STPA-Sec

stealing from safety engineering
to improve threat modeling

Journey: Aviation Safety

seattlepi

Search | seattlepi.com | Web Search by YAHOO! | Businesses

Home | Local | U.S./World | Business | Sports | A&E | Life | Comics | Photos | Blogs | Forums | Traff

Weather | Transportation/Traffic | Forums

It's never been safer to fly; deaths at record low

JOSHUA FREED, AP Airlines Writers, SCOTT MAYEROWITZ, AP Airlines Writers
Updated 11:12 a.m., Saturday, December 31, 2011

FILE - In this Dec. 23, 2011 file photo, travelers check their luggage at a United Airlines express check-in area at O'Hare International Airport in Chicago. Boarding an airplane has never been safer. In the last 10 years, there were 153 fatalities in U.S. airline crashes. That's 2 deaths for every 100 million passengers and the safest decade in the country's aviation history, according to an Associated Press analysis of government accident data. Photo: Nam Y. Huh / AP

NEW YORK (AP) — Boarding an airplane has never been safer.

The past 10 years have been the best in the country's aviation history with 153 fatalities. That's two deaths for every 100 million passengers on commercial flights, according to an Associated Press analysis of government accident data.

The improvement is remarkable. Just a decade earlier, at the time the safest, passengers were 10 times as likely to die when flying on an American plane. The risk of death was even greater during the start of the jet age, with 1,696 people dying — 133 out of every 100 million passengers — from 1962 to 1971. The figures exclude acts of terrorism.

The New School of Information Security

The Blog Inspired By The Book

Aviation Safety

The past 10 years have been the best in the country's aviation history with 153 fatalities. That's two deaths for every 100 million passengers on commercial flights, according to an Associated Press analysis of government accident data.

The improvement is remarkable. Just a decade earlier, at the time the safest, passengers were 10 times as likely to die when flying on an American plane. The risk of death was even greater during the start of the jet age, with 1,696 people dying — 133 out of every 100 million passengers — from 1962 to 1971. The figures exclude acts of terrorism.

...

There are a number of reasons for the improvements.

- *The industry has learned from the past. New planes and engines are designed with prior mistakes in mind. Investigations of accidents have led to changes in procedures to ensure the same missteps don't occur again.*
- *Better sharing of information. New databases allow pilots, airlines, plane manufactures and regulators to track incidents and near misses. Computers pick up subtle trends. For instance, a particular runway might have a higher rate of aborted landings when there is fog. Regulators noticing this could improve lighting and add more time between landings.*

("It's never been safer to fly; deaths at record low", AP, link to Seattle PI version.)

Well, it seems there's nothing for information security to learn here. Move along.

Filed under: Doing it Differently, measurement, Science of Risk Management by adam on Wednesday, January 25, 2012

Journey: Systems Safety

"How Complex Systems Fail,"
Richard I Cook, MD
http://web.mit.edu/2.75/resources/random/How%20Complex%20Systems%20Fail.pdf

Engineering a Safer World,
Nancy G Leveson
https://mitpress.mit.edu/sites/default/files/titles/free_download/9780262016629_Engineering_a_Safer_World.pdf

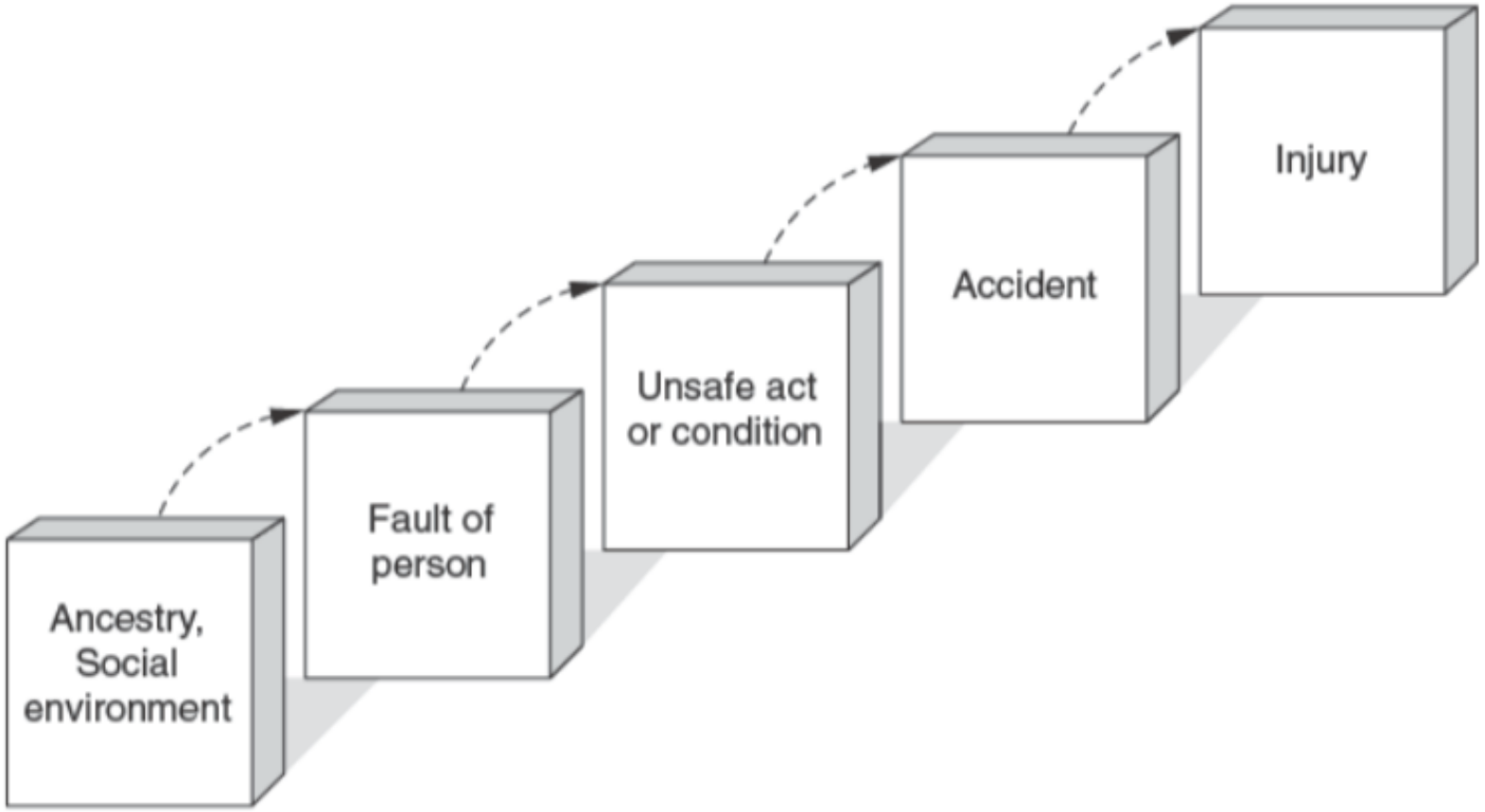
2016 STAMP Workshop
http://psas.scripts.mit.edu/home/2016-stamp-workshop/

Why STAMP?

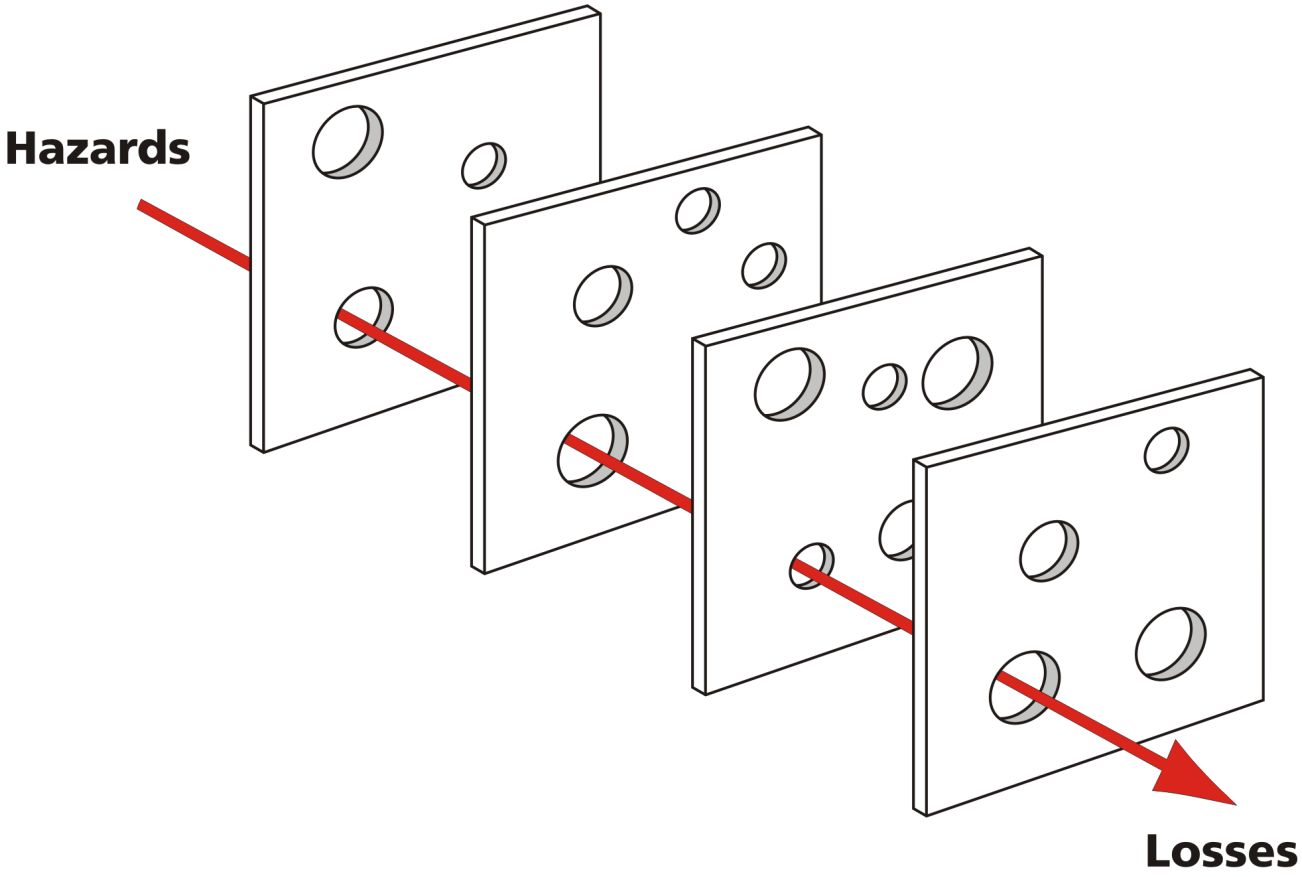
STAMP Tools

- STAMP: Causality Model
- CAST: Accident Analysis
- STPA: Hazard Analysis
- STECA: Early Concept Analysis
- STPA-Sec: Security Analysis
- Leading Indicators

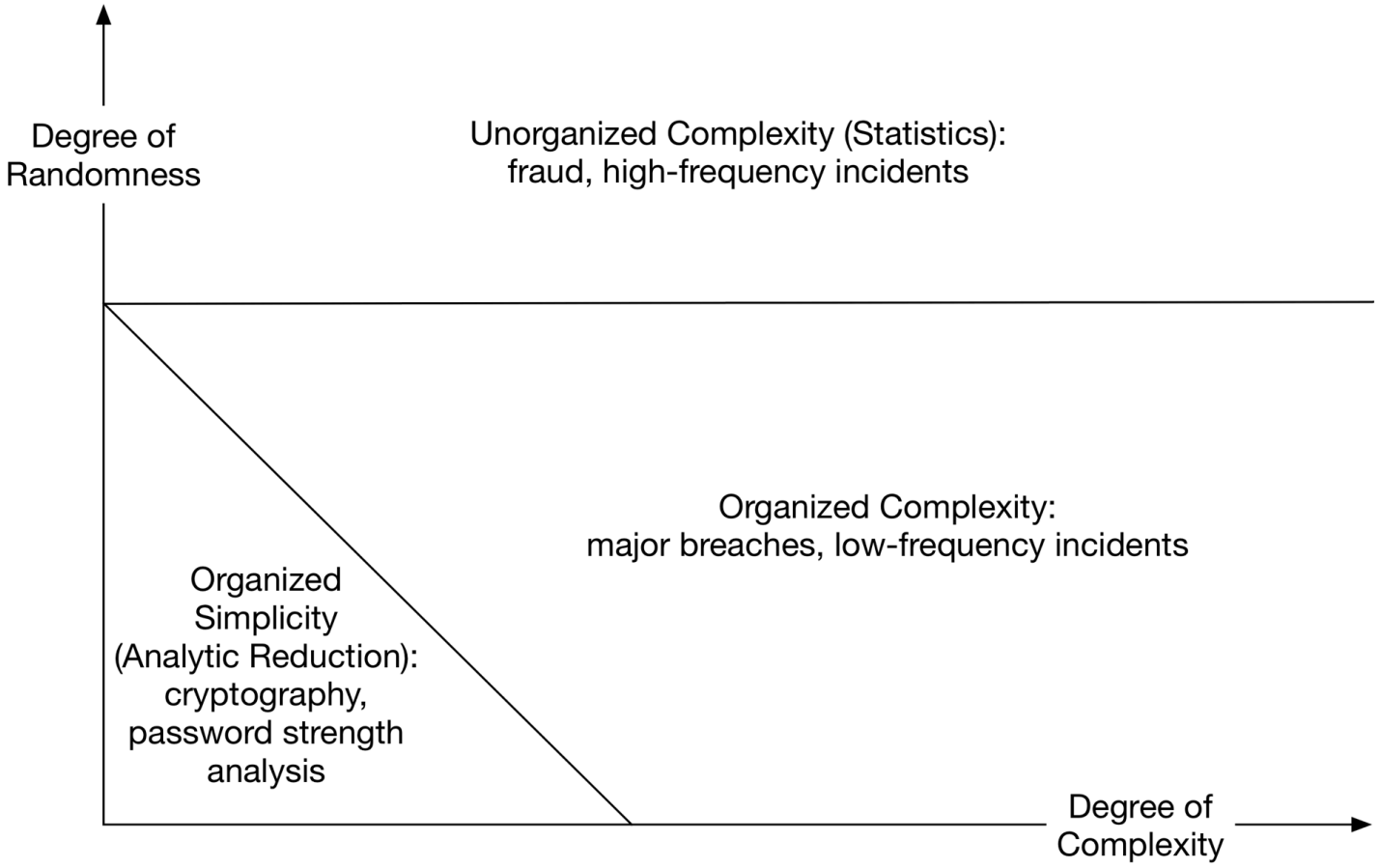
Heinrich's Domino model



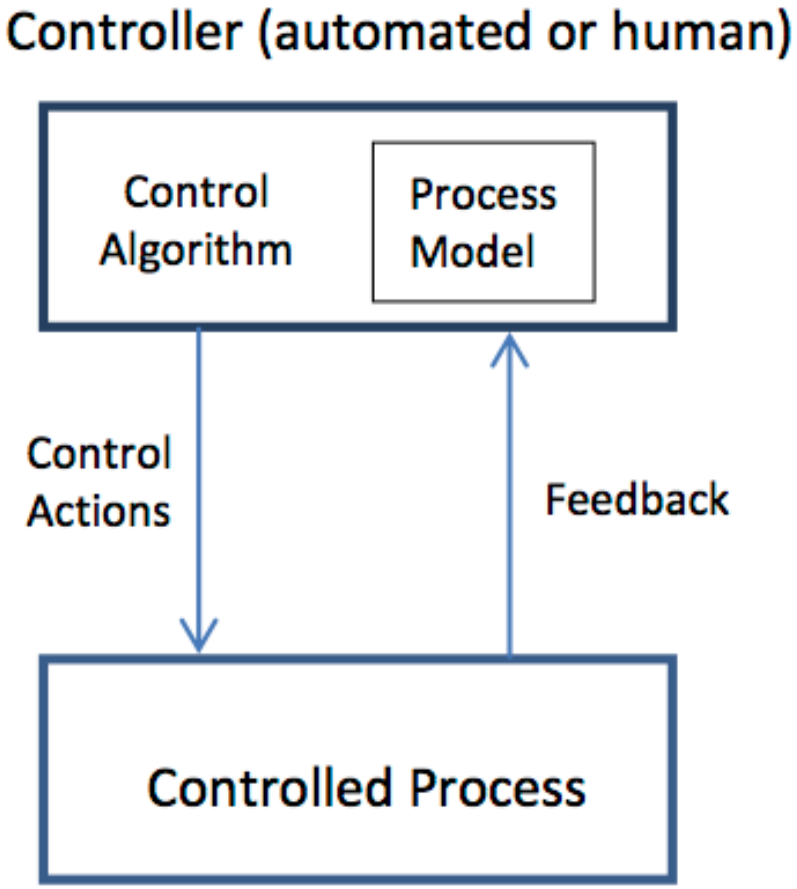
Swiss Cheese model

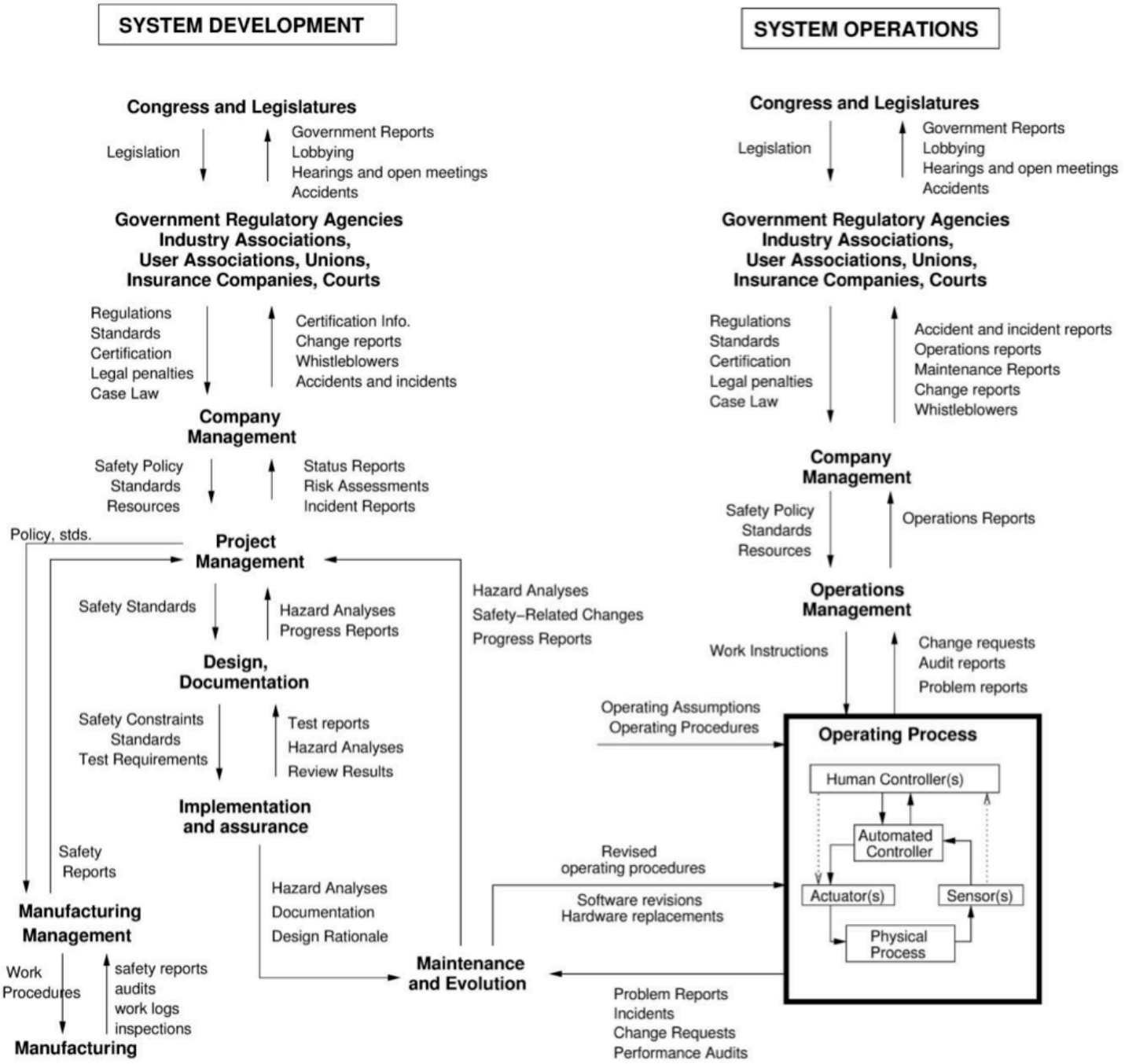


Organized Complexity



STAMP model


Controller (automated or human)
Control Algorithm
Process Model
Control Actions
Feedback
Controlled Process


SYSTEM DEVELOPMENT
SYSTEM OPERATIONS
Congress and Legislatures
Legislation
Government Reports
Lobbying
Hearings and open meetings
Accidents
Government Regulatory Agencies
Industry Associations,
User Associations, Unions,
Insurance Companies, Courts
Regulations
Standards
Certification
Legal penalties
Case Law
Certification Info.
Change reports
Whistleblowers
Accidents and incidents
Company
Management
Safety Policy
Standards
Resources
Status Reports
Risk Assessments
Incident Reports
Policy, stds.
Project
Management
Safety Standards
Hazard Analyses
Progress Reports
Design,
Documentation
Safety Constraints
Standards
Test Requirements
Test reports
Hazard Analyses
Review Results
Implementation
and assurance
Safety
Reports
Manufacturing
Management
Work
Procedures
safety reports
audits
work logs
inspections
Manufacturing
Hazard Analyses
Documentation
Design Rationale
Maintenance
and Evolution
Hazard Analyses
Safety-Related Changes
Progress Reports
Revised
operating procedures
Software revisions
Hardware replacements
Problem Reports
Incidents
Change Requests
Performance Audits
Congress and Legislatures
Legislation
Government Reports
Lobbying
Hearings and open meetings
Accidents
Government Regulatory Agencies
Industry Associations,
User Associations, Unions,
Insurance Companies, Courts
Regulations
Standards
Certification
Legal penalties
Case Law
Accident and incident reports
Operations reports
Maintenance Reports
Change reports
Whistleblowers
Company
Management
Safety Policy
Standards
Resources
Operations Reports
Operations
Management
Work Instructions
Change requests
Audit reports
Problem reports
Operating Assumptions
Operating Procedures
Operating Process
Human Controller(s)
Automated
Controller
Actuator(s)
Sensor(s)
Physical
Process

STAMP Assumptions

Old Assumption	New Assumption
Safety is increased by increasing system or component reliability; if components do not fail, then accidents will not occur.	High reliability is neither necessary nor sufficient for safety.
Accidents are caused by chains of directly related events. We can understand accidents and assess risk by looking at the chains of events leading to the loss.	**Accidents are complex processes involving the entire sociotechnical system. Traditional event-chain models cannot describe this process adequately.**
Probabilistic risk analysis based on event chains is the best way to assess and communicate safety and risk information.	**Risk and safety may be best understood and communicated in ways other than probabilistic risk analysis.**

STAMP Assumptions

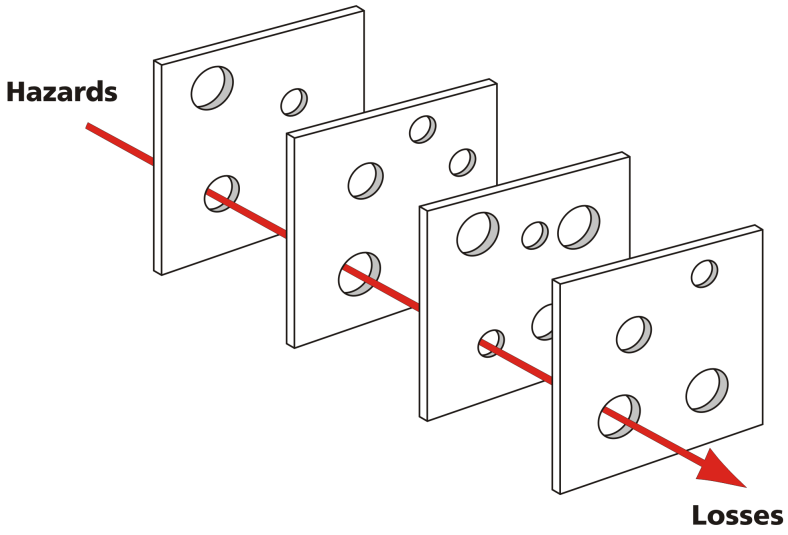
Old Assumption	New Assumption
Most accidents are caused by operator error. Rewarding safe behavior and punishing unsafe behavior will eliminate or reduce accidents significantly.	Operator error is a product of the environment in which it occurs. To reduce operator “error” we must change the environment in which the operator works.
Highly reliable software is safe.	Highly reliable software is not necessarily safe. Increasing software reliability will have only minimal impact on safety.

STAMP Assumptions

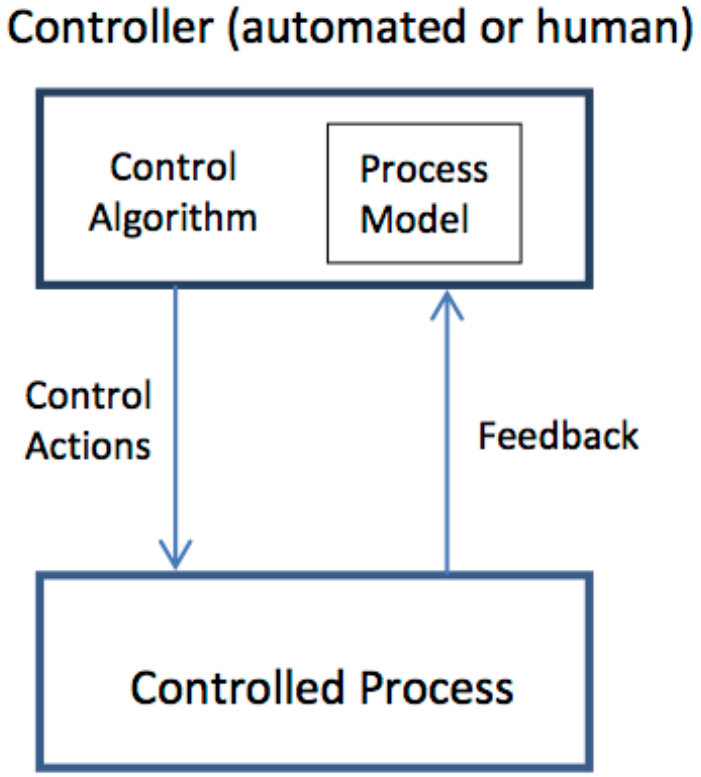
Old Assumption	New Assumption
Assigning blame is necessary to learn from and prevent accidents or incidents.	Blame is the enemy of safety. Focus should be on understanding how the system behavior as a whole contributed to the loss and not on who or what to blame for it.
Major accidents occur from the chance simultaneous occurrence of random events.	**Systems will tend to migrate toward states of higher risk. Such migration is predictable and can be prevented by appropriate system design or detected during operations using leading indicators of increasing risk.**

Security vs Safety

Security: Layered defenses against possible attacks

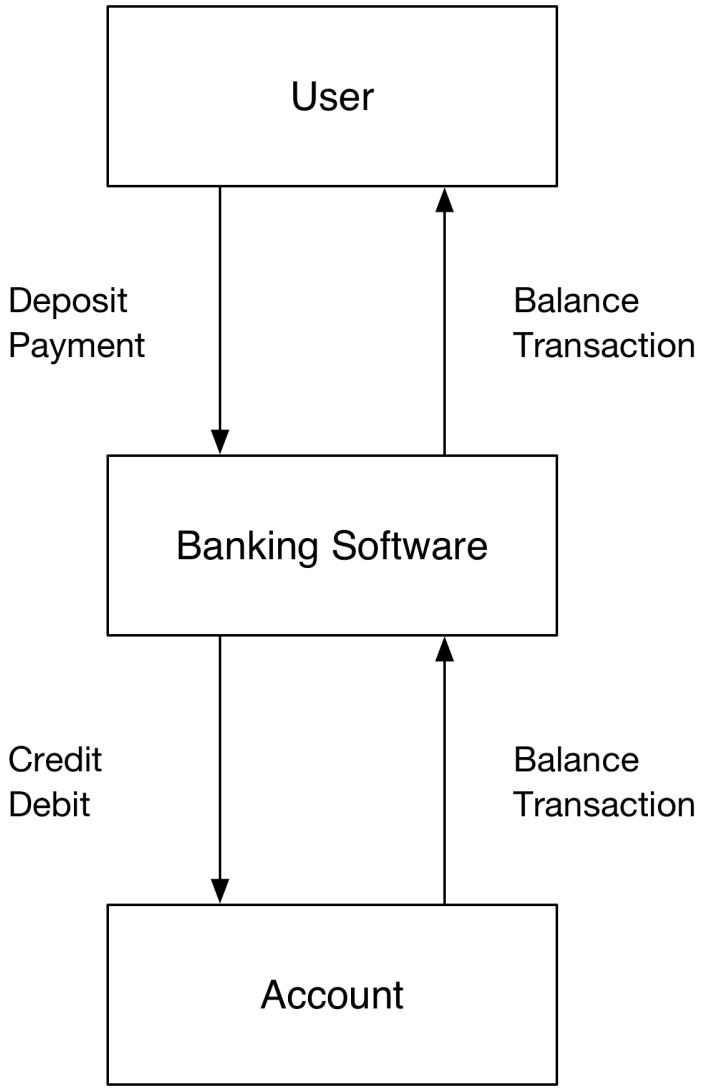


Safety: Keep system out of hazardous state



A walkthrough of STPA-Sec using a simple banking application

STPA-SEC EXAMPLE


User
Deposit
Payment
Balance
Transaction
Banking Software
Credit
Debit
Balance
Transaction
Account

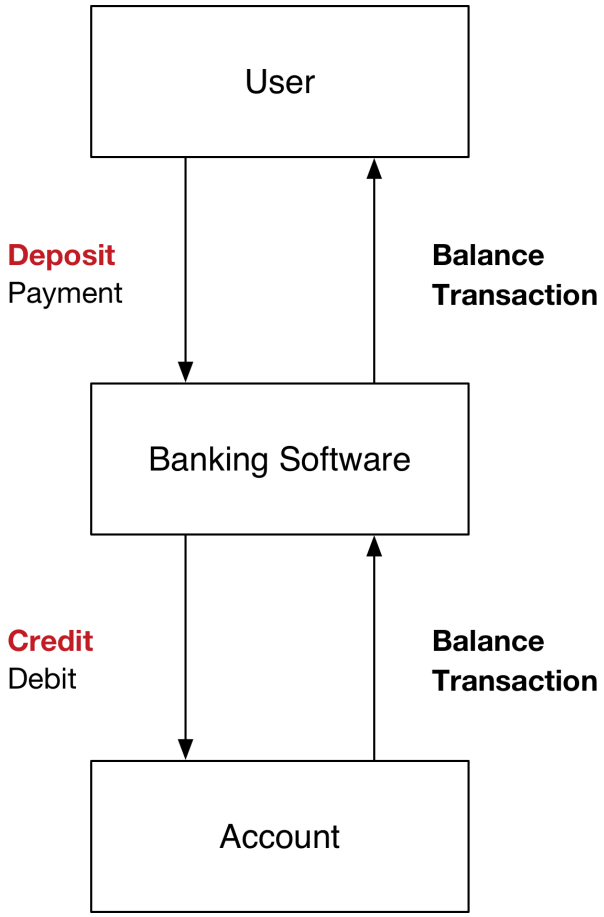
Accidents, Hazards, Constraints

- A1: Loss of money from bank account
- A2: Loss of privacy, banking data exposed

- H1: Unintended payment of funds
- H2: Failure to receive deposits
- H3: Data exposed to unauthorized party

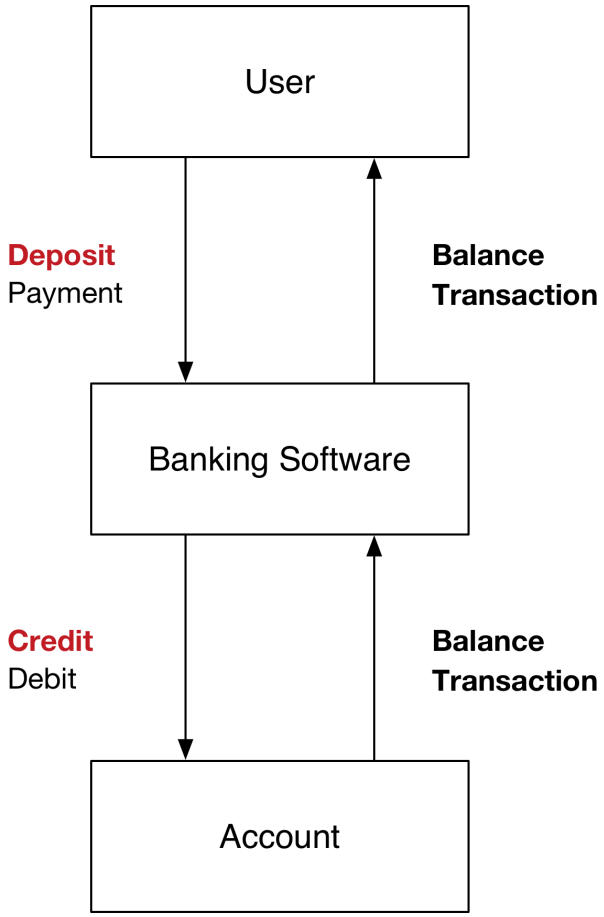
- C1: System must prevent unintended debits
- C2: System must fully credit accounts
- C2: System must not expose transaction details

Deposit – Unsafe Control Actions



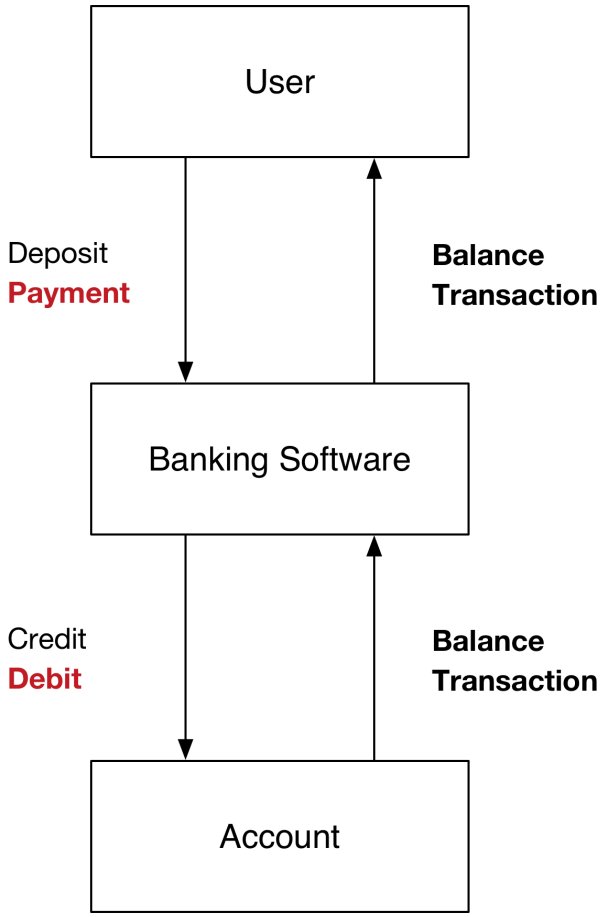
Control Action	CREDIT
Not Provided	
Provided	
Timing	
Duration	

Deposit – Unsafe Control Actions



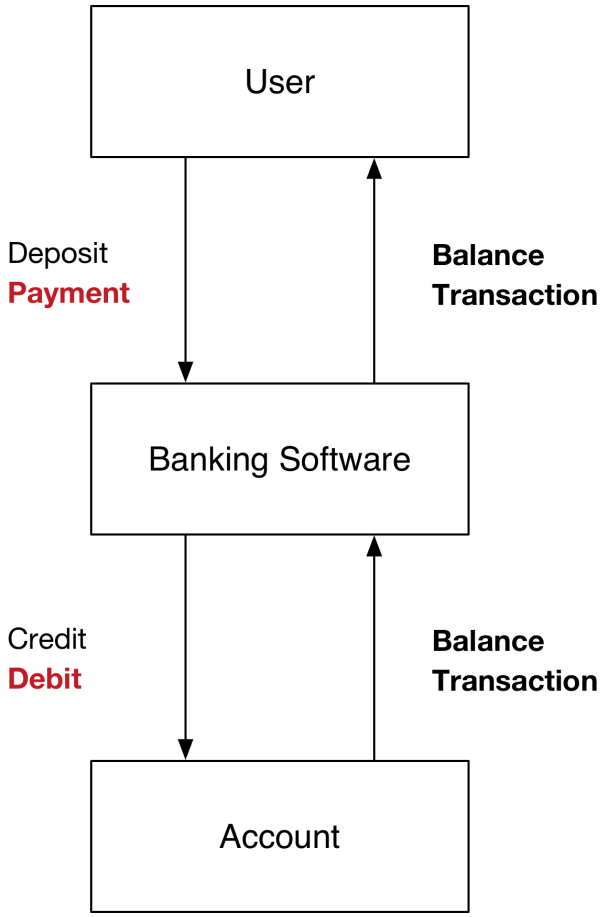
Control Action	CREDIT
Not Provided	UCA1: Software does not credit account when user provides valid deposit (H2)
Provided	No Hazard
Timing	No Hazard
Duration	UCA2: Software does not credit full amount when user provides valid deposit (H2)

Payment – Unsafe Control Actions



Control Action	DEBIT
Not Provided	
Provided	
Timing	
Duration	

Payment – Unsafe Control Actions



Control Action	DEBIT
Not Provided	No Hazard
Provided	UCA3: Software debits account when user has not requested a payment (H1)
Timing	No Hazard
Duration	UCA4: Software debits more than full amount when user has requested a payment (H1)

Unsafe Control Actions

Control Action	Not Provided	Provided	Timing	Duration
Credit (Deposit)	UCA1: Software does not credit account when user provides valid deposit (H2)	No Hazard	No Hazard	UCA2: Software does not credit full amount when user provides valid deposit (H2)
Debit (Payment)	No Hazard	UCA3: Software debits account when user has not requested a payment (H1)	No Hazard	UCA4: Software debits more than full amount when user has requested a payment (H1)

Generating Scenarios with STRIDE

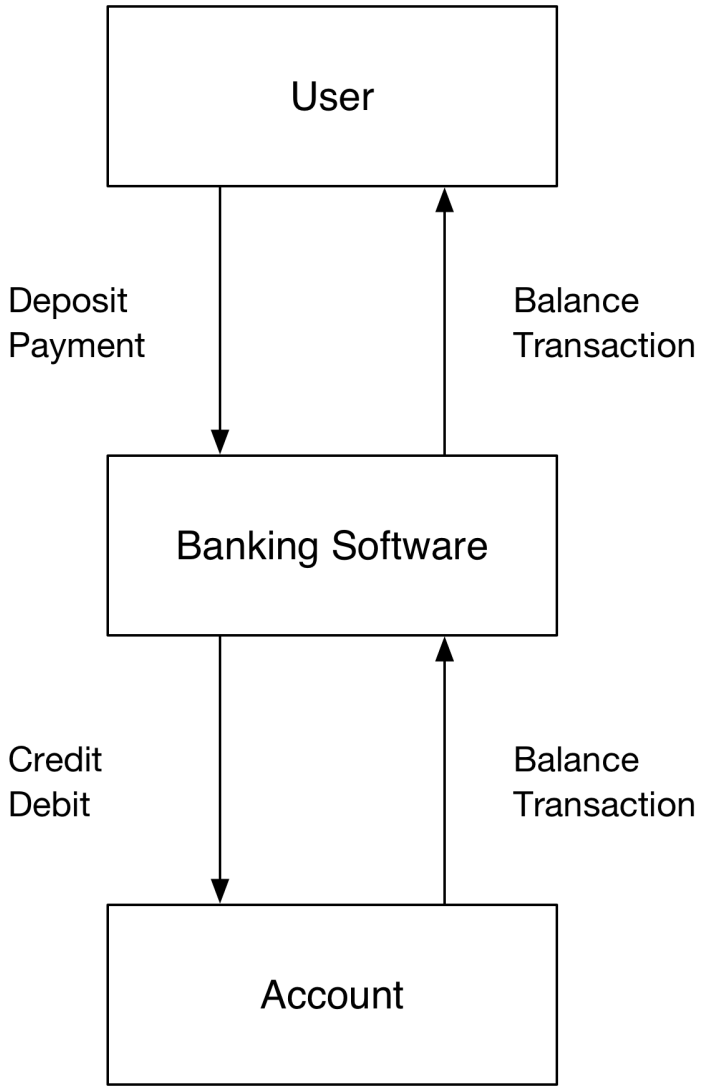
	Spoofing	Tampering	Denial of Service
UCA1	S1: Attacker spoofs bank software and user (MITM) and changes deposit/payment	S2: Attacker alters deposit/payment command	S3: Attacker blocks deposit command
UCA2	S1: Attacker spoofs bank software and user (MITM) and changes deposit/payment	S2: Attacker alters deposit/payment command	N/A
UCA3	S4: Attacker spoofs user and provides payment command	N/A	N/A
UCA4	S1: Attacker spoofs bank software and user (MITM) and changes deposit/payment	S2: Attacker alters deposit/payment command	N/A

Scenarios

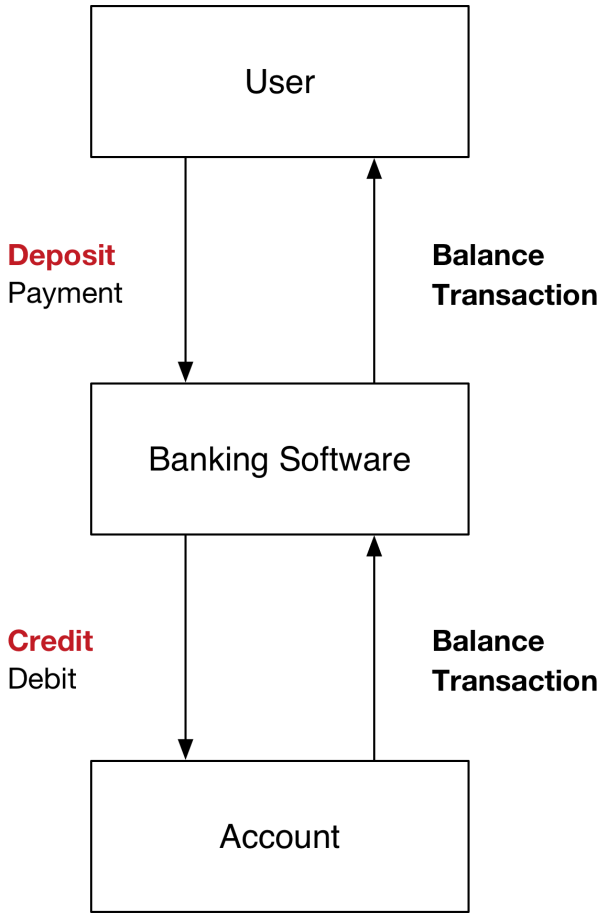
- S1: Attacker spoofs bank software and user (MITM) and changes deposit/payment
- S2: Attacker alters deposit/payment command
- S3: Attacker blocks deposit command
- S4: Attacker spoofs user and provides payment command

Wait!

- Repudiation: skipped, doesn't impact hazards we identified
- Elevation of Privilege: skipped, doesn't make sense in the context of the system
- Information Disclosure: hmm…

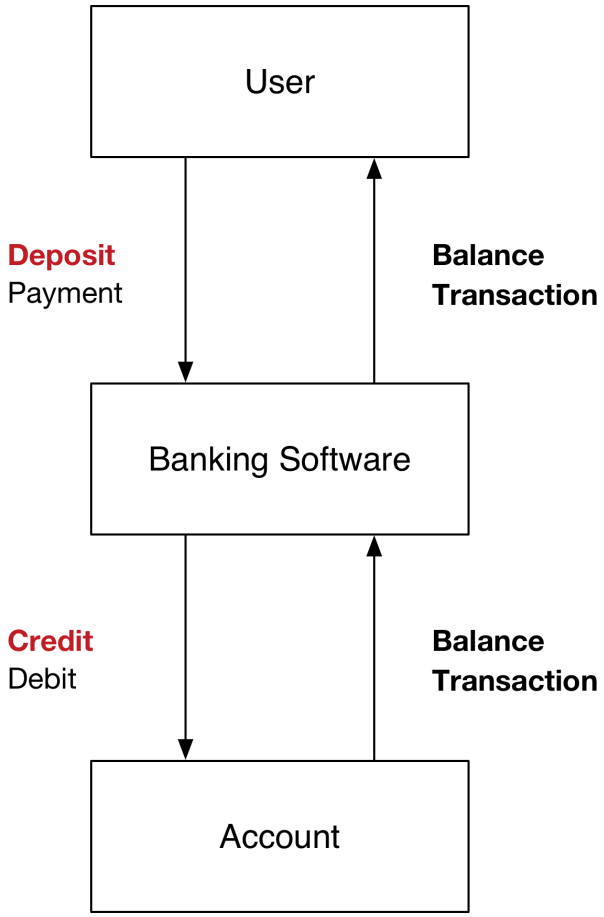

User
Deposit
Payment
Balance
Transaction
Banking Software
Credit
Debit
Balance
Transaction
Account

Deposit – UCA v2



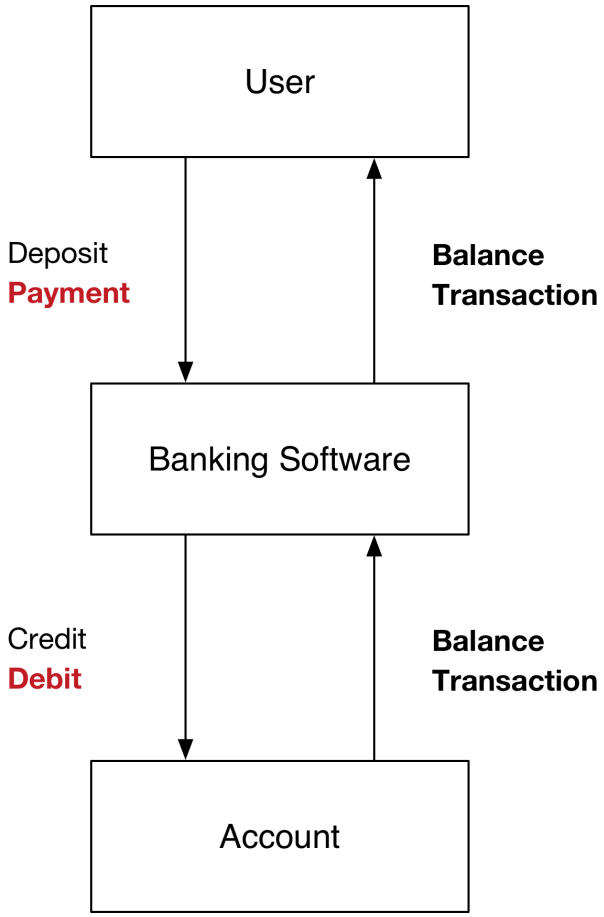
Control Action	CREDIT
Not Provided	UCA1: Software does not credit account when user provides valid deposit (H2)
Provided	No Hazard
Timing	No Hazard
Duration	UCA2: Software does not credit full amount when user provides valid deposit (H2)
Intercepted	

Deposit – UCA v2



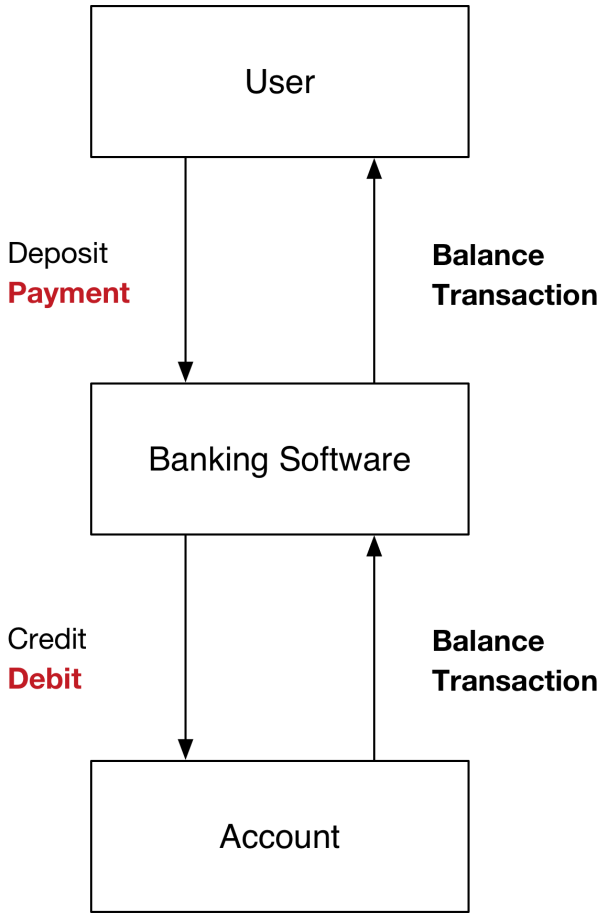
Control Action	CREDIT
Not Provided	UCA1: Software does not credit account when user provides valid deposit (H2)
Provided	No Hazard
Timing	No Hazard
Duration	UCA2: Software does not credit full amount when user provides valid deposit (H2)
Intercepted	UCA5: Software discloses account balance and/or transaction details (H3)

Payment – UCA v2



Control Action	DEBIT
Not Provided	No Hazard
Provided	UCA3: Software debits account when user has not requested a payment (H1)
Timing	No Hazard
Duration	UCA4: Software debits more than full amount when user has requested a payment (H1)
Intercepted	

Payment – UCA v2



Control Action	DEBIT
Not Provided	No Hazard
Provided	UCA3: Software debits account when user has not requested a payment (H1)
Timing	No Hazard
Duration	UCA4: Software debits more than full amount when user has requested a payment (H1)
Intercepted	UCA5: Software discloses account balance and/or transaction details (H3)

Generating Scenarios v2

	Spoofing	Tampering	Information Disclosure	Denial of Service
UCA1	S1: Attacker spoofs bank software and user (MITM) and changes deposit/ payment	S2: Attacker alters deposit/payment command	N/A	S3: Attacker blocks deposit command
UCA2	S1: Attacker spoofs bank software and user (MITM) and changes deposit/ payment	S2: Attacker alters deposit/payment command	N/A	N/A
UCA3	S4: Attacker spoofs user and provides payment command	N/A	N/A	N/A

Generating Scenarios v2

	Spoofing	Tampering	Information Disclosure	Denial of Service
UCA4	S1: Attacker spoofs bank software and user (MITM) and changes deposit/ payment	S2: Attacker alters deposit/payment command	N/A	N/A
UCA5	S5: Attacker spoofs user and reads balance/ transaction	N/A	S6: Attacker reads deposit/ payment command **S7: Attacker reads balance/ transaction reply**	N/A

Scenarios

- S1: Attacker spoofs bank software and user (MITM) and changes deposit/payment
- S2: Attacker alters deposit/payment command
- S3: Attacker blocks deposit command
- S4: Attacker spoofs user and provides payment command
- S5: Attacker spoofs user and reads balance/ transaction
- S6: Attacker reads deposit/payment command
- S7: Attacker reads balance/transaction reply

Observations

STPA-Sec

Pro	Con
More efficient, effective at systematically modeling system and identifying unsafe system states	Does not account for information disclosure (generally not a safety concern) in current model
Useful for both engineering and failure analysis	Does not provide methods for prioritization
Accounts for human behavior – the sociotechnical system	Not widely adopted; primarily an academic model today
Avoids blame	

Thank You!

Contact Information:

John Benninghoff
john@transvasive.com
http://transvasive.com
https://information-safety.org
Twitter: @transvasive