Access Control and Privacy Policies (7)

Email: christian.urban at kcl.ac.uk
Office: S1.27 (1st floor Strand Building)
Slides: KEATS (also homework is there)

Recall the following scenario:

- If Admin says that file should be deleted, then this file must be deleted.
- Admin trusts Bob to decide whether file should be deleted (delegation).
- Bob wants to delete file.

$$\Gamma = \begin{array}{l} (\text{Admin says del\_file}) \Rightarrow \text{del\_file}, \\ (\text{Admin says } ((\text{Bob says del\_file}) \Rightarrow \text{del\_file})), \\ \text{Bob says del\_file} \end{array}$$

$$\Gamma \vdash \text{del\_file}$$

The Access Control Problem

Access Policy (Γ)

access request (F)

AC-Checker: applies inference rules

provable/ not provable

- $\boldsymbol{P}$ says $\boldsymbol{F}$ means $\boldsymbol{P}$ can send a “signal” $\boldsymbol{F}$ through a wire, or can make a “statement” $\boldsymbol{F}$

- P says F means P can send a "signal" F through a wire, or can make a "statement" F

- P is entitled to do F
  $P \text{ controls } F \stackrel{\text{def}}{=} (P \text{ says } F) \Rightarrow F$

$$\frac{\Gamma \vdash P \text{ controls } F \qquad \Gamma \vdash P \text{ says } F}{\Gamma \vdash F}$$

Security Levels

- Top secret (*TS*)
- Secret (*S*)
- Public (*P*)

$$slev(P) < slev(S) < slev(TS)$$

Security Levels

- Top secret (TS)
- Secret (S)
- Public (P)

$$slev(P) < slev(S) < slev(TS)$$

- Bob has a clearance for “secret”
- Bob can read documents that are public or sectret, but not top secret

Reading a File

$$\frac{\text{Bob controls Permitted (File, read)} \quad \text{Bob says Permitted (File, read)}}{\text{Permitted (File, read)}}$$

Reading a File

$$\frac{\begin{array}{l} slev(\text{File}) < slev(\text{Bob}) \Rightarrow \\ \qquad\qquad \text{Bob controls Permitted (File, read)} \\ \text{Bob says Permitted (File, read)} \\ slev(\text{File}) < slev(\text{Bob}) \end{array}}{\text{Permitted (File, read)}}$$

Reading a File

$$\frac{\begin{array}{l} \mathit{slev}(\text{File}) < \mathit{slev}(\text{Bob}) \Rightarrow \\ \qquad\qquad \text{Bob controls Permitted (File, read)} \\ \text{Bob says Permitted (File, read)} \\ \mathit{slev}(\text{File}) = P \\ \mathit{slev}(\text{Bob}) = S \\ \mathit{slev}(P) < \mathit{slev}(S) \end{array}}{\text{Permitted (File, read)}}$$

Substitution Rule

$$\frac{\Gamma \vdash \boldsymbol{slev(P) = l_1} \quad \Gamma \vdash \boldsymbol{slev(Q) = l_2} \quad \Gamma \vdash \boldsymbol{l_1 < l_2}}{\Gamma \vdash \boldsymbol{slev(P) < slev(Q)}}$$

Substitution Rule

$$\frac{\Gamma \vdash \boldsymbol{slev}(\boldsymbol{P}) = \boldsymbol{l}_1 \quad \Gamma \vdash \boldsymbol{slev}(\boldsymbol{Q}) = \boldsymbol{l}_2 \quad \Gamma \vdash \boldsymbol{l}_1 < \boldsymbol{l}_2}{\Gamma \vdash \boldsymbol{slev}(\boldsymbol{P}) < \boldsymbol{slev}(\boldsymbol{Q})}$$

- $\boldsymbol{slev}(\mathrm{Bob}) = \boldsymbol{S}$
- $\boldsymbol{slev}(\mathrm{File}) = \boldsymbol{P}$
- $\boldsymbol{slev}(\boldsymbol{P}) < \boldsymbol{slev}(\boldsymbol{S})$

Reading a File

$$\frac{\begin{array}{l} slev(\text{File}) < slev(\text{Bob}) \Rightarrow \\ \qquad\qquad \text{Bob controls Permitted (File, read)} \\ \text{Bob says Permitted (File, read)} \\ slev(\text{File}) = P \\ slev(\text{Bob}) = TS \\ ? \end{array}}{\text{Permitted (File, read)}}$$

Reading a File

$$
\begin{array}{l}
\textit{slev}(\text{File}) < \textit{slev}(\text{Bob}) \Rightarrow \\
\qquad\qquad \text{Bob controls Permitted (File, read)} \\
\text{Bob says Permitted (File, read)} \\
\textit{slev}(\text{File}) = \boldsymbol{P} \\
\textit{slev}(\text{Bob}) = \boldsymbol{TS} \\
\textit{slev}(\boldsymbol{P}) < \textit{slev}(\boldsymbol{S}) \\
\textit{slev}(\boldsymbol{S}) < \textit{slev}(\boldsymbol{TS}) \\
\hline
\text{Permitted (File, read)}
\end{array}
$$

Transitivity Rule

$$\frac{\Gamma \vdash \boldsymbol{l_1} < \boldsymbol{l_2} \qquad \Gamma \vdash \boldsymbol{l_2} < \boldsymbol{l_3}}{\Gamma \vdash \boldsymbol{l_1} < \boldsymbol{l_3}}$$

- $\boldsymbol{slev}(\boldsymbol{P}) < \boldsymbol{slev}(\boldsymbol{S})$
- $\boldsymbol{slev}(\boldsymbol{S}) < \boldsymbol{slev}(\boldsymbol{TS})$

  $\boldsymbol{slev}(\boldsymbol{P}) < \boldsymbol{slev}(\boldsymbol{TS})$

Reading Files

- Access policy for Bob for reading

$$\frac{\begin{array}{l}\forall f.\ slev(f) < slev(\text{Bob}) \Rightarrow \\ \qquad\qquad \text{Bob controls Permitted } (f, \text{read}) \\ \text{Bob says Permitted (File, read)} \\ slev(\text{File}) = P \\ slev(\text{Bob}) = TS \\ slev(P) < slev(S) \\ slev(S) < slev(TS)\end{array}}{\text{Permitted (File, read)}}$$

Reading Files

- Access policy for Bob for reading

$$
\begin{array}{l}
\forall f.\ slev(f) \leq slev(\text{Bob}) \Rightarrow \\
\qquad\qquad\qquad \text{Bob controls Permitted } (f, \text{read}) \\
\text{Bob says Permitted (File, read)} \\
slev(\text{File}) = TS \\
slev(\text{Bob}) = TS \\
slev(P) < slev(S) \\
slev(S) < slev(TS) \\
\hline
\qquad\qquad \text{Permitted (File, read)}
\end{array}
$$

Writing Files

- Access policy for Bob for **writing**

$$\frac{\begin{array}{l}\forall f.\ slev(\mathrm{Bob}) \leq slev(f) \Rightarrow \\ \qquad\qquad \mathrm{Bob\ controls\ Permitted}\ (f, \mathrm{write}) \\ \mathrm{Bob\ says\ Permitted}\ (\mathrm{File}, \mathrm{write}) \\ slev(\mathrm{File}) = TS \\ slev(\mathrm{Bob}) = S \\ slev(P) < slev(S) \\ slev(S) < slev(TS)\end{array}}{\mathrm{Permitted}\ (\mathrm{File}, \mathrm{write})}$$

Encrypted Messages

- Alice sends a message m

$$\text{Alice says } m$$

Encrypted Messages

- Alice sends a message m

$$\text{Alice says } m$$

- Alice sends an encrypted message m (with key K)

$$\text{Alice says } \{m\}_K$$

Encrypted Messages

- Alice sends a message m

$$\text{Alice says } m$$

- Alice sends an encrypted message m (with key K)

$$\text{Alice says } \{m\}_K$$

- Decryption of Alice's message

$$\frac{\Gamma \vdash \text{Alice says } \{m\}_K \quad \Gamma \vdash \text{Alice says } K}{\Gamma \vdash \text{Alice says } m}$$

Encryption

- Encryption of a message

$$\frac{\Gamma \vdash \text{Alice says } m \quad \Gamma \vdash \text{Alice says } K}{\Gamma \vdash \text{Alice says } \{m\}_K}$$

Trusted Third Party

Simple protocol for establishing a secure connection via a mutually trusted 3rd party (server):

Message 1 $A \rightarrow S : A, B$
Message 2 $S \rightarrow A : \{K_{AB}\}_{K_{AS}}$ and $\{\{K_{AB}\}_{K_{BS}}\}_{K_{AS}}$
Message 3 $A \rightarrow B : \{K_{AB}\}_{K_{BS}}$
Message 4 $A \rightarrow B : \{m\}_{K_{AB}}$

Sending Rule

$$\frac{\Gamma \vdash P \text{ says } F \quad \Gamma \vdash P \text{ sends } Q : F}{\Gamma \vdash Q \text{ says } F}$$

Sending Rule

$$\frac{\Gamma \vdash P \text{ says } F \quad \Gamma \vdash P \text{ sends } Q : F}{\Gamma \vdash Q \text{ says } F}$$

$$\begin{aligned} &P \text{ sends } Q : F \stackrel{\text{def}}{=} \\ &\quad (P \text{ says } F) \Rightarrow (Q \text{ says } F) \end{aligned}$$

Trusted Third Party

$$
\begin{array}{l}
A \text{ sends } S : \text{Connect}(A, B) \\
S \text{ says } (\text{Connect}(A, B) \Rightarrow \\
\qquad\qquad \{K_{AB}\}_{K_{AS}} \wedge \{\{K_{AB}\}_{K_{BS}}\}_{K_{AS}}) \\
S \text{ sends } A : \{K_{AB}\}_{K_{AS}} \wedge \{\{K_{AB}\}_{K_{BS}}\}_{K_{AS}} \\
A \text{ sends } B : \{K_{AB}\}_{K_{BS}} \\
A \text{ sends } B : \{m\}_{K_{AB}}
\end{array}
$$

Trusted Third Party

$$
\begin{aligned}
&A \text{ sends } S : \text{Connect}(A, B)\\
&S \text{ says } (\text{Connect}(A, B) \Rightarrow\\
&\qquad\qquad \{K_{AB}\}_{K_{AS}} \wedge \{\{K_{AB}\}_{K_{BS}}\}_{K_{AS}})\\
&S \text{ sends } A : \{K_{AB}\}_{K_{AS}} \wedge \{\{K_{AB}\}_{K_{BS}}\}_{K_{AS}}\\
&A \text{ sends } B : \{K_{AB}\}_{K_{BS}}\\
&A \text{ sends } B : \{m\}_{K_{AB}}
\end{aligned}
$$

$\Gamma \vdash B \text{ says } m$?

Public/Private Keys

- Bob has a private and public key: $K_{Bob}^{pub}, K_{Bob}^{priv}$

$$\frac{\Gamma \vdash \text{Alice says } \{m\}_{K_{Bob}^{pub}} \qquad \Gamma \vdash K_{Bob}^{priv}}{\Gamma \vdash \text{Alice says } m}$$

Public/Private Keys

- Bob has a private and public key: K^{pub}_{Bob}, K^{priv}_{Bob}

$$\frac{\Gamma \vdash \text{Alice says } \{m\}_{K^{pub}_{Bob}} \qquad \Gamma \vdash K^{priv}_{Bob}}{\Gamma \vdash \text{Alice says } m}$$

- this is **not** a derived rule!

Sending Rule

$$\frac{\Gamma \vdash P \; \textit{says} \; F \quad \Gamma \vdash P \; \textit{sends} \; Q : F}{\Gamma \vdash Q \; \textit{says} \; F}$$

Sending Rule

$$\frac{\Gamma \vdash P \; \mathit{says} \; F \quad \Gamma \vdash P \; \mathit{sends} \; Q : F}{\Gamma \vdash Q \; \mathit{says} \; F}$$

$$P \text{ sends } Q : F \stackrel{\text{def}}{=} \\ (P \text{ says } F) \Rightarrow (Q \text{ says } F)$$

Trusted Third Party

$$
\begin{array}{l}
A \text{ sends } S : \mathit{Connect}(A, B) \\
S \text{ says } (\mathit{Connect}(A, B) \Rightarrow \\
\qquad\qquad \{K_{AB}\}_{K_{AS}} \wedge \{\{K_{AB}\}_{K_{BS}}\}_{K_{AS}}) \\
S \text{ sends } A : \{K_{AB}\}_{K_{AS}} \wedge \{\{K_{AB}\}_{K_{BS}}\}_{K_{AS}} \\
A \text{ sends } B : \{K_{AB}\}_{K_{BS}} \\
A \text{ sends } B : \{m\}_{K_{AB}}
\end{array}
$$

Trusted Third Party

$$
\begin{array}{l}
A \text{ sends } S : \mathit{Connect}(A, B) \\
S \text{ says } (\mathit{Connect}(A, B) \Rightarrow \\
\qquad\qquad \{K_{AB}\}_{K_{AS}} \wedge \{\{K_{AB}\}_{K_{BS}}\}_{K_{AS}}) \\
S \text{ sends } A : \{K_{AB}\}_{K_{AS}} \wedge \{\{K_{AB}\}_{K_{BS}}\}_{K_{AS}} \\
A \text{ sends } B : \{K_{AB}\}_{K_{BS}} \\
A \text{ sends } B : \{m\}_{K_{AB}}
\end{array}
$$

$\Gamma \vdash B \text{ says } m$?

Challenge-Response Protocol

- an engine E and a transponder T share a key K
- E sends out a nonce N (random number) to T
- T responds with $\{N\}_K$
- if E receives $\{N\}_K$ from T, it starts engine

Challenge-Response Protocol

$$
\begin{array}{lr}
E \text{ says } N & \text{(start)} \\
E \text{ sends } T : N & \text{(challenge)} \\
(T \text{ says } N) \Rightarrow (T \text{ sends } E : \{N\}_K \wedge & \\
\qquad\qquad T \text{ sends } E : \mathrm{Id}(T)) & \text{(response)} \\
T \text{ says } K & \text{(key)} \\
T \text{ says } \mathrm{Id}(T) & \text{(identity)} \\
(E \text{ says } \{N\}_K \wedge E \text{ says } \mathrm{Id}(T)) \Rightarrow & \\
\qquad\qquad \text{start\_engine}(T) & \text{(engine)}
\end{array}
$$

$\Gamma \vdash \text{start\_engine}(T)$?

Exchange of a Fresh Key

A and B share a ("super-secret") key K_{AB} and want to share another key

- assumption K_{AB} is only known to A and B

- A sends B : $A, \{N_A\}_{K_{AB}}$
- B sends A : $\{N_A + 1, N_B\}_{K_{AB}}$
- A sends B : $\{N_B + 1\}_{K_{AB}}$
- B sends A : $\{K_{AB}^{new}, N_B^{new}\}_{K_{AB}}$

Assume K_{AB}^{new} is compromised by I

Exchange of a Fresh Key

A and B share a ("super-secret") key K_{AB} and want to share another key

- assumption K_{AB} is only known to A and B

- A sends B : $A, \{N_A\}_{K_{AB}}$
- B sends A : $\{N_A + 1, N_B\}_{K_{AB}}$
- A sends B : $\{N_B + 1\}_{K_{AB}}$
- B sends A : $\{K_{AB}^{new}, N_B^{new}\}_{K_{AB}}$
- A sends B : $\{msg\}_{K_{AB}^{new}}$

Assume K_{AB}^{new} is compromised by I

The Attack

An intruder I convinces A to accept the compromised key K_{AB}^{new}

- A sends B : $A, \{N_A\}_{K_{AB}}$
- B sends A : $\{N_A + 1, N_B\}_{K_{AB}}$
- A sends B : $\{N_B + 1\}_{K_{AB}}$
- B sends A : $\{K_{AB}^{new}, N_B^{new}\}_{K_{AB}}$ recorded by I

The Attack

An intruder I convinces A to accept the compromised key K_{AB}^{new}

- A sends B : $A, \{N_A\}_{K_{AB}}$
- B sends A : $\{N_A + 1, N_B\}_{K_{AB}}$
- A sends B : $\{N_B + 1\}_{K_{AB}}$
- B sends A : $\{K_{AB}^{new}, N_B^{new}\}_{K_{AB}}$ recorded by I
- A sends B : $A, \{M_A\}_{K_{AB}}$
- B sends A : $\{M_A + 1, M_B\}_{K_{AB}}$
- A sends B : $\{M_B + 1\}_{K_{AB}}$
- B sends I : $\{K_{AB}^{newer}, N_B^{newer}\}_{K_{AB}}$ intercepted by I
- I sends A : $\{K_{AB}^{new}, N_B^{new}\}_{K_{AB}}$

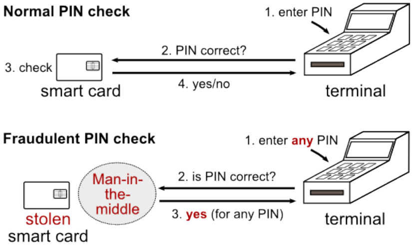
The Attack

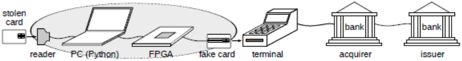
An intruder I convinces A to accept the compromised key K_{AB}^{new}

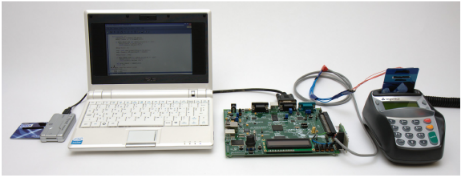
- A sends B : $A, \{N_A\}_{K_{AB}}$
- B sends A : $\{N_A + 1, N_B\}_{K_{AB}}$
- A sends B : $\{N_B + 1\}_{K_{AB}}$
- B sends A : $\{K_{AB}^{new}, N_B^{new}\}_{K_{AB}}$ recorded by I
- A sends B : $A, \{M_A\}_{K_{AB}}$
- B sends A : $\{M_A + 1, M_B\}_{K_{AB}}$
- A sends B : $\{M_B + 1\}_{K_{AB}}$
- B sends I : $\{K_{AB}^{newer}, N_B^{newer}\}_{K_{AB}}$ intercepted by I
- I sends A : $\{K_{AB}^{new}, N_B^{new}\}_{K_{AB}}$
- A sends B : $\{msg\}_{K_{AB}^{new}}$ I can read it also

A Man-in-the-middle attack in real life:

- the card only says yes or no to the terminal if the PIN is correct
- trick the card in thinking transaction is verified by signature
- trick the terminal in thinking the transaction was verified by PIN







Problems with EMV

- it is a wrapper for many protocols
- specification by consensus (resulted unmanageable complexity)
- its specification is 700 pages in English plus 2000+ pages for testing, additionally some further parts are secret
- other attacks have been found
- one solution might be to require always online verification of the PIN with the bank

Problems with WEP (Wifi)

- a standard ratified in 1999
- the protocol was designed by a committee not including cryptographers
- it used the RC4 encryption algorithm which is a stream cipher requiring a unique nonce
- WEP did not allocate enough bits for the nonce
- for authenticating packets it used CRC checksum which can be easily broken
- the network password was used to directly encrypt packages (instead of a key negotiation protocol)
- encryption was turned off by default

Protocols are Difficult

- even the systems designed by experts regularly fail
- try to make everything explicit (you need to authenticate all data you might rely on)
- the one who can fix a system should also be liable for the losses
- cryptography is often not **the** answer

logic is one way protocols are studied in academia (you can use computers to search for attacks)

Public-Key Infrastructure

- the idea is to have a certificate authority (CA)
- you go to the CA to identify yourself
- CA: "I, the CA, have verified that public key P^{pub}_{Bob} belongs to Bob"

- CA must be trusted by everybody
- What happens if CA issues a false certificate? Who pays in case of loss? (VeriSign explicitly limits liability to \$100.)