Dependability and Security

Attributes
- Availability
- Reliability
- Safety
- Confidentiality
- Integrity
- Maintainability

Threats
- Faults
- Errors
- Failures

Means
- Fault Prevention
- Fault Tolerance
- Fault Removal
- Fault Forecasting
**Safety** - avoidance of catastrophic consequences -
As a function of time, $S(t)$, is the probability that the system either behaves correctly or will discontinue its functions in a manner that causes no harm (operational or Fail-safe)

**Coverage** – The coverage is the measure $c$ of the system ability to reach a fail-safe state after a fault.

Modeling coverage in a Markov chain means that every unfailed state has two transitions to two different states, one of which is fail-safe, the other is fail-unsafe.
**MTTR** - The Mean Time To Repair is the average time required to repair the system. MTTR is expressed in terms of a repair rate \( m \) which is the average number of repairs that occur per time period, generally number of repairs per hours

\[
\mu = \frac{1}{MTTR}
\]

**Maintenability** - \( M(t) \) is the conditional probability that the system is repaired throughout the interval of time \([0, t]\), given that the system was faulty at time 0

\[
M(t) = 1 - e^{-\mu t}
\]

with \( \mu \) constant repair rate.
Security

Security is defined as resilience to malicious attacks

This can be viewed as computer systems failures due to intentional attacks

Causes of security violations are different from the causes of failures in hw or sw

- Failures are caused by human intent
- Failures are correlated
- Failures depend on subtle way on the system structure

Attackers learn over time
Survivability:

Capability of a system to fulfill its mission in a timely manner, in presence of attacks, failures or accidents

Survivability is related to the ability of the system to perform an intended function (modelling approaches related to reliability and availability can be applied)
Reliability in the face of system’s vulnerability and malicious attacks

Development of stochastic descriptions of events that may occur during a cyber attack
Probabilities in modelling cyber attacks
Stochastic models for computing measures
Availability in the face of system’s vulnerability and malicious attacks

Depends on
- attack’s own impact on the system
- effort to diagnose the attack
- restore system service following the attack
  - how long a system remains following a successful attack
• Safety under malicious attacks
  - safety depends on the effects of a system failure other than on the causes of failures
  - quantification of safety in the context of cyber attacks

Example: Denial of service cyber attack
  - Impact of that type of attack on system safety
Additional attribute of security not included in reliability and availability

- **Data confidentiality**
  protected data not read by unauthorized users

- **Data integrity**
  protected data not modified by unauthorized users

both attributes are not related to the functionality of the system

- **Nonrepudiation**
  Prevents future false denial of involvement by either party in a transaction

- **Authentication**
  The claimed identity of a party to a transaction can be independently verified
Security

Models for security analysis must describe

1. How and when security attacks occur
2. Impact of an attack on the system when it is executed successfully
3. Mechanisms, effects and costs of system recovery, system maintenance and defenses

There are differences with classical dependability
- In the nature and details of security models

Asset: information or resources that could be subject to attack
Example: Denial of service cyber attack
- Impact of that type of attack on system safety
- The system’s attempts to cope with it
  -> we can evaluate the time spent in states that reflect the attack
Examples of Security failure

*Misconfiguration* (at any level of the application stack: network service, web server, databases, virtual machine, ...) is a source of security vulnerability.

Failure due to misconfiguration can occur in many contexts but some external agent must deliberately exercise the vulnerability in order for the failure to occur.
Examples of Security failure

*Latent software faults* (e.g., buffer overflow problems) Are another cause of security failure

Any given fault has a specific behavior and requirements for accessing and exploiting it.

*Security penetration* made possible by latent sw fault does not occur accidentally but is induced by an attacker

A security penetration may require an attacker to exercise several vulnerabilities before compromising an asset
Microsoft Security Development Lifecycle (SDL)
Threat Modeling tool

The STRIDE threat model provides a way to methodically review system designs and highlight security threats.

STRIDE uses six security threat categories to review system design (developed at Microsoft):

<table>
<thead>
<tr>
<th>Threat</th>
<th>Desired property</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spoofing</td>
<td>Authenticity</td>
</tr>
<tr>
<td>Tampering</td>
<td>Integrity</td>
</tr>
<tr>
<td>Repudiation</td>
<td>Non-repudiability</td>
</tr>
<tr>
<td>Information disclosure</td>
<td>Confidentiality</td>
</tr>
<tr>
<td>Denial of Service</td>
<td>Availability</td>
</tr>
<tr>
<td>Elevation of Privilege</td>
<td>Authorization</td>
</tr>
</tbody>
</table>

“what can go wrong in this system we're working on?”

SDL report

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Threat Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ECU OBD:Response</td>
<td>Tampering</td>
</tr>
<tr>
<td>ECU OBD:Response</td>
<td>Information Disclosure</td>
</tr>
<tr>
<td>ECU OBD:Response</td>
<td>Denial Of Service</td>
</tr>
<tr>
<td>ECU Request:Mem Location:Request</td>
<td>Tampering</td>
</tr>
<tr>
<td>ECU Request:Mem Location:Request</td>
<td>Information Disclosure</td>
</tr>
<tr>
<td>ECU Request:Mem Location:Request</td>
<td>Denial Of Service</td>
</tr>
<tr>
<td>Mem Location:Request:Mem Location:Response</td>
<td>Tampering</td>
</tr>
<tr>
<td>Mem Location:Request:Mem Location:Response</td>
<td>Information Disclosure</td>
</tr>
<tr>
<td>Mem Location:Request:Mem Location:Response</td>
<td>Denial Of Service</td>
</tr>
<tr>
<td>OBD:ECU:Request</td>
<td>Tampering</td>
</tr>
</tbody>
</table>

Other approaches classify vulnerabilities and threats that may appear in general in a computer system.

PLOVER: Preliminary List Of Vulnerability Examples for Researchers identifies 28 specific Weaknesses, Idiosyncrasies, Faults and Flaws (WIFFs).

- Authentication error
- Buffer overflows
- Permissions, Privileges, Access Control List

......
Risk analysis

Also approaches for the risk analysis are applied to security threats

Risk analysis has been extensively applied in safety critical systems, using established techniques for quantitative evaluation of dependability, like Fault Trees and Failure Mode and Effects Analysis, Stochastic models.
Functional safety: the ability of the system to deliver the expected functionality during its operational life

The objective of functional safety is to reduce the probability of failures at a given acceptable rate in presence of malfunctioning behaviors.

In the hazard and risk analysis, hazardous events are identified and the necessary risk reduction for these events determined.

Tolerable risk: risk which is accepted in context based on the current values of society.
SAFURE project: System development and Safety analyses

Safety critical systems regulations

**IEC 61508:**  
*Functional Safety of Electrical/Electronic/Programmable Electronic Safety-related Systems*  
an international standard of rules for programmable systems applied in industry

ISO 26262: Road vehicles – Functional safety

adaptation of IEC 61508 specific to the application sector of electrical and electronic systems in automotive industry

**ISO 26262 is the established safety standard in automotive**

IEC - International Electrotechnical Commission

ISO - International Organization for Standardization
Problems caused by malicious attacks are not addressed by the hazard analysis and risk assessment within the ISO 26262 standard.

**Cyber-security as a risk factor to be considered in the hazard and risk analysis**

ISO/SAE CD 21434 -Road Vehicles –Cybersecurity engineering
https://www.iso.org/standard/70918.html (under development)

Schmittner et al., Towards a Framework for Alignment between Automotive Safety and Security Standards Conference Paper · September 2015
Risk assessment tools implement several risk assessment methodologies

- DREAD risk assessment method
- Common Vulnerability Scoring System (CVSS)
- OWASP Risk Rating Methodology
- SAHARA
DREAD risk assessment method

Categories of risk analysis

– **Damage potential**
  Ranks the extent of damage that occurs if a vulnerability is exploited

– **Reproducibility**
  Ranks how often an attempt at exploiting a vulnerability really works

– **Exploitability**
  Effort required to exploit the vulnerability (a number) e.g. authentication is considered

– **Affected users**
  Number of instances of the system that would be affected if an exploit became widely available

– **Discoverability** Measures
  The likelihood that a vulnerability will be found by hackers
DREAD risk assessment method

Risk = \frac{\text{Damage} + \text{Reproducibility} + \text{Exploitability} + \text{Affected Users} + \text{Discoverability}}{5}

Rating scale for each category: 0-10

1 being the least probability of the occurrence and the least damage potential
CVSS is comprised of three different metric groups: Base, Temporal, and Environmental. Each one consists of their own set of metrics.

<table>
<thead>
<tr>
<th>Base:</th>
<th>Temporal:</th>
<th>Environmental:</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Access Vector</td>
<td>- Exploitability</td>
<td>- Collateral damage potential</td>
</tr>
<tr>
<td>- Access Complexity</td>
<td>- Remediation Level</td>
<td>- Target distribution</td>
</tr>
<tr>
<td>- Authentication</td>
<td>- Report Confidence</td>
<td>- Confidentiality requirement</td>
</tr>
<tr>
<td>- Confidentiality Impact</td>
<td></td>
<td>- Integrity requirement</td>
</tr>
<tr>
<td>- Integrity Impact</td>
<td></td>
<td>- Availability requirement</td>
</tr>
<tr>
<td>- Availability Impact</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Common Vulnerability Scoring System (CVSS) - open framework

Risk = Base Metrics $f(x_1, x_2, \ldots, x_n)$

Refined by

Temporal Metrics $f(y_1, y_2, \ldots, y_n)$

Environmental Metrics $f(z_1, z_2, \ldots, z_n)$

Exploitability sub-score equation

Impact sub-score equation

severity posed by a vulnerability to a user’s environment at a specific point in time in a computing environment.

https://www.first.org/cvss/calculator/3.0
## CVSS: example

<table>
<thead>
<tr>
<th>Category</th>
<th>Subcategory</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Access Vector (AV)</strong></td>
<td>L (Local) accessible only on device</td>
<td>0.395</td>
</tr>
<tr>
<td></td>
<td>A (Adjacent network) accessible via directly attached bus</td>
<td>0.646</td>
</tr>
<tr>
<td></td>
<td>N (Network) accessible via any number of networks</td>
<td>1</td>
</tr>
<tr>
<td><strong>Authentication (Au)</strong></td>
<td>M (Multiple) multiple auth. steps</td>
<td>0.45</td>
</tr>
<tr>
<td></td>
<td>S (Single) one auth. step</td>
<td>0.56</td>
</tr>
<tr>
<td></td>
<td>N (None) No authentication is required</td>
<td>0.704</td>
</tr>
</tbody>
</table>

Score: 0 – 10
Open Web Application Security Project (OWASP) estimates both technical and business impact factors.

Starts from the standard risk model:

\[
\text{Risk} = \text{Likelihood} \times \text{Impact}
\]

The following methodology is defined, where factors for the likelihood and impact of each risk are considered.

https://owasp.org/
Step 1: Identify Risk

Step 2: Factors for estimating likelihood
- Threat Agent Factors
- Vulnerability Factors

Step 3: Factors for estimating impact
- Technical Impact Factors
- Business Impact Factors

Step 4: Determining severity of risk
- Informal Method
- Repeatable Method
- Determining Severity

Step 5: Deciding what to fix

Step 6: Customizing your risk rating model

**OWASP top 10 vulnerabilities in web applications**
SAHARA method allows the evaluation of the impact of security issues on safety at the system level.

Threats are **quantified** according to

- **Required Resources**
- **Know-How** that are required to define threats
- Threats **Criticality**

The impact of the threat on the system determines whether the threat is safety-related or not. If the threat is safety-related, it will be analysed and the resulting hazards will be evaluated.

### Security-Aware Hazard Analysis and Risk Assessment

**SAHARA**

<table>
<thead>
<tr>
<th>Level</th>
<th>Threat Criticality</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>no security impact</td>
<td>No security impact</td>
</tr>
<tr>
<td>1</td>
<td>Moderate security relevance</td>
<td>Reduced availability</td>
</tr>
<tr>
<td>2</td>
<td>High security relevance</td>
<td>non availability, privacy intrusion</td>
</tr>
<tr>
<td>3</td>
<td>High security and possibly safety relevance</td>
<td>Life threatening abuse possible</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Level</th>
<th>Required Know-How</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>no prior knowledge (black-box approach)</td>
<td>Unknown internals</td>
</tr>
<tr>
<td>1</td>
<td>Technical knowledge (gray-box approach)</td>
<td>Electrician, mechanic basic understanding of internals</td>
</tr>
<tr>
<td>2</td>
<td>Domain knowledge (white-box approach)</td>
<td>person with technical training, internal disclosed</td>
</tr>
</tbody>
</table>
Security-Aware Hazard Analysis and Risk Assessment
SAHARA

<table>
<thead>
<tr>
<th>Level</th>
<th>Required resource</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>no additional tool or everyday commodity</td>
<td>randomly using of user interface</td>
</tr>
<tr>
<td>1</td>
<td>standard tool</td>
<td>screwdriver, coin</td>
</tr>
<tr>
<td>2</td>
<td>simple tool</td>
<td>CAN sniffer, oscilloscope</td>
</tr>
<tr>
<td>3</td>
<td>advanced tool</td>
<td>debugger, bus communication simulator</td>
</tr>
</tbody>
</table>

Classification of hazards according to the matrix
4 is the highest security class

Security Level Determination matrix

<table>
<thead>
<tr>
<th>Required Resources 'R'</th>
<th>Required Knowledge 'K'</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0 0 1 2 3</td>
</tr>
<tr>
<td>2</td>
<td>0 0 1 2 3</td>
</tr>
<tr>
<td>1</td>
<td>0 2 3 4</td>
</tr>
<tr>
<td>2</td>
<td>0 1 2 3</td>
</tr>
<tr>
<td>3</td>
<td>0 1 2 3</td>
</tr>
</tbody>
</table>
Evaluation of Security

Combinatorial models
All basic events must be statistically independent
Do not model state - they model operational dependency of the system on the components

Reliability block diagrams: not used in security
Fault trees -> Attack trees

Attack trees:
used to explore a system to find possible vulnerabilities
Attack Trees

The tree describes sets of events that can lead to the goal in a combinatorial way.

Security of the system:
set of attack trees, where the root of each tree is the goal of an attacker that can damage the system operation

1. Root = goal of an attacker
2. Leaf nodes = different basic ways to achieve that goal (atomic attacks)
3. OR nodes = a node of which only one of its child nodes needs to be successful
4. AND nodes = a node of which all of its child nodes need to be successful

Attack tree published in [Buldas 20]
Evaluation of different aspects of the system security, depending on the kind of values assigned to the leaf nodes

Since an atomic attack can have multiple values, the attack tree can be used to combine these values and help users to learn more about a system’s vulnerability

Example
Possible/impossible, cost -> lowest possible cost attack

Example
probability, special equipment value -> most probable attack with no special equipment required

assign values to leaf nodes and propagate the node value up to the root
Evaluation of Security

Minimum cut-set -> set of atomic attacks that achieve a goal

S = {{Steal credit card, Shouldersurf PIN} 
{Hack online Bank account}}

Impact of certain atomic attacks on the overall system security

Attack Trees: systematic ways to describe system vulnerability, making possible to assess risks and making security decisions

Attack trees: reusable as part of a larger attack tree for a system
Each attack method will be based on a logical combination (AND/OR) of attacks against one or more “assets” populating the lowest levels of the attack tree.

Probability of success can be estimated for asset attacks

Reference architecture

In-vehicle network structure

- Powertrain PTC
  - Engine Control
  - Hybrid Drive
  - Transmission
  - PT Sensors

- Chassis & Safety CSC
  - Brake Control
  - Chassis / Steering
  - Environmental Sensors
  - Passive Safety
  - Chassis Sensors e.g. Steer Angle

- Body Electronic BEM
  - Instrument
  - Door Modules
  - Light Control
  - Climate
  - Seat ECU

- Head Unit HU
  - Audio
  - Display / Video
  - Navigation
  - Telephone

- Communication Unit CU

- GPS/Galileo
- UMTS
- DSRC

Diagnosis Interface

Bluetooth

USB

Mobile Device

Taken from [EVITA-D2.3]
Attack tree: Compromise driver privacy

Misuse the OBD updates or manipulate the CU to gain access to personal data.