & Herakles



Kerberos – threat model

- Users reading messages of other users
- Users replaying messages of other users
- Users altering a workstation network address
- Users impersonating themselves

Kerberos – approach

- Centralised authentication server authenticating both users and machines
- Using symmetric-key techniques, no public-key techniques

Kerberos

- Trusted third-party authentication service
- Key Distribution Center (KDC) grants authentication tokens (“tickets”) to users
  - Trusted, dedicated machine
- Applications can use Kerberos for:
  - Data authentication
  - Data integrity
  - Data confidentiality

Kerberos used for applications

- telnet, rlogin, rcp, FTP, etc.
- Use Kerberos Protocol to exchange authentication information
- Client application uses Ticket-Granting-Ticket to obtain service tickets from KDC
- May use session key to encrypt data checksums (data integrity) or encrypt data (data confidentiality)

Kerberos – simple authentication

- $C \rightarrow AS$: $ID_C \,||\, P_C \,||\, ID_V$
- $AS \rightarrow C$: Ticket
- $C \rightarrow V$: $ID_C \,||\, \text{Ticket}$

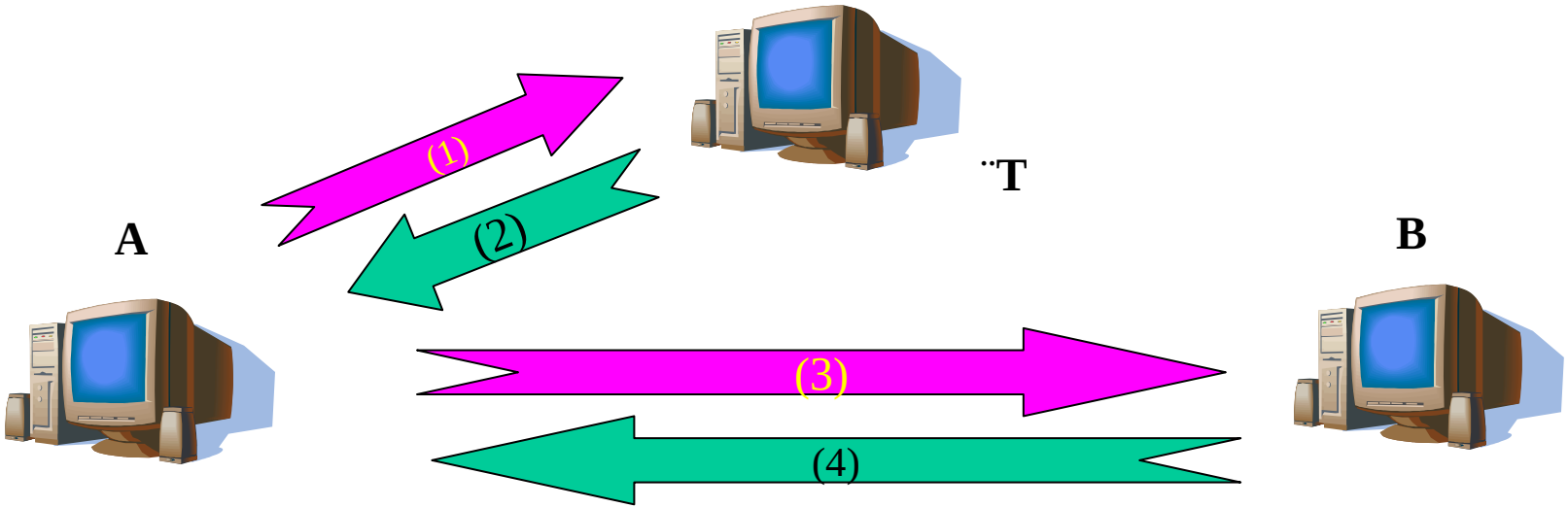
$\text{Ticket} = E_{Kv}(ID_C \,||\, P_C \,||\, ID_V)$

Kerberos Tickets

- Ticket-Granting-Ticket
  - Used to obtain further tickets
  - Requires password or additional authentication from user
  - Lifetime in hours
- Service Tickets
  - Issued to user from KDC
  - User can not decrypt ticket
  - User passes ticket to authenticate to server

Kerberos

- Simplified version of the protocol
  - L – ticket lifetime
  - Def.: $ticket_B = E_{K_{BT}}(k, \text{“A”}, L)$, $auth = E_k(\text{“A”}, T_A)$
  - (1) $A \rightarrow T$: ”A”, ”B”, n_A
  - (2) $A \leftarrow T$: $ticket_B$, $E_{K_{AT}}(k, n_A, L, \text{“B“})$
  - (3) $A \rightarrow B$: $ticket_B$, auth
  - (4) $A \leftarrow B$: $E_k(T_A)$



Kerberos Tickets (Credentials)

- Partly encrypted data structures
  - client ID
  - server ID
  - timestamp
  - session key
  - encrypted part (session key, client info, timestamp)
- Passed the way KDC $\rightarrow$ client $\rightarrow$ server
- Encrypted with the key of intended recipient

Kerberos – important terms

- C = Client
- AS = authentication server
- V = server
- ID_C = identifier of user on C
- ID_V = identifier of V
- P_C = password of user on C
- AD_C = network address of C
- K_V = secret encryption key shared by AS an V
- TS = timestamp
- || = concatenation

Kerberos – time vs. replay issue

- The threat: an opponent steals a ticket and uses it before its expiry time
- Lifetime of the ticket-granting ticket
  - Too short $\Rightarrow$ frequent ticket requests
  - Too long $\Rightarrow$ greater risk of replay attack

Tickets

- Ticket-Granting Ticket – *get once per logon*
- Service-Granting Ticket – *get then once before first use of a service (usually in a given logon session)*

- Authenticated Service Request – *once per (service) session*

Kerberos(v4) Authentication Process

Authentication Service Exhange – To obtain the Ticket-Granting Ticket

1) $C \rightarrow AS$: $ID_C \| ID_{TGS} \| TS_1$

2) $AS \rightarrow C$: $E_{Kc}(K_{C,TGS} \| ID_{TGS} \| TS_2 \| Lifetime_2 \| Ticket_{TGS})$

Ticket-Granting Service Echange – To obtain the Service-Granting Ticket

3) $C \rightarrow TGS$: $ID_V \| Ticket_{TGS} \| Authenticator_C$

4) $TGS \rightarrow C$: $E_{Kc}(K_{C,V} \| ID_V \| TS_4 \| Ticket_V)$

Client/Server Authentication Exhange: To Obtain Service

5) $C \rightarrow V$: $Ticket_V \| Authenticator_C$

6) $V \rightarrow C$: $E_{Kc,v}(TS_5 + 1)$

Kerberos today

- **Currently two broadly used versions:**
- 4 - restricted to a single realm (domain)
- 5 - allows inter-realm authentication
- Kerberos v5 is an Internet standard (RFC1510)
- MSFT implementation (since Windows 2000)

X.509 based authentication

- X.509 specifies the format for public-key certificates.
- The certificate contains the public key of a user and is signed with the private key of a Certification Authority (CA).
- Distributed environment using a database with certificate (user) information.
- Used in S/MIME, IP Security, SSL/TLS, SET.

What is SSL/TLS?

Secure Sockets Layer / Transport Layer Security

- Protocols providing security and reliability
- Protecting communication of two applications
- Running over standard protocols like TCP
- SSL – developed by Netscape, supported also by Microsoft…
- TLS – IETF standard (sometimes called SSL v3.1)
- Transparent for higher-level protocols like HTTP
- Using PKI and X.509 certificates

What security SSL/TLS provide?

Three basic security services:

- **Entity authentication –** the entities are authenticated using server and client certificates.
- **Integrity –** message authentication code (MAC) which ensures the data received is same as the data sent.
- **Confidentiality –** after the initial "handshake", a symmetric key is defined and used to encrypt all subsequent communication (even checked passwords, etc.).

Concepts of SSL/TLS

- Record Protocol
  - –The basic layer of the protocol.
  - –Works over TCP/IP (or other transport protocol).
  - –Allows for encapsulation of different higher level protocols (HTTP, FTP, telnet, etc.) which run unmodified.
- Handshake Protocol
  - –Allows the server and client to authenticate each other.
  - –By default, server authentication is mandatory, client authentication optional.
  - –Authentication through presentation of digital certificates.
    - And verification of the ability to use the related private key!

… more detail

- Establish Session
  - Send random challenge value, accept public key.
  - Verify signed challenge.
  - Deliver session key protected by recipient's public key.
- Communicate Protected Data
  - Encrypt data using agreed cipher and the session key.
  - Produce hash regularly to protect integrity.
  - Data packed into sequenced records.
- (Change Cipher - optional)
- Finish Session

- *http://www.ietf.org/rfc/rfc2246.txt*

Typical network attacks

- Holes in software (sendmail, RPC, NFS, or the firewall itself ☺)
- Network snooping – search for gold
- IP/DNS spoofing – masquerade
- Holes in new (higher-level) protocols
- Denial of Service (or even DDoS)
- …

Intrusion Detection Systems

- Intrusion – activity aimed at disrupting or circumventing a service within an organization’s system
- Also penetration, breach (, attack)
- Social engineering ☺
- Technical methods

IDS Principles

- Anomaly detection
  - Unusual pattern (as compared to typical user/system behavior).
  - False positives!
- Misuse detection
  - Pattern of intrusion(-like) behavior
  - False negatives!

Combine these two approaches!

IDS Topologies

Network-based

- Checking network traffic
- Use raw network packets.
- Typically a network adapter running in promiscuous - monitoring and analyzing all traffic.
- Responses like admin notification, connection termination, session recording (for forensic analysis), other detailed evidence collection.

Host-based

- Checking machines (log files, etc.).
- Started in 80s – log file review.
- Typically monitor system, event, and security logs on WinNT and syslog on Unix.
- Also critical file checksum control, response time, port activities.
- Responses analogous…

Combine these two approaches!

Email Security

- Postcard-like service

- PGP (Pretty Good Privacy)
- S/MIME (Secure Multipurpose Internet Mail Extension)
- (X.400)

S/MIME messages

- Combinations of two separately defined formats
  - (1) MIME entities
  - (2) Cryptographic Message Syntax (CMS) objects
- S/MIME entity formats
  - **one** for **enveloped** (i.e., encrypted) **–** provides confidentiality and key distribution services
  - **two** for **signed –** each provides integrity and data origin authentication services
  - **nested combinations** of signed and encrypted formats
  - may nest in any order to any “reasonable” depth
  - multiple nesting is used to construct S/MIME Enhanced Security Services

S/MIME version 2

- RFC 2311 – *S/MIME Version 2 Message Specification*, which is based on . . .
- RFC 2315 – *PKCS #7: Cryptographic Message Syntax Version 1.5*
  - Public-Key Cryptography Standards (PKCS): specifications begun in 1991 by RSA Laboratories and other industry and academic participants
  - PKCS #7: a general syntax for data that may have cryptography applied to it, e.g., digital signatures
  - defines a "**digital envelope** for a recipient" :
    (1) data encrypted in a content encryption key (CEK)
    (2) CEK encrypted in a second key, known to the recipient

S/MIME version 3

- RFC 2633 – *S/MIME Version 3 Message Specification*, which is based on the following:
- RFC 2630 – *Cryptographic Message Syntax*
  - enhancements to PKCS #7
  - adds attribute certificates, key agreement methods
  - adds encapsulation syntax for data protection
  - adds multiple, nested encapsulations
- S/MIME uses three of the CMS data types
  - enveloped data
  - signed data
  - just plain data
- S/MIME adds signed and unsigned attributes

IPSEC

- Authentication Header (AH), RFC-1826
  - Authenticity & integrity
- Encapsulating Security Payload (ESP), RFC-1827
  - Confidentiality (non-repudiation of origin)
  - Tunneling mode (encapsulation incl. headers)
  - Transport mode (data encapsulation)
- Security Associations (SA)

IPSEC – Security Associations

- Set of security features for a given session between two or more systems.
- Identifiable by Security Parameter Index (SPI) and the IP address.
- SPI depends on its Domain of Interpretation (DOI), this defines format, type of key-exchange, naming conventions, etc. One system can support more DOIs.

Parameters of Security Associations

- For AH – key authentication alg.
- For ESP – encryption alg., crypto synchronization, initiation vector.
- Both for AH and ESP – level of security, key lifetime, support of certificates, etc.

Firewalls

- Protect against attacks from the outside (across the firewall)
- Attacks against internal data
- Denial-of-service attacks
- Communication options:
  1. Allow
  2. Deny
  3. Translate (Proxy)

Basic options – firewalls

TCP/UDP	Allow/Deny	Packet filtering *(routers)*
TCP	A/D/Translate	Circuit-gateway *(trust inside)*
HTTP, FTP…	A/D/T	Applic.-gateway

Secure SHell

- SSH
- http://www.ssh.com/
- Non-commercial downloads

- WinSCP
- http://winscp.sourceforge.net/eng/
- WinSCP

Closely related topics – to be discussed later.

- Firewalls and network security
  - Guest lecture next week – Josef Pojsl, Technical Director, Trusted Network Solutions

Course reading – week 6

- *The Evolution of the Kerberos Authentication System* – Kohl, Neumann, Ts'o; 1991

`ftp://athena-dist.mit.edu/pub/kerberos/doc/krb_evol.PS`

  - Limitations of Kerberos 4, and changes made in Kerberos 5.

Questions?

Term project presentations April 17 & 24!
(And then May 15, of course. ☺)

Schedule follows…

Term project presentations!!!

April 17:

- **Barányi**
- **Halabica**
- **Červenka**
- **Drašar**
- **Folkman**
- **Gerguri**

April 24:

- **Benkovský**
- **Henzl**
- **Hubr**
- **Hulán**
- **Kocian**
- **Ondrák**

May 15:

- **Ashurova**
- **Honus**
- **Puškár**
- **Soběslavský**
- **Svoboda**
- **Synak**
- **Štverák**

Reminders: the presentation is worth (up to) 5 points from your course score; it should last at most 15 minutes (time for questions & discussion will be provided); laptop with AcroRead and PowerPoint will be available. ***Rehearse!!!!!***