More Powerful Safety and Cybersecurity Analysis

Prof. Nancy G. Leveson

Aeronautics and Astronautics
MIT
– You’ve carefully thought out all the angles
– You’ve done it a thousand times
– It comes naturally to you
– You know what you’re doing, it’s what you’ve been trained to do your whole life.
– Nothing could possibly go wrong, right?
Think Again

(Courtesy of Michael Brown)
General Definition of “Safety”

• **Accident = Mishap = Loss**: Any undesired and unplanned event that results in a loss
  – Including loss of human life or injury, property damage, environmental pollution, mission loss, loss of protected information, negative business impact (damage to reputation, etc.), etc.
  – Both inadvertent and intentional losses: includes cyber security

• System goals vs. constraints (limits on how can achieve the goals)

• Safety: Absence of losses
Current Analysis Techniques 50-70 years old

- Introduction of computer control
- Exponential increases in complexity
- New technology
- Changes in human roles

Assumes accidents caused by component failures

- FMEA
- FTA
- ETA
- HAZOP
- STPA/CAST

- Analog Helicopter Cockpit
- Personal Computers
- Computer Driven Helicopter Glass Cockpit
- Autonomy, Teaming, etc.
What Failed Here?

• Navy aircraft were ferrying missiles from one location to another.

• One pilot executed a planned test by aiming at aircraft in front and firing a dummy missile.

• Nobody involved knew that the software was designed to substitute a different missile if the one that was commanded to be fired was not in a good position.

• In this case, there was an antenna between the dummy missile and the target so the software decided to fire a live missile located in a different (better) position instead.
Warsaw A320 Accident

• Software protects against activating thrust reversers when airborne

• Hydroplaning and other factors made the software think the plane had not landed

• Pilots could not activate the thrust reversers and ran off end of runway into a small hill.
Two Types of Accidents

• **Component Failure Accidents**
  – Single or multiple component failures
  – Usually assume random failure

• **Component Interaction Accidents**
  – Arise in interactions among components
  – Related to complexity (coupling) in our system designs, which leads to design and system engineering errors
  – No components may have “failed”
  – Exacerbated by introduction of computers and software but the problem is system design errors
    • Software allows almost unlimited complexity in our designs
Unreliable but not unsafe
(FMEA)

Unreliable and unsafe
(FTA, HAZOP, FMECA, STPA …)

Unsafe but not unreliable
(STPA)

Scenarios involving failures

Unsafe scenarios

Confusing Safety and Reliability

Preventing Component or Functional Failures is Not Enough
Another Accident Involving Thrust Reversers

- Tu-204, Moscow, 2012
- Red Wings Airlines Flight 9268
- The soft 1.12g touchdown made runway contact a little later than usual.
- With the crosswind, this meant weight-on-wheels switches did not activate and the thrust-reverse system would not deploy.
Another Accident Involving Thrust Reversers

• Pilots believe the thrust reversers are deploying like they always do. With the limited runway space, they quickly engage high engine power to stop quicker. Instead this accelerated the Tu-204 forwards, eventually colliding with a highway embankment.
Another Accident Involving Thrust Reversers

- Pilots believe the thrust reversers are deploying like they always do. With the limited runway space, they quickly engage high engine power to stop quicker. Instead this accelerates the Tu-204 forwards, eventually colliding with a highway embankment.

In complex systems, human and technical considerations cannot be isolated
Human factors concentrates on the “screen out”

Hardware/Software engineering concentrates on the “screen in”
Not enough attention on integrated system as a whole

(e.g, mode confusion, situation awareness errors, inconsistent behavior, etc.)
Role of humans in systems is changing
Typical assumption is that operator error is cause of most incidents and accidents

- So do something about operator involved (admonish, fire, retrain them)

- Or do something about operators in general
  - Marginalize them by putting in more automation
  - Rigidify their work by creating more rules and procedures
Fumbling for his recline button Ted unwittingly instigates a disaster
A New Systems View of Operator Error

• Operator error is a symptom, not a cause

• All behavior affected by context (system) in which occurs
  – Role of operators is changing in software-intensive systems as is the errors they make
  – Designing systems in which operator error inevitable and then blame accidents on operators rather than designers

• To do something about operator error, must look at system in which people work:
  – Design of equipment
  – Usefulness of procedures
  – Existence of goal conflicts and production pressures

• Human error is a symptom of a system that needs to be redesigned
We need a paradigm change!

Prevent failures

Enforce constraints on behavior:
- components
- interactions among components

Treat Safety as a Reliability Problem

Treat Safety as a Control Problem
A Broad View of “Control”

Behavior can be controlled by human operators

or component failures and unsafe interactions can be “controlled” through system design
(e.g., redundancy, interlocks, fail-safe design)

or through process
  – Manufacturing processes
  – Maintenance processes
  – Test or assurance processes
  – Operational processes

or through social controls
  – Governmental or regulatory
  – Culture
  – Insurance
  – Law and the courts
  – Individual self-interest (incentive structure)
Controls/Controllers Enforce Safety Constraints

- Aircraft must maintain sufficient lift to remain airborne
- Two aircraft/automobiles must not violate minimum separation
- Public health system must prevent exposure of public to contaminated water, food products, and viruses
- Integrity of hull must be maintained on a submarine
- Toxic chemicals/radiation must not be released from plant
- Workers must not be exposed to workplace hazards
STAMP
(System-Theoretic Accident Model and Processes)

• A new, more powerful accident/loss causality model
• Based on systems theory, not reliability theory
• Defines accidents/losses as a dynamic control problem (vs. a failure problem)
• Applies to VERY complex systems
• Includes
  – Scenarios from traditional hazard analysis methods (failure events)
  – Component interaction accidents
  – Software and system design errors
  – Human/operator errors
  – Entire socio-technical system (not just technical part)
STAMP-Based vs. Traditional Analysis

STAMP-Based Analysis

Traditional Analysis

Scenarios

S₁ + S₂

S₁
STPA: System-Theoretic Process Analysis

• Identifies safety and security requirements and constraints

• Identifies scenarios leading to violation of constraints and requirements; use results to design or redesign system to be safer

• Finds hazardous design flaws in addition to failures

• Includes hardware, software, humans, organizational processes

• Supports test:
  – Test planning; what tests need to be done
  – Risk management
  – Investigating test abnormalities
Treating Safety as a Control Problem

- Controllers use a **process model** to determine control actions
- Software/human related accidents often occur when the process model is incorrect (inconsistent with real state of process)
- Captures software errors, human errors, flawed requirements ...
Warsaw (Reverse Thrusters)

Hazard: Inadequate aircraft deceleration after landing

Pilot

- Decision Making
- Process Model

Software Controller

- Control Algorithm
- Process Model

Aircraft
Warsaw (Reverse Thrusters)

Hazard: Inadequate aircraft deceleration after landing

1. **Pilot**
   - Decision Making
   - Process Model

2. **Software Controller**
   - Control Algorithm
   - Process Model

3. **Aircraft**

Feedback indicates plane has not landed
Warsaw (Reverse Thrusters)

Hazard: Inadequate aircraft deceleration after landing

Pilot
- Decision Making
- Process Model

Software Controller
- Control Algorithm
- Process Model

Aircraft

Plane has not landed
Feedback indicates plane has not landed
Hazard: Inadequate aircraft deceleration after landing

Pilot
- Decision Making
- Process Model

Software Controller
- Control Algorithm
- Process Model

Aircraft

Feedback indicates plane has not landed

Plane has not landed

Plane has landed
Warsaw (Reverse Thrusters)

Hazard: Inadequate aircraft deceleration after landing

- **Pilot**
  - Decision Making
  - Process Model

- **Software Controller**
  - Control Algorithm
  - Process Model

- **Aircraft**

Turn on reverse thrusters

Plane has landed

Plane has not landed

Feedback indicates plane has not landed
Warsaw (Reverse Thrusters)

**Hazard: Inadequate aircraft deceleration after landing**

- **Decision Making**
- **Process Model**

**Pilot**

**Software Controller**

- **Control Algorithm**
- **Process Model**

**Aircraft**

- Turn on reverse thrusters
- Ignore command
- Plane has landed
- Plane has not landed
- Feedback indicates plane has not landed
Moscow (Reverse Thrusters)

Hazard: Inadequate Deceleration after Landing

Pilot
- Decision Making
- Process Model

Software Controller
- Control Algorithm
- Process Model

Aircraft

Plane has landed
- Engage reverse thrust

Plane has not landed
- Ignore reverse thruster command

Feedback indicates plane has not landed
Moscow (Reverse Thrusters)

Hazard: Inadequate Deceleration after Landing

Aircraft

Software Controller

Process Model
Control Algorithm

Pilot

Decision Making
Process Model

Plane has landed
Reverse thrusters will come on
Plane has not landed
Feedback indicates plane has not landed

Engage reverse thrust
Ignore reverse thruster command

34
Moscow (Reverse Thrusters)

Pilot

- Decision Making
- Process Model

Software Controller

- Control Algorithm
- Process Model

Aircraft

Hazard: Inadequate Deceleration after Landing

- Plane has landed
- Reverse thrusters will come on
- Plane has not landed
- Feedback indicates plane has not landed

Short runway, need more power to stop

Engage reverse thrust

Ignore reverse thruster command

Engage reverse thrust
Moscow (Reverse Thrusters)

Hazard: Inadequate Deceleration after Landing

Pilot

Decision Making

Process Model

Software Controller

Control Algorithm

Process Model

Aircraft

Plane has landed

Reverse thrusters will come on

Plane has not landed

Feedback indicates plane has not landed

Short runway, need more power to stop

Engage reverse thrust

Engage high engine power

Ignore reverse thruster command

Engage reverse thrust

Short runway, need more power to stop

Engage high engine power
Hazard: Inadequate Deceleration after Landing

Pilot

- Decision Making
- Process Model

Software Controller

- Control Algorithm
- Process Model

Aircraft

- Plane has landed
- Reverse thrusters will come on
- Plane has not landed
- Feedback indicates plane has not landed

- Short runway, need more power to stop
- Engage reverse thrust
- Engage high engine power
- Ignore reverse thruster command
- Engage high engine power

Moscow (Reverse Thrusters)
How Did We Find This?

- We were doing a comparison/evaluation of STPA with SAE ARP 4761 (basically fault trees)
- We found this scenario
- Looked around web to find out if it had ever happened in operations
- Amazed to find many instances in different types of aircraft
- The problem is that standard hazard analysis cannot find it because there were no component “failures.”
Four types of unsafe control actions:
- Control commands required for safety are not given
- Unsafe ones are given
- Potentially safe commands given too early, too late
- Control stops too soon or applied too long

Analysis
- Identify potential unsafe control actions
- Identify why they might be (or were) given and then eliminate or mitigate them
- If safe ones provided, then why not followed?

(Leveson, 2003); (Leveson, 2011)
A NEW MODEL FOR HUMAN CONTROLLERS

Captures the controller’s goals and how decisions are made based on the mental models.

Captures specific types of flaws in the way the human controller conceptualizes the system and environment.

Captures the influence of human experiences, and expectations on the processing of sensory input.

(Thomas & France, 2016)
Example Analysis Results:
Some Hazards in Teaming Among Humans and Automation

- Conflicting Commands from Multiple Controllers
- Improper Handoff across Controller Boundary
- Human Machine Teaming Semantics
- Operator Workload Prohibits Teaming
- Incompatible System Configuration
- Mission too Dynamic to Effectively Plan
- Human Machine Teaming Trust
- Flawed Team Planning & Execution
- Operator(s)
- Controller
- UAS(s)
- AL
- AU
- AS
- Aircraft Software Enabled Controller
- Aircraft Subsystems
Integrated Approach to Safety and Security

- Both concerned with losses (intentional or unintentional)
  - Mission assurance (vs. information protection)
  - Ensure that critical functions and services are maintained
  - New paradigm for safety will work for security too
    - May have to add new causes, but rest of process is the same
  - A top-down, system engineering approach to designing safety and security into systems
### Some Current Uses

<table>
<thead>
<tr>
<th>Aviation</th>
<th>Maritime</th>
</tr>
</thead>
<tbody>
<tr>
<td>Automobiles</td>
<td>Dams</td>
</tr>
<tr>
<td>Defense</td>
<td>Manufacturing</td>
</tr>
<tr>
<td>Transportation</td>
<td>Communication</td>
</tr>
<tr>
<td>Healthcare/Hospitals</td>
<td>Rail</td>
</tr>
<tr>
<td>Workplace Safety</td>
<td>Robotics</td>
</tr>
<tr>
<td>Petrochemicals</td>
<td>Unmanned vehicles (air, ground, water)</td>
</tr>
<tr>
<td>Space</td>
<td>Process Control</td>
</tr>
<tr>
<td>Power and Energy</td>
<td>Mining</td>
</tr>
<tr>
<td>Nuclear</td>
<td>...</td>
</tr>
<tr>
<td>Agriculture</td>
<td></td>
</tr>
</tbody>
</table>

- 2,316 registrants for STAMP Workshop this summer from 73 countries
- Standards (auto, aircraft, defense) have been created or are in development for STPA
Evaluations and Estimates of ROI

• Hundreds of evaluations and comparison with traditional approaches used now
  – Controlled scientific and empirical (in industry)
  – All show STPA is better (identifies more critical requirements or design flaws)
  – Identified real accidents that other methods missed
  – All (that measured) show STPA requires orders of magnitude fewer resources than traditional techniques

• ROI estimates only beginning but one large defense industry contractor claims to have saved billions by using STPA
UH-60MU (Blackhawk)

- Analyzed Warning, Caution, and Advisory (WCA) system

- STPA results were compared with an independently conducted hazard analysis of the UH-60MU using traditional safety processes described in SAE ARP 4761 and MIL-STD-882E.
  - STPA found the same hazard causes as the traditional techniques and
  - Also identified many things not found using traditional methods, including design flaws, human behavior, and component integration and interactions
Navy Escort Vessels
(Lt. Blake Abrecht)

- Dynamic positioning system
- Ran into each other twice during test
- Performed a CAST analysis (on two incidents) and STPA on system as a whole
- STPA found scenarios not found by MIL-STD-882 analysis (fault trees and FMEA)
- Navy admiral rejected our findings saying “We’ve used PRA for 40 years and it works just fine”
- Put into operation and within 2 months ran into a submarine
- Scenario was one we had found
More Information

- [http://psas.scripts.mit.edu](http://psas.scripts.mit.edu) (papers, presentations from conferences, tutorial slides, examples, etc.)

Free download:
[http://mitpress.mit.edu/books/engineering-safer-world](http://mitpress.mit.edu/books/engineering-safer-world)

(STPA HANDBOOK)

NANCY G. LEVESON
JOHN P. THOMAS
MARCH 2018

Free download:
[http://psas.scripts.mit.edu](http://psas.scripts.mit.edu)
(100,000+ downloads in 30 mos. Japanese, Chinese, and Korean versions)

Free download: