System Security, CS 5430

How to define security properties for a system

By Elisavet Kozyri

Building an airplane

- Functionality
  - Take-off, fly/navigate, land, …
- Assets
  - Passengers, pilot, machinery, …
- Threat model
  - Naughty children who want to fly the plane, …

- Security properties
  - The pilot should not be distracted by the passengers, …
- Enforcement mechanism
  - A high security door separates the pilot from passengers, …

Building an airplane

- Functionality
  - Take-off, fly/navigate, land, …
- Assets
  - Passengers, pilot, machinery, …
- Threat model
  - Naughty children who want to fly the plane, …

- **Security properties**
  - The pilot should not be distracted by the passengers, …
- Enforcement mechanism
  - A high security door separated pilot from passengers, …

Today

- Is there a **methodology** to **specify security properties** for a system?
- Detailed discussion about an imaginary example.
- Summary of the strategy we followed.
- Examination of a second example, lead by you.

Stork Baby Delivery

- System's brief description :
  - It allows a *stork* to deliver a *baby* to a geographic location prespecified by the *providence*.
- We **need** the specification of the system **functionality**.

Stork Baby Delivery

- The *stork* delivers a *baby* to a geographic location (coordinates) prespecified by the *providence*. Prior to take-off, providence programs the stork with the coordinates where the baby should be delivered. Throughout the mission, the stork transmits back to providence a video of the landscape , labeled with coordinates. While a stork is in flight, providence may issue commands to that stork and change the location for the delivery, alter the path being followed to that location, or abort the mission.

Stork Baby Delivery

- The *stork* delivers a *baby* to a geographic location (coordinates) prespecified by the *providence*. Prior to take-off, providence programs the stork with the coordinates where the baby should be delivered. Throughout the mission, the stork transmits back to providence a video of the landscape , labeled with coordinates. While a stork is in flight, providence may issue commands to that stork and change the location for the delivery, alter the path being followed to that location, or abort the mission.

Stork Baby Delivery

- **F1**: Prior to take-off, providence programs the stork with the coordinates where the baby should be delivered.
- **F2**: Throughout the mission, the stork transmits back to providence a video of the landscape , labeled with coordinates.
- **F3**: While a stork is in flight, providence may issue commands to that stork and change the location for the delivery, alter the path being followed to that location, or abort the mission.

Stork Baby Delivery

- **Asset**:
- **Threat model**:
  - **Goal**: The adversary desires to prevent baby deliveries.
  - **Methods**:
    - **T1**: Access to radio equipment that transmits and receives on the same frequencies that providence uses for communication with a stork.
    - **T2**: Use long nets to catch a stork in flight.

Stork Baby Delivery: Security Properties

What must be done and

what must not be done.

Stork Baby Delivery

Functions	Thread Model
• **F1**: Prior to take-off, prov programs coordinates to stork • **F2**: While flying, stork transmits to prov video with coordinates • **F3**: While a stork is in flight, prov commands stork to change delivery loc/path or abort	• **T1**: Intrude prov-stork radio communication. • **T2**: Catch stork with nets

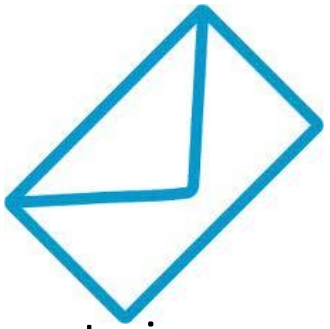
Security Properties

- I: Commands executed by stork are sent by the prov and they are the most recent.
- A: The stork will eventually execute prov's commands, if it is still in flight.
- C: The attacker should not know the coordinates.

A Methodology for finding security properties

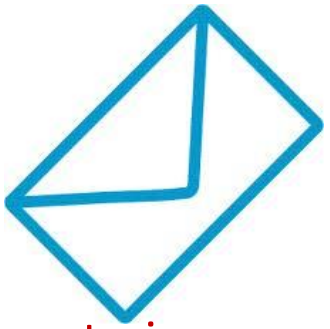
- Knowledge of system's functionality.
- Knowledge of system's assets.
  - Adversaries try to harm assets' value.
- Threat analysis.
  - Motivation, resources, capabilities
- Security Properties.
  - What should the behavior of the system be in the light of the threat model?
  - Properties may be relaxed after a feasibility analysis.
- Enforcement Mechanisms
  - The generality of a property should be mapped to specific system choices.
- Iteration

A Web-based Mail System



- Users login by visiting a prespecified URL for the system and then entering both an identifier (i.e., a name) and a password. This starts a session that is associated with the specified identity. The system then displays in a *preview frame* a list of messages that have been sent to that identity and have not been deleted during this or some prior session associated with that identity. Here, for each message, the name of the sender and the contents of the message are displayed.
- During a session, a user can:
  - Click on an icon to generate a reply to the message the user is currently viewing. The user then types the body of the reply. That reply later becomes a message that will be available for viewing by the sender of the original message to which this serves as a reply.
  - Click on an icon to generate a new message. The user then enters an identity of some receiver and enters a body for the message. That body is incorporated into a message that will be available for later viewing by the intended receiver.
  - Click on an icon to delete the message that the user is currently viewing.
  - Click on an icon to end the session.

A Web-based Mail System



- Users login by visiting a prespecified URL for the system and then entering both an identifier (i.e., a name) and a password. This starts a session that is associated with the specified identity. The system then displays in a *preview frame* a list of messages that have been sent to that identity and have not been deleted during this or some prior session associated with that identity. Here, for each message, the name of the sender and the contents of the message are displayed.
- During a session, a user can:
  - Click on an icon to generate a reply to the message the user is currently viewing. The user then types the body of the reply. That reply later becomes a message that will be available for viewing by the sender of the original message to which this serves as a reply.
  - Click on an icon to generate a new message. The user then enters an identity of some receiver and enters a body for the message. That body is incorporated into a message that will be available for later viewing by the intended receiver.
  - Click on an icon to delete the message that the user is currently viewing.
  - Click on an icon to end the session.

A Web-based Mail System

- **F1**: Users login by visiting a prespecified URL for the system and then entering both an identifier and a password.
- **F2**: This starts a session that is associated with the specified identity. The system then displays in a *preview frame* a list of messages that have been sent to that identity and have not been deleted during this or some prior session associated with that identity. Here, for each message, the name of the sender and the contents of the message are displayed.
- F3: Click on an icon to generate a reply to the message the user is currently viewing. The user then types the body of the reply.
- **F4**: That reply later becomes a message that will be available for viewing by the sender of the original message to which this serves as a reply.
- F5: Click on an icon to generate a new message. The user then enters an identity of some receiver and enters a body for the message.
- F6: That body is incorporated into a message that will be available for later viewing by the intended receiver.
- F7: Click on an icon to delete the message that the user is currently viewing.
- F8: Click on an icon to end the session.

A Web-based Mail System

- **Thread Model**:
  - **Goal**: The adversary is a user who desires to read email, generate bogus email, and/or alter email that has been generated by bona fide users.
  - **Methods**:
    - **T1**: The adversary has access to the URL for the mail system.
    - **T2**: The adversary can read, delete, and/or update network packets in transit.
    - The adversary cannot physically access or run programs on a user's machine that is running a browser to access the mail system.
    - And the adversary cannot physically access or run programs on the mail system server.

A Web-based Mail System

Functions	Thread Model
• **F1**: Login by visiting URL and enter identifier & password. • **F2**: Display sent messages and for each message display the name of the sender and its content. • **F4**: Reply message will be received by the original sender.	• **T1**: Access to URL • **T2**: Process packets in transit.

Security Properties
• C: No user can learn the password of other users. • A: The user can eventually log-in. • I: The website user sees is the genuine one.

Conclusion

- References
  - http://www.cs.cornell.edu/courses/CS5431/2011sp/lectures/requirements.php by Michael Clarkson
  - http://www.cs.cornell.edu/courses/CS5430/2012sp/hw.01.c1SecProps.html
  - http://www.cs.cornell.edu/courses/CS5430/2013sp/paper.chptr01.pdf