Systems thinking for safety and security

Systems Thinking for Safety and Security[1]

William Young
Engineering Systems Division
MIT
Cambridge, MA 02139
781-981-7352
wyoung@mit.edu

Nancy Leveson
Aeronautics and Astronautics
MIT
Cambridge, MA 02139
617-258-0505
leveson@mit.edu

ABSTRACT

The fundamental challenge facing security professionals is preventing losses, be they operational, financial or mission losses. As a result, one could argue that security professionals share this challenge with safety professionals. Despite their shared challenge, there is little evidence that recent advances that enable one community to better prevent losses have been shared with the other for possible implementation. Limitations in current safety approaches have led researchers and practitioners to develop new models and techniques. These techniques could potentially benefit the field of security. This paper describes a new systems thinking approach to safety that may be suitable for meeting the challenge of securing complex systems against cyber disruptions. Systems-Theoretic Process Analysis for Security (STPA-Sec) augments traditional security approaches by introducing a top-down analysis process designed to help a multidisciplinary team consisting of security, operations, and domain experts identify and constrain the system from entering vulnerable states that lead to losses. This new framework shifts the focus of the security analysis away from threats as the proximate cause of losses and focuses instead on the broader system structure that allowed the system to enter a vulnerable system state that the threat exploits to produce the disruption leading to the loss.

Categories and Subject Descriptors

K.6.5 Security and Protection

General Terms

Security

Keywords

STAMP, STPA, STPA-SEC, Critical Infrastructure, Systems Thinking.

1. INTRODUCTION

Rapid developments in software and the rise of software intensive systems have produced significant benefits to the global economy and society as a whole. These benefits have given rise to a growing dependence on the services provided by these systems and their corresponding physical and logical infrastructures. Disrupting or otherwise exploiting these infrastructures has become the goal of a wide range of potential adversaries. The problem is further aggravated by the fact that disruptions may also result from unintentional actions taken by well-intentioned operators within the systems themselves.

Despite increased funding and resources, we do not appear to be making satisfactory progress in our ability to secure the complex systems that we are increasingly able to create. Arguably, new approaches are needed. This paper presents one such approach. Applying lessons learned from nearly three decades of research in safety engineering for complex systems, this paper presents a modified version of a new, more powerful hazard analysis technique, called System-Theoretic Process Analysis (STPA), developed by Leveson. The extension of STPA, called STPA for Security (STPA-Sec), addresses the growing problem of securing software intensive systems against intentional disruptions.

Cyber security has largely been framed as a tactics problem, focusing on how best to defend networks and other information assets against threats. While necessary, we believe this misses the greater objective of securing the systems' ability to produce the services and functions society depends on. Defending networks is not an end in itself; rather it is a means to protecting these higher-level services and missions against disruptions. Reframing the problem into one of strategy may ultimately produce better outcomes. In practice, this reframing involves shifting the majority of security analysis away from guarding against *attacks* (tactics) and more towards the broader socio-technical *vulnerabilities* that allow disruptions to propagate throughout the system (strategy). Put another way, rather than focusing the majority of the security efforts on threats from adversary action, which are beyond the control of the security specialist, security efforts should be focused on the larger, more inclusive goal of controlling system vulnerabilities. To accomplish this goal, STPA-Sec identifies and enforces required constraints on unsecure control actions that place the system in vulnerable states when subjected to disturbances (whether intentional or unintentional).

[1] This paper is in the *Proceedings of the 2013 Annual Computer Security Applications Conference (ACSAC 2013)*, New Orleans, December 2013.

This paper is organized into three parts. The first section discusses the limitations associated with treating cyber security solely as a tactics problem. The second section introduces systems thinking as a means to reframe cyber security as a strategy problem and presents a systems approach used successfully to improve safety in complex systems. The third section of the paper discusses STPA-Sec and presents a simple example.

2. LIMITATIONS IN TREATING CYBER SECURITY AS A TACTICS PROBLEM

The cyber security field tends to draw heavily on language, metaphors, and models from military operations. There is an important distinction in military doctrine between tactics and strategy. *Strategy* can be considered as the art of gaining and maintaining continuing advantage. In contrast, *tactics* are prudent means to accomplish a specific action. Tactics are focused on threats, while strategy models are focused on outcomes.

Most current cyber security assessments have knowingly or unknowingly adopted tactics models. Tactics models emphasize how best to defeat a given threat. For example, a pilot has specific tactics that should be employed to defeat an adversary aircraft in combat. The threat dictates the tactics that will most likely lead to success, so properly identifying the threat is the first step in solving the tactical problem. Likewise, analyzing the threat is the first step in the National Institute of Standards and Technology (NIST) security standards [1].

In tactics models, losses are conceptualized as specific events *caused* by threats. For example, a security incident consisting of a data breach with an accompanying loss of customer Personally Identifiable Information (PII) is viewed as a single occurrence where an adversary successfully precipitates a chain of events leading to a loss. In almost all such cases, security analysts will identify some proximate cause that should have served as the last barrier or line of defense. According to this model, if only the barrier would have been in place, then the attack would have failed.

This type of approach is often described as “breaking the chain” and is a commonly used in security literature as a framework for conceptualizing the goal of successful security practices. In the case of the TJMAXX data loss, for example, the proximate cause of the data loss was attributed to the failure of the store to use the proper wireless encryption on their networks [2]. Although threats exploiting vulnerabilities produce the loss event, tactics models treat the threat as the *cause* of the loss. According to this thinking, the loss is attributed to a threat successfully circumventing several barriers to reach its goal. Preventing losses, then, is heavily dependent on the degree to which security analysts can correctly identify potential attackers, their motives, capabilities, and targeting. Once equipped with this knowledge, security analysts can analyze their systems to determine the most likely route (or causal chain) attackers may take to reach their goal. Resources can then be allocated to place barriers along the chain and prevent losses.

This chain-of-events model of causality is the same one used in safety engineering, where the attempt to avoid accidents is focused on breaking the chain by either preventing the individual failure events or erecting barriers between them to prevent their propagation.

The current threat-based security approach succeeds best under the same circumstances that allow tactical success on the battlefield: good intelligence and a context where cause and effect are closely linked temporally and spatially. Good intelligence reduces uncertainty. When the means, motives and capabilities of potential attackers are so well understood that their preferred “route” to their goal can be predicted, then security barriers can be erected to break the chain. In these cases, losses are prevented when defenders skillfully execute the well-established practices and procedures the situation demands. An example is network administrators disabling unused ports or updating the latest malware signatures.

A threat-based approach is useful for identifying and countering security threats against a single, well-defined and well-understood system asset or component. In these cases, a threat actor’s potential actions might be evaluated through stochastic models to yield a most likely course of action to attack the asset. Once this adversary course of action is identified, the security analyst can provide advice to senior leaders on how best to allocate limited resources to thwart the attack and break the chain. In other words, the high level of threat understanding enables security analysts to predict not only where an adversary will attack, but also the logical and physical infrastructure that is most important to defend in order to thwart the attack.

Unfortunately, this approach suffers significant limitations when applied to securing diverse, interconnected infrastructure supporting large-scale, complex organizational activities against little understood and rapidly evolving adversaries. The current security model doesn't accommodate the properties of software intensive systems, nor can the loss mechanism be accurately reflected in a linear causality model. Losses occur as the result of complex interactions between the various socio-technical components in modern organizations and businesses. The loss is an emergent system outcome, not one found in the failure of individual components.

The rest of this paper describes an alternative *strategy* model for cyber security.

3. A NEW APPROACH BASED ON SYSTEMS THINKING

Conceiving of causality as a chain of directly related events is at least 200 years old. Traditional safety engineering techniques, such as fault tree analysis, based on this model were developed over 50 years ago, before computers were used to create the highly-interactive, tightly coupled, software intensive systems common today.

The limitations of traditional engineering methods and the need to field increasingly complex systems during and immediately following World War II led to the development of modern systems theory in the 1940s and 1950s [3]. Systems theory provides the philosophical and intellectual foundation for systems engineering and also for a new, more powerful model of accident causality developed by Leveson called STAMP (System-Theoretic Accident Model and Processes) [4].

STAMP extends traditional causality models from a focus on component failures to defining losses as resulting from interactions among humans, physical system components, and the environment. Losses result when safety constraints on system component behavior and interactions are violated. Thus the focus shifts from "preventing failures" to "enforcing safety constraints on system behavior."

In systems theory, the system is conceived as a hierarchical structure, where each level enforces constraints on the behavior of components at the next lower level. These constraints control emergent system behavior, such as safety and security. Control loops operate between each level of this hierarchical control structure. Figure 1 shows the general form of such control loops.

Every controller contains a model of the process it is controlling. This model is used to determine what control actions are necessary. Many accidents related to software or human operators are not the result of software or human "failure" (whatever that might mean) but stem from inconsistencies between the controller's model of the controlled process and the actual process state. For example, friendly fire accidents are usually the result of mistaking a friendly aircraft for an enemy. Unsafe control actions can result from providing a control action that leads to a hazard, not providing a control action that is needed to prevent a hazard, providing a control action too early or too late, or continuing a control action too long or stopping it too soon.

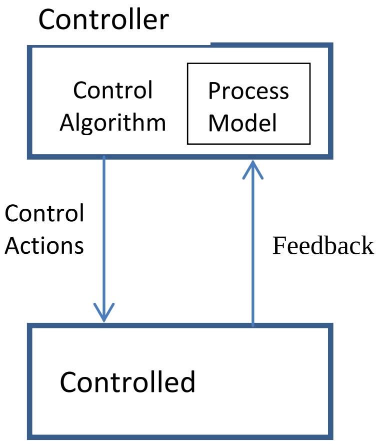


Figure 1. Basic Control Loop

STPA (System-Theoretic Process Analysis) is a new hazard analysis method based on STAMP. It is being used successfully in almost every industry and even non-engineering applications such as food safety and financial systems safety. We believe it also has potential for application to security. The rest of this paper shows how STPA might be extended into a new cyber security analysis technique called STAMP-Sec.

4. APPLYING STPA TO SECURITY

In the broad sense, security can be considered as protecting a system against *intentional* disruptions. Adversary activity is a common source of these disruptions, but it is not the only source. Trusted insiders can also take action to disrupt the operations of systems. Safety can be considered as protecting that same system against *unintentional* disruptions. Hazards lead to safety incidents in the same way that vulnerabilities lead to security incidents. We believe that the key question facing today's security analysts is how to control vulnerabilities, not how to avoid threats.

The example provided here, a nuclear reactor system, represents a type of critical infrastructure that needs to be protected against cyber attack. Physical plants represent high payoff targets for any number of potential adversaries and clearly must be defended. In the following example, a real nuclear power plant design was used but the details had to be changed for obvious reasons. The full analysis (for safety) can be found in Thomas [5].

The analysis was done on a fully digital Pressurized Water Reactor (PWR). Computers direct all control systems including those protecting the nuclear reactor (called the "safety system" in nuclear engineering). The plant produces electricity by using heat from the reactor to generate steam that powers a turbine. The turbine produces electricity that is transferred into the power grid for consumer and commercial users.

The example STPA-Sec analysis focuses on the operation of the Main Steam Isolation Valve (MSIV) located on the steam line from the steam generator. The MSIV is open during normal operations to enable system cooling. The MSIV can be closed to isolate the steam generator from the rest of the system in case of a problem with the steam containment system, such as a leak or break. However, closing the MSIV also prevents the secondary system from providing adequate cooling to the primary system. Lack of adequate cooling can lead to equipment damage or even a plant meltdown. Therefore, it is critical to plant operations that the MSIV be open or closed as dictated by the situation. Failure to do so can have disastrous consequences. Note that several real world cyber security incidents have occurred over the malfunctioning of valves [6].

STPA-Sec shares the same four basic process steps with its safety counterpart, STPA, although the results and detailed procedures may be different. The first step is establishing the systems engineering foundation for the security analysis. Then the control actions that threaten system security are identified. These control actions are used to create security requirements and constraints. The fourth and final step is to identify causal scenarios that can give rise to violations of the security constraints.

Step 1: Establishing the Systems Engineering Foundation

Because the current security approach is largely threat based, security specialists may be tempted to conduct their assessments in isolation. This approach is logical from a tactical security perspective, but likely misses the larger systems perspective. Threats exercise physical or logical infrastructure vulnerabilities to disrupt or otherwise hinder system function. In turn, the adverse impacts on system function prevent the targeted organization from delivering the services that represent its *raison de entre*. Starting with physical threats represents a bottom-up tactics approach in contrast with a system engineering top-down strategy.

STPA-Sec reverses the tactics-based bottom-up approach by starting at the highest level of the system. The critical first step is identifying the set of losses that are considered unacceptable. These losses likely extend beyond the physical and logical entities into the higher level services provided by these entities. Rather than beginning with tactics questions of *how* best to guard the network against threats, STPA-Sec's systems thinking approach begins with strategy questions of *what* essential services and functions must be secured against disruptions or *what* represents an unacceptable loss. This step requires clearly identifying the "what(s)" and then using that information to reason more thoroughly about the "how(s)" that can lead to the undesirable outcomes. The analysis moves from general to specific, from abstract to concrete.

Two distinctions of this approach are immediately clear. The first distinction is that security experts are unlikely to be capable of answering the "what" questions isolated from organizational leaders and operations personnel. Security involves tradeoffs and the allocation of scarce resources. Although security concerns and insights should inform these decisions, the ultimate responsibility for making them rests with the senior leaders charged with ensuring that the organization provides its essential business or functional services. Although security can advise, it will be the senior leaders that decide.

During the security analysis, it is possible or even likely that potential conflicts may arise between priorities. For example, there is a constant tension between the need to secure and the need to share access to information resources. A bottom-up approach might identify the security challenge associated with granting expanded access to information systems, however, the approach lacks the larger context to provide insight into the corresponding necessity to share in order to accomplish key organizational outcomes. If a decision is made with regard to one of the priorities without consideration for the other, a problem is likely to arise. This problem may not be visible to the security team, but will be visible to the operations team that requires the access in order to perform the higher-level system functions.

A prudent way to properly address the potential conflict is through a top-down process such as STPA-Sec. Such an approach provides the necessary context for decision makers to evaluate the higher-level needs rather than focusing on tactical level details. Certainly, the tactical details are important. However, over emphasis and premature emphasis on the details of task execution absent the larger context of the systemic purpose the tasks support can lead to substituting tactics for strategy.

The second distinction is that STPA-Sec begins with organizational purpose and goals, not physical or logical assets. Successful security assessments require a careful establishment of priorities. By establishing the priorities at the start of the assessment as opposed to the end, the priorities form a framework to both focus and guide the security assessment. This evaluation can only be properly made with the benefit of perspective into the larger, overall system function.

One of the most important aspects of the environment is adversary activity. Certainly adversary action is a critical consideration in addressing security and preventing intentional losses. Yet, focusing on adversaries or threats too early in the process and absent the benefit of context, limits the overall strategic-level utility of the security assessment. Put another way, the goal of security is not to guard the physical network and prevent intrusions. The goal is to ensure that the critical functions and ultimately the services the network and systems provide are maintained in the face of disruptions.

Adversary action is only one such disruption (albeit an important one). One potential benefit of applying STPA-Sec to security would be to expand the focus of security efforts more toward those things that are actually within the control of the organization's leaders, rather than simply expecting cyber security experts to defend from a position of disadvantage. The disadvantage occurs because security analysts and defenders are forced to react to threats and other environmental disruptions, rather than proactively shaping the situation by identifying and controlling system vulnerabilities.

This shift also represents a more judicious use of resources. Multiple threats and disruptions can exploit a given system vulnerability. Even under current tactics-based models, a threat must ultimately exploit a vulnerability to produce the system loss. Rather than trying to initially identify all of the threats and then move up to the vulnerabilities they might exploit to produce the loss, a more reasonable approach might be to start with addressing system vulnerabilities which are likely far fewer than threats and, if controlled, can prevent losses to numerous threats and disruptions.

Additionally, controlling vulnerabilities allows security analysts to prevent not only the disruptions from known threats, but also disruptions introduced by unknown threats. In other words, the source of the disruption does not matter. What matters is identifying and controlling the vulnerability. This limits the intelligence burden required to perform the initial system security analysis. STPA-Sec eventually addresses threats, but does so much later in the analysis process after generating a deeper systemic understanding of the context under which the threats may operate and the disruptions that actually lead to critical loss events.

In the nuclear power plant example, Table 1 shows high-level vulnerabilities and their relation to four identified loss events. The four loss events are:

L1: Human Serious Injury or Loss of Life

L2: Environmental Contamination

L3: Significant Equipment Damage

L4: Loss of Power Production to the Grid

Table 1. Vulnerabilities and Related Loss Events

Vulnerability	Related Loss Event
V-1: Release of radioactive materials	L1, L2
V-2: Reactor temperature too high	L1, L2, L3, L4
V-3: Equipment operated beyond limits	L3, L4
V-4: Reactor shut down	L4

In this paper, V-4 is used to illustrate how STPA-Sec identifies the potential vulnerable system states that can lead to the loss of power (L4). The shutdown of the reactor is a specific state. If the reactor is shutdown *and* if other worst-case environmental conditions are present, then one of the specific loss events (L4) can result. The reactor shutdown represents a vulnerable state that can yield a specific system loss that security analysts must guard against. However, the shutdown of the reactor may NOT necessarily lead to a loss of power production to the grid. For instance, there could be other auxiliary generators that could provide a small amount of backup power for a limited duration. Also, the reactor shutdown might occur during a time when the peak demand was low and capable of being met by other sources on the power grid.

The potential causes of the reactor shutdown are not addressed at this point, that is done later in the process. What is important is that the analysts identify the system's vulnerable states and their relationship to the specific losses.

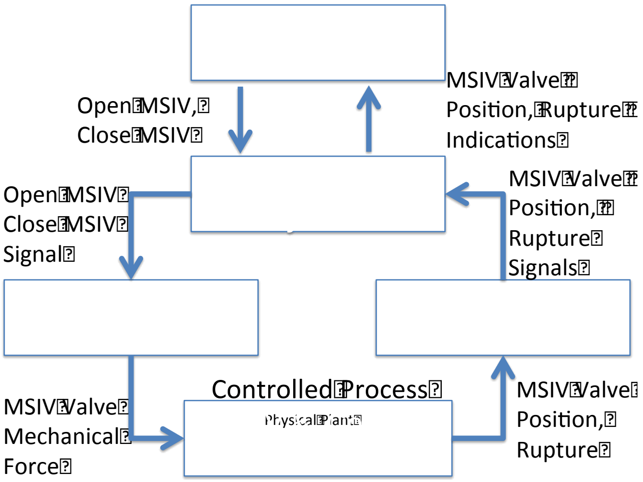
There is another, more subtle consideration. If defenders prevent a reactor shutdown, then L4 should not occur. Reactor shutdown is the state that must be controlled by analysts (strategy). This is different than trying to identify and counter all adversary actions or other potential disruptions (tactics).

The causality model that underlies STPA-Sec is based on control and hierarchy. Rather than attributing the loss to a single event or chain of events, STPA-Sec focuses on the development and maintenance of proper controls over the system itself. These controls take the form of constraints on system behavior. In the example, the system must be constrained from entering the vulnerable states (V1 to V4 in Table 1). These constraints extend beyond traditional security constraints, such as access control, to include a much broader set of systemic concerns and issues. The High Level Control Structure monitors and enforces constraints.

Creating a Model of the High Level Control Structure

The loss model underlying STPA-Sec is based on a lack of constraints, and developing the High Level Control Structure (HLCS) model provides a concise graphical specification of the functional controls in the system. The HLCS modeling is both iterative and decomposable into smaller sub-elements. Starting at a high level allows analysts to delve deeper where necessary, while simultaneously maintaining perspective on the functional whole. HLCS models include both control actions and feedback.

The HLCS model represents not only the technological, but the organizational sources of control. As a result, it provides a wider perspective on the potential actions available to assist in securing operations than might otherwise be available through other approaches. For space reasons, however, only the technical parts of the control structure are included in the nuclear power plant HLCS shown in Figure 2.



Open MSIV, Close MSIV
MSIV Valve Position, Rupture Indications
Open MSIV Close MSIV Signal
MSIV Valve Position, Rupture Signals
Controlled Process
Physical Plant
MSIV Valve Mechanical Force
MSIV Valve Position, Rupture

Figure 2. Simplified High Level Control Structure Model

The simplified HLCS model in Figure 2 has five basic components. The first component is the operator. In this example, the operator would be the individual charged with monitoring the overall status of the power plant. This individual would likely be located in a centralized control center.

The automated Digital Control System (DCS) is the second component of the HLCS model. The automation consists of the computer system that the operator uses to monitor the status of the actual plant and issue commands necessary to ensure safe operation of the system itself. The Digital Control System is responsible for interpreting operator inputs and providing signals to the actuator. In the example used here, only two signals are considered: *open isolation valve* and *close isolation valve*.

An actuator is the third component and resides at the cyber-physical junction. The actuator converts signals from the computer system into mechanical activity to open or close the physical isolation valve to the cooling system. A typical plant would consist of many of these valves, each executing different functions.

The physical plant is the fourth component. In this case, the isolation valve being controlled resides on the cooling system itself.

The fifth and final element of the example HLCS is the sensors. The sensors provide information to the control system about the actual condition of the plant. This information could include data on whether or not an emergency (rupture) exists, but also includes more obvious information such as the condition of the isolation valve (open or closed).

Identifying Unsafe/Unsecure Control Actions

The HLCS model combined with the other information in Step 1 sets the foundation for the remainder of the STPA-Sec analysis. Step 1 identified loss events, the vulnerabilities that can lead to these losses under worst-case environmental conditions, and the HLCS model that captures the control information that is transmitted throughout the system in order to allow it to accomplish its purpose. Control information is depicted in Figure 1 and consists of both control actions from the controller to the component directing and prohibiting specific activity and feedback from the component back to the controller on the status of the component. Control information is not limited to data and signals. Depending on the part of the control structure being considered, it can include regulations, operating procedures and other forms of guidance. It can also include feedback such as status updates or After Action Reports.

Regardless of format, the control information flows throughout the hierarchical structure and regulates system performance. Some vulnerabilities may only be evident if the connections or interactions between the various sub-systems are examined. For instance, a safety constraint in a train door controller may require that the door never be opened unless the train is at a station or an emergency exists. If a terrorist seeking to kill or injure individuals through a cyber attack is able to attack the door controller by mimicking the “emergency” state, then the controller’s logic might send the “open door” command and the train doors would open. If this command was sent with a loaded train operating at full speed, it is easy to see how loss of life or damage to the system could occur.

Note that the vulnerability is not in the controller itself, it may perform exactly as the software engineer desired it to (sending the “open door” command in case of an emergency). Unfortunately, a well-conceived and executed cyber attack in this example uses the controller’s logic to achieve a higher-level system loss of killing or injuring riders. There is no security violation in the individual system components. The vulnerability lies in the interactions between the components and only manifests under certain worst-case conditions.

The simple train door example highlights a key benefit of the approach, i.e., the focus on identifying and controlling vulnerable states that lead to systems-level losses, not component losses themselves. Step 2 of STPA-Sec identifies which control actions are vulnerable and under what circumstances.

As stated earlier, there are four types of potential unsafe/unsecure control actions:

1. Providing a control action leads to a hazard or exploits the vulnerability
2. Not providing a control action leads to a hazard or exploits a vulnerability
3. Providing control actions too late, too early, or in the wrong order leads to a hazard or exploits a vulnerability
4. Stopping a control action too soon or continuing it too long leads to a hazard or exploits a vulnerability.

Determining the potential causes of the unsafe/unsecure control actions is left to the next step. At this point, only the areas needing deep dives are identified, potentially leading to a more efficient analysis process.

Table 2 shows examples of each type of unsafe/unsecure control actions related to vulnerabilities from Table 1 in the nuclear power plant example.

Table 2. Potentially Unsecure Control Actions for *Close MSIV*

Control	**Unsafe/Unsecure Control Actions**			
Action	**Not Providing Causes Hazard**	**Providing Causes Hazard**	**Wrong Timing or Order Causes Hazard**	**Stopped Too Soon or Applied Too Long**
Close MSIV	Close MSIV not provided when there is a rupture in steam tube, leak in main feedwater, or leak in main steam line [V- 2, V-1, V-3]	Close MSIV provided when there is no rupture or leak [V-4]	Close MSIV provided too early (while steam pressure is high): Steam pressure may rise, trigger relief valve, abrupt steam expansion [V-2, V-3]	N/A

An important difference between STPA-Sec analysis and that of standard safety and security analysis is that the former identifies problematic situations beyond those resulting from simple confidentiality, integrity, and availability violations. STPA-Sec also highlights situations where the system behavior emerges from multiple interactions among the system components, all of which are behaving “correctly.”

Developing Security Requirements and Constraints

The previous steps in the analysis have proceeded in a top-down deliberate process. The first step provided the engineering information needed to examine and understand the functioning of the system. This information provides important context used in Step 2 to identify a list of unsafe/unsecure control actions.

The unsafe/unsecure control actions can be used to develop high-level safety and security requirements and constraints. As an example, a constraint on system behavior can be generated that a *Close MSIV command must never be provided when there is no rupture or leak.*

Identifying Causal Scenarios

The final step in the analysis is the one that bears the most resemblance to traditional security analyses. This step involves analyzing the existing physical and logical infrastructure to determine how the safety and security requirements and constraints identified in the previous step might be violated, that is, scenarios that can lead to losses.

Figure 3 shows potential problems in a control loop that can violate constraints and lead to a hazardous or vulnerable state. The analysis is performed by using these “clues” to generate viable scenarios.

The scenarios, in turn, can be used by system designers to create protection against the scenarios occurring or, if not possible, to limit damage from them. New types of causes may be used to assist in identifying security-related scenarios.

Traditional safety and security techniques, such as fault trees and attack trees, share STPA-Sec’s goal of identifying causal scenarios. The major difference is that STPA identifies a large set of scenarios, in particular, those not involving component failures or compromise but arising from interactions among components.

STPA-Sec also approaches scenario construction in a much more structured manner than simply assembling experts and having them brainstorm scenarios that could go wrong from scratch. After establishing the necessary appreciation for the system under evaluation, STPA-Sec's top-down, systems thinking process guides analysts through not only determination of the potential logical and physical component failures capable of producing the generated scenarios, but also the interaction failures (e.g. feedback delays, conflicting control actions), and the combination of component *and* interaction failures capable of producing the generated scenarios. Equipped with this deeper insight into technical and non-technical aspects of the system, security analysts are then better prepared to select and apply the most appropriate protection tactics.

In applying STPA-Sec Step 4 to the nuclear plant example, the goal of the step is to identify scenarios violating the constraint requiring that "close MSIV" not be issued when there is no rupture or leak present. The HLCS model shows that the operator issues the close MSIV control action to the automated digital control system based on feedback on the valve status (ruptured or not ruptured). If a rupture exists, the "close MSIV" control action should be given. If no rupture is actually present, then the previous steps of the analysis identify the fact that issuing the "close MSIV" control action introduces a vulnerability that can lead to a loss.

For the nuclear power plant example, one possible violation scenario involves the human operator receiving the wrong information about the rupture status of the system, that is, a scenario that causes the operator to believe the pipe has been ruptured when it has not or vice versa. Because the operator depends on the system feedback that flows from the physical cooling system to make the proper decision, any of the control flaws in Figure 3 between the controlled process and the controller could potentially cause the operator to believe a rupture exists when it does not and issue the close MSIV control action. Depending on the design of the specific hardware and software used in the plant, a very unsophisticated cyber attack might prove plausible. The attack need not necessarily change the operator's display or inject false data. It is possible that simply preventing the sensor from transmitting information to the DCS (generating a missing feedback problem) through a Denial-of-Service Attack might be sufficient to create the scenario if the DCS software was written to issue the rupture indication to the operator as a

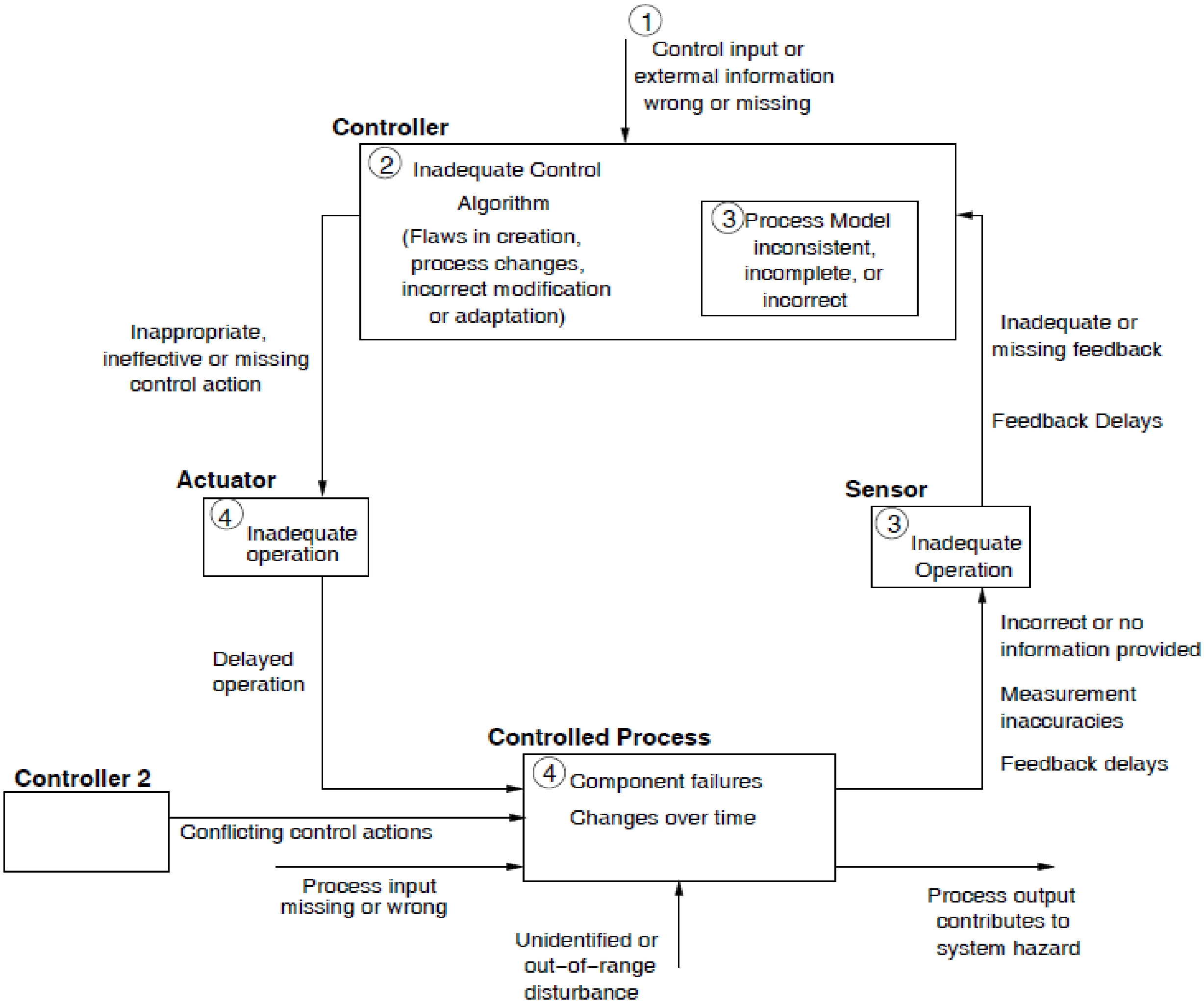

1
Control input or external information wrong or missing
Controller
2 Inadequate Control Algorithm
(Flaws in creation, process changes, incorrect modification or adaptation)
3 Process Model inconsistent, incomplete, or incorrect
Inappropriate, ineffective or missing control action
Inadequate or missing feedback
Feedback Delays
Actuator
4 Inadequate operation
Sensor
3 Inadequate Operation
Delayed operation
Incorrect or no information provided
Measurement inaccuracies
Feedback delays
Controlled Process
4 Component failures
Changes over time
Controller 2
Conflicting control actions
Process input missing or wrong
Unidentified or out-of-range disturbance
Process output contributes to system hazard

Figure 3. Control Loop Disruptions Leading to Hazardous / Vulnerable States

precaution in the case of a lost feedback signal.

Under most circumstances, this logic (reflected in the DCS process model) could be prudent, especially if the programmers thought that absence of a rupture status signal would only occur in situations where significant physical damage had already taken place. This assumption would necessitate closing the MSIV to isolate the main steam generator from the rest of the system. Clearly, the security analysts must assess the viability of the scenario to determine if deeper analysis or even reengineering is warranted. The probability of the feedback between the sensor and DCS being disrupted is not the question or focus. STPA-Sec reveals the fact that *if* the missing feedback problem arises, it *will* place the system in a hazardous/vulnerable state. This state occurs despite the fact that all components; DCS, actuators, sensors, MSIV, cooling system, and operator are all functioning normally. In this case, system analysts and operations experts will need to work together to apply their skill and judgment to determine which scenarios require even deeper technical analysis. Unlike other approaches, security analysts using STPA-Sec are not forced to depend on their creativity to generate the full list of scenarios from scratch. Rather, STPA-Sec helps illuminate loss scenarios in ever-increasing detail all the while allowing analysts to maintain their perspective on the larger system. In informal evaluations of STPA-Sec by security analysts and operations personnel, participants were surprised that using it helped them to consider threat scenarios that they had not thought of previously. A more scientific evaluation of the STPA-Sec is currently being performed.

STPA-Sec does not provide answers about what specific counter measures should be taken. Identifying protection mechanisms is and remains the realm of the security specialists. What STPA-Sec does provide is a potentially useful tool for identifying those scenarios that should be the focus of cyber security efforts to secure specific systems. Additionally, STPA-Sec provides traceability between the scenarios and the losses.

5. SUMMARY AND CONCLUSIONS

This paper has described how the tactical-level cyber security problem can be elevated from simply guarding the network to the higher-level problem of assuring the overall function of the enterprise. A new paradigm employing systems theory that has recently been introduced into safety is shown to apply to security as well as to safety.

In some ways this reframing will require redefining and expanding how security specialists think about their jobs. Perhaps one of the most important questions to ask about the current threat-based tactics model is whether or not organizations will devote resources to addressing system vulnerabilities that may not appear to be likely to be threatened. For example, STPA-Sec has shown how the particular set of conditions in the example could lead to a loss. However, if the scenario was presented just in terms of threat activity and absent the top-down traceability STPA-Sec provides, how likely are senior leaders to expend resources to address the vulnerability? Perhaps rather than framing the decision in terms of likelihoods that cannot be known, security specialists would be better off presenting decision makers with the scenarios that if acted upon will lead to a loss.

There will always be a need for good tactics. If current trends are any indication, the need for educated and skilled security analysts and engineers will only grow. Tactical models will continue to play an important role in security, yet strategy models must complement them. STPA-Sec will not replace good security practices, but it may improve them by providing a more clear focus for those designing and defending our software-intensive systems. The scope of the paper is limited in that it focuses on losses resulting from violations of integrity and availability but not confidentiality violations. We believe these can be handled equally well within this framework. Another feature of STPA-Sec, which was not covered, is its ability to assist analysts in examining how security constraints might degrade over time. See Leveson and Laracy for more on this topic [7] .

6. REFERENCES

[1] National Institute of Standards and Technology (NIST), *Special Publication 800-30 Revision 1 Guide for Conducting Risk Assessments*. U.S. Department of Commerce, September 2012

[2] Berg, G. G., Freeman, M. S. and Schneider, K. N. Analyzing the TJ Maxx Data Security Fiasco: Lessons for Auditors. *CPA Journal*, 78, 8 2008, 34-37.

[3] Checkland, P. *Systems Thinking, Systems Practice*. J. Wiley, Chichester Sussex ; New York, 1981

[4] Leveson, N. *Engineering a Safer World : Systems Thinking Applied to Safety*. MIT Press, Cambridge, Mass., 2011.

[5] Thomas, J. P., IV *Extending and Automating a Systems-theoretic Hazard Analysis for Requirements Generation and Analysis*. Massachusetts Institute of Technology, Massachusetts Institute of Technology, 2013.

[6] Weiss, J. *Protecting Industrial Control Systems from Electronic Threats*. Momentum Press; New York, 2010

[7] Laracy, J. and Leveson, N. "Applying STAMP to Critical Infrastructure Protection*" 2007 IEEE Conference on Technologies for Homeland Security*. IEEE, 2007.