This document is the result of a single chat with Gemini Pro 2.5 Deep Research mode on 6 September 2025. The response to the prompt below has not been modified. It was exported to Google Docs using Gemini's “Export to Docs” function.

Prompt (typos included):

> We're considering performing a STPA-Sec analysis on an example of an incidence response system. Take a look at the following link https://www.ncbi.nlm.nih.gov/books/NBK441863/.
>
> Find any other links that may relevant to understand how to perform the analysis on this system. We're specifically intersted in how to draw a control diagram. Is this possible?

Response:

Applying System-Theoretic Process Analysis for Security (STPA-Sec) to the Incident Command System: A Practical Guide to Control Structure Modeling and Vulnerability Analysis

Section 1: Foundations of System-Theoretic Analysis for Security

1.1 The Paradigm Shift: From Component Failure to Inadequate Control

For decades, the fields of safety and security engineering have been dominated by analytical methods rooted in a common paradigm: the analysis of component failures and event chains. Methodologies such as Fault Tree Analysis (FTA) and Failure Modes and Effects Analysis (FMEA) operate on the assumption that accidents and security breaches are the result of a linear sequence of failure events, typically initiated by a malfunctioning or broken component.[1] While valuable, this perspective is increasingly insufficient for understanding and preventing losses in modern, complex systems. Today's systems are characterized by intricate interactions, software-intensive control, and deep integration with human operators, creating scenarios where catastrophic losses can occur without any single component failing.[3]

Accidents are frequently caused by flawed system requirements, unforeseen and dysfunctional interactions between non-failing components, software errors, and human-machine interface flaws.[3]

In response to these limitations, a new paradigm has emerged based on systems theory. The Systems-Theoretic Accident Model and Processes (STAMP) reframes the cause of accidents, viewing them not as a chain of failure events but as the result of inadequate control and enforcement of safety and security constraints within the system's design and operation.[3] In this model, safety and security are treated as a dynamic

control problem rather than a static reliability problem.[1] The goal is to design a system where constraints on behavior are continuously maintained, preventing the system from migrating into a hazardous or insecure state.

This system-theoretic perspective is exceptionally well-suited for analyzing the Incident Command System (ICS). The ICS is, by its very nature, a socio-technical system designed to impose control on a chaotic and hazardous process—the incident itself.[10] Its effectiveness depends entirely on the ability of human controllers to make decisions and enforce constraints on the behavior of responders and resources. Therefore, analyzing the ICS through the lens of control theory provides a uniquely powerful framework for understanding its vulnerabilities.

1.2 Anatomy of the STPA Control Loop

System-Theoretic Process Analysis (STPA) is the hazard analysis methodology derived from the STAMP model. Its fundamental analytical building block is the feedback control loop, a concept borrowed from control theory that provides a structured way to model system behavior.[4] Every basic control loop consists of four primary components:

1. **Controller:** The component responsible for making decisions and issuing commands. A controller can be a human operator, a piece of software, or an organization.
2. **Controlled Process:** The system or process being managed by the controller.
3. **Control Actions:** The commands or interventions issued by the controller to influence the state of the controlled process.
4. **Feedback:** Information about the state of the controlled process that is sent back to the controller, allowing it to monitor the effects of its actions.

Within the controller itself lies the most critical and often most vulnerable element of the entire loop: the **Process Model**. The process model represents the controller's internal set of beliefs about the state of the controlled process.[4] This is not just a static picture but a dynamic understanding of how the system works, its current state, and how it is expected to change. The controller uses this process model, in conjunction with its control algorithm (the rules it follows), to determine which control actions to issue.

An accident or security breach occurs when a controller issues an unsafe or insecure control action. This almost always happens because the controller's process model is flawed—it is inconsistent with the actual state of the system. The feedback channel is the primary means by which the process model is updated, but if that feedback is missing, delayed, inaccurate, or deliberately manipulated, the process model becomes corrupted, leading to hazardous decisions.

The 2013 crash of Asiana Airlines Flight 214 provides a tragic and clear illustration of a process model failure.[4] The pilots (the controllers) believed the aircraft's autothrottle system (part of the controlled process) would automatically maintain airspeed. However, due to a specific sequence of actions, the autothrottle had entered a "HOLD" mode. The pilots' process model was incorrect; they believed the automation was active when it was not. Lacking clear feedback to contradict this flawed belief, they failed to manually increase thrust, leading to a stall and the subsequent crash. No component failed; the accident was the result of a flawed mental model leading to an unsafe control action (or, more accurately, the omission of a necessary one). This example underscores how STPA's focus on the process model provides a structured way to analyze the root causes of what is often simplistically labeled "human error."

1.3 Extending to Security: The Emergence of STPA-Sec

The principles of STPA, originally developed for safety, were naturally extended to address security. The resulting methodology, System-Theoretic Process Analysis for Security (STPA-Sec), adapts the control-based framework to account for the presence of an intelligent adversary.[6] The fundamental concepts remain the same, but the focus and terminology are adapted. The analysis shifts from identifying "Unsafe Control Actions" (UCAs) to identifying "Insecure Control Actions" (ICAs), which are control actions that could lead to a security loss if executed.[11]

A key contribution of STPA-Sec is its top-down, vulnerability-centric approach. Instead of starting with a list of known threats or attack vectors, the analysis begins by defining unacceptable system-level losses (e.g., loss of life, mission failure, data breach).[7] From these losses, the analysis identifies system vulnerabilities—weaknesses in the control structure that could allow the system to enter a state where a loss is possible.[6] This approach fundamentally redefines "cybersecurity" from a purely technological problem of defending assets against external threats to a broader systems engineering problem of designing and maintaining adequate control over system behavior. The vulnerability is not just a software bug but can be a flawed procedure, a missing feedback channel, or an ambiguous line of authority—any aspect of the system design that allows security constraints to be violated.

To systematically brainstorm how an adversary might exploit these control structure vulnerabilities, STPA-Sec can be powerfully combined with established threat modeling frameworks. One of the most common is STRIDE, an acronym for Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service, and Elevation of Privilege.[14] By considering each control action and feedback channel in the system model, analysts can use the STRIDE categories as guide words to develop credible attack scenarios.[6] For example, an attacker might

spoof a feedback signal to corrupt a controller's process model, *tamper* with a control action in transit, or execute a *denial of service* attack on a feedback channel to prevent the process model from being updated. This integration provides a structured link between high-level system vulnerabilities and specific, malicious causal scenarios.

1.4 The Four-Step STPA-Sec Analytical Process

The STPA-Sec analysis is conducted in a systematic, four-step process. This top-down approach allows analysts to manage the complexity of modern systems by beginning at a high level of abstraction and iteratively adding detail as the analysis progresses.[4] The four steps, which will form the structure for the remainder of this report, are:

1. **Define the Purpose of the Analysis:** This initial step establishes the foundation for the entire analysis. It involves identifying the unacceptable system-level losses to be prevented, defining the system hazards (for safety) or high-level vulnerabilities (for security) that could lead to those losses, and specifying the corresponding high-level safety or security constraints that the system must enforce.[11] This step also requires defining the boundary of the system being analyzed.
2. **Model the Control Structure:** In this step, the analyst creates a model of the system as a hierarchical control structure. This is typically visualized as a control structure diagram, which depicts the system's controllers, controlled processes, and the control actions and feedback pathways that connect them.[2] This model serves as the blueprint for the subsequent analysis.
3. **Identify Insecure Control Actions (ICAs):** Using the control structure model, the analyst examines each identified control action to determine how it could become insecure. This is done systematically using four guide words: a hazardous or insecure state results if the control action is (1) not provided, (2) provided when it should not be, (3) provided at the wrong time or in the wrong order, or (4) stopped too soon or applied for too long.[7]
4. **Identify Loss Scenarios:** For each ICA identified in the previous step, the analyst investigates the causal factors that could lead to it. This step explains *why* the controller might issue the insecure control action.[7] The scenarios can involve component failures, software errors, environmental disturbances, and, critically for STPA-Sec, intentional actions by a malicious adversary. The goal is to understand the weaknesses in the control system that allow the ICA to occur.

Section 2: The Incident Command System as a Hierarchical Control Structure

2.1 Step 1: Defining the Purpose of the Analysis for the ICS

The first step in any STPA-Sec analysis is to rigorously define its purpose and scope. This involves establishing the system boundary, identifying the unacceptable losses the system is designed to prevent, defining the high-level vulnerabilities that can lead to those losses, and specifying the overarching security constraints that must be enforced.

System Boundary
For this analysis, the Incident Command System (ICS) is the system under consideration. The boundary of this socio-technical system includes 10:

- **Internal Components:** The ICS command and general staff (e.g., Incident Commander, Section Chiefs), first responders organized into tactical units (e.g., Strike Teams, Task Forces), the formal communication systems (e.g., radios, data networks), and the resource management and planning processes (e.g., Incident Action Plan development).
- **External Elements:** Entities that interact with the system but are not under its direct control. These include the incident itself (e.g., a wildfire, a hazardous material spill), the public, the media, officials from other government agencies, and, critically from a security perspective, potential malicious actors seeking to disrupt the response.

Identifying System-Level Losses (L)
Based on the fundamental purpose of the ICS—to provide a standardized approach to the command, control, and coordination of emergency response—we can identify the highest-level unacceptable outcomes, or losses.12

- **L-1:** Loss of human life or serious injury to responders or the public.
- **L-2:** Failure to achieve primary incident objectives (e.g., fire is not contained, hazardous material is not secured, patients are not triaged and transported effectively).
- **L-3:** Significant damage to or loss of property and critical infrastructure.
- **L-4:** Loss of public trust and confidence in the response effort, leading to long-term societal impact.
- **L-5:** Misappropriation, waste, or inefficient use of critical and often scarce response resources.

Identifying System Hazards and Vulnerabilities (H/V)
Next, we define the system states that, under certain environmental conditions, will lead to one or more of the identified losses. These are the high-level hazards (for safety) and vulnerabilities (for security) that the system must control.12

- **H-1:** Responders are in a state of uncontrolled exposure to physical harm (e.g., fire, chemical exposure, structural collapse). (Leads to L-1).
- **H-2:** Incident stabilization efforts are inadequate, misdirected, or based on an incorrect understanding of the incident. (Leads to L-2, L-3).
- **H-3:** Critical resources (personnel, equipment, supplies) are unavailable, unassigned, or deployed incorrectly when needed. (Leads to L-2, L-5).
- **V-4:** The integrity, authenticity, or availability of incident-related information is compromised (e.g., via disinformation, data tampering, or communications jamming).

(Leads to all losses).

- **V-5:** The formal command authority of the ICS is subverted, spoofed, or bypassed, leading to uncoordinated and conflicting actions. (Leads to all losses).

Defining High-Level Security Constraints (SC)
Finally, we define the top-level system requirements by inverting the identified hazards and vulnerabilities. These security constraints represent the fundamental goals that the system's design and operation must continuously enforce to prevent losses.9

- **SC-1:** The ICS must prevent responder exposure to uncontrolled harm by maintaining situational awareness of all known hazards.
- **SC-2:** The ICS must ensure that all tactical operations are effective, correctly targeted, and based on the most accurate available information.
- **SC-3:** The ICS must maintain accurate, real-time tracking, allocation, and status of all managed resources.
- **SC-4:** The ICS must ensure the authenticity, integrity, and availability of all command, control, and coordination communications.

2.2 Deconstructing the ICS: Mapping Roles to Control Functions

A key realization when applying STPA-Sec to the ICS is that the system is already, by design, a real-world implementation of a hierarchical control system. Its foundational principles, such as "Chain of Command" and "Unity of Command," directly reflect the structure of a multi-layered control hierarchy.[10] This makes the ICS an ideal candidate for modeling with STPA. The task, therefore, is to translate the established ICS roles and processes into the formal language of control theory.

Mapping Controllers
The organizational structure of the ICS maps directly to the STPA concept of a hierarchy of controllers, each with a defined scope of responsibility.

- **High-Level Controller:** The **Incident Commander (IC)** or, in a multi-jurisdictional incident, the **Unified Command (UC)**. This controller is responsible for the overall strategic direction of the incident response.
- **Mid-Level Controllers:** The **Command Staff** (Public Information Officer, Safety Officer, Liaison Officer) and the **General Staff Section Chiefs** (Operations, Planning, Logistics, Finance/Administration). Each of these controllers manages a major functional area of the response.
- **Low-Level Controllers:** As the ICS organization expands, lower-level controllers are established. These include **Branch Directors**, **Division/Group Supervisors**, and **Strike Team/Task Force Leaders**. These controllers are responsible for executing specific tactical assignments.

Mapping Controlled Processes
Each controller in the hierarchy is responsible for managing a specific controlled process.

- **Incident Commander's Process:** The overall **Incident Management Process**, which encompasses the entire strategic response, setting of objectives, and high-level resource allocation.

- **Operations Section Chief's Process:** The execution of all **Tactical Field Operations**. This includes the actions of all responders directly engaged in controlling the incident.
- **Planning Section Chief's Process:** The **Information Collection and Incident Action Planning Process**. This involves gathering, analyzing, and disseminating information to create a coherent operational plan.
- **Logistics Section Chief's Process:** The **Resource Procurement and Delivery Process**. This includes ordering, tracking, staging, and delivering all personnel, equipment, and supplies needed for the response.

This mapping reveals a crucial aspect of the system. The official ICS structure described in documentation represents an *idealized* control structure.[10] However, real-world incidents are subject to immense pressure, pre-existing relationships between responders, and external influences (e.g., political pressure, media reporting) that can create informal or "out-of-band" communication and influence channels. The original FIRESCOPE study, which led to the creation of the ICS, identified "freelancing"—individuals or groups acting outside the formal command structure—as a major problem to be solved.[10] From a security standpoint, these informal channels represent high-risk vulnerabilities. An adversary could exploit them to bypass the formal chain of command, introduce disinformation, or influence decisions without being subject to the checks and balances of the formal ICS process.[5] Therefore, a comprehensive STPA-Sec analysis must model not only the formal hierarchy but also these informal and external control and feedback pathways to identify more subtle, socio-technical vulnerabilities.

Furthermore, this deconstruction highlights the central role of the **Incident Action Plan (IAP)**. The IAP, which answers the core questions of "What do we want to do?" and "Who is responsible?", is more than just a document.[10] In STPA terms, it is the physical embodiment of the high-level controller's Process Model and the primary mechanism for propagating control actions through the hierarchy. The IAP codifies the beliefs of the Command and Planning sections about the incident's current state and the desired future state. Its dissemination and execution represent the primary "Control Action" from the strategic levels of the ICS to the tactical levels. This means that protecting the integrity of the IAP development and dissemination process is a paramount security objective. An attack that tampers with the IAP before it is distributed could misdirect the entire operational response, leading directly to Hazard H-2 and multiple system losses.[14] The analysis must therefore focus intensely on the control loops surrounding the creation, approval, and communication of the IAP.

Section 3: A Practical Guide to Drawing the ICS Control Structure Diagram

The central query regarding the feasibility of creating a control structure diagram for the

Incident Command System can be answered with a definitive "yes." Not only is it possible, but the inherent hierarchical nature of the ICS makes it an exceptionally suitable subject for this type of modeling. This section provides a practical, multi-layered guide to constructing such a diagram, moving from a high-level strategic view down to tactical details.

3.1 Principles of Hierarchical Control Modeling

Constructing a useful control structure diagram requires adherence to two key principles:

1. **Top-Down Abstraction:** The most effective way to model a complex system is to start at a high, abstract level and iteratively add layers of detail.[4] This approach prevents the analyst from becoming overwhelmed by low-level specifics at the outset and ensures that the overall control philosophy and major interactions are captured first. The initial diagram might show only the highest-level controller and the overall process it manages. Subsequent iterations "open up" these abstract boxes to reveal the sub-controllers and sub-processes within them.
2. **Annotating for Security:** For a standard STPA safety analysis, the diagram focuses on functional control. For STPA-Sec, the diagram must also serve as a map of the system's security landscape. This is achieved by annotating the control actions and feedback channels to identify potential vulnerabilities. Using a framework like STRIDE, a radio communication channel could be annotated as vulnerable to *Spoofing* (an attacker impersonating a valid user) and *Information Disclosure* (eavesdropping), while a digital planning document could be marked as vulnerable to *Tampering*.[6] This annotation transforms the diagram from a simple flowchart into a rich analytical tool for identifying attack surfaces.

3.2 Level 1: The Strategic Command Control Structure

The highest level of the ICS control structure focuses on the strategic leadership of the incident. At this level, the entire response effort is treated as a single controlled process.

- **Controller:** Incident Commander (IC) / Unified Command (UC).
- **Control Actions:** The primary directives issued by the IC/UC that shape the overall response. These include: Set Incident Objectives, Approve Incident Action Plan (IAP), and Allocate Strategic Resources.
- **Controlled Process:** The overall Incident Management Process.
- **Feedback:** The information channels the IC/UC uses to update their process model. These include: Situation Status Updates (typically from the Planning Section), Resource Status Reports (from the Logistics Section), and Public/Media Reaction (relayed by the Public Information Officer).
- **Process Model:** The IC/UC's process model is their comprehensive understanding of the incident. It contains their beliefs about the nature, scope, and predicted behavior of the incident; the capabilities and status of available resources; the effectiveness of the current response strategy; and the external political and public context. A flaw in this high-level process model can lead to a fundamentally misdirected response.

3.3 Level 2: The General Staff Functional Control Loops

The next step is to drill down into the Incident Management Process block from the Level 1 diagram. This reveals the functional control loops managed by the General Staff Section Chiefs. Each section chief acts as a controller for their respective functional area.

Example: The Operations Section Control Loop

- **Controller:** Operations Section Chief.
- **Control Actions:** Assign Tactical Missions to Divisions/Groups, Request Resources via Logistics, Modify Tactical Plans.
- **Controlled Process:** Tactical Field Operations (the collective actions of all responders on the front lines).
- **Feedback:** Progress Reports from field units, Scout/Reconnaissance Information, Direct Observation.

Example: The Planning Section Control Loop

- **Controller:** Planning Section Chief.
- **Control Actions:** Develop Draft IAP, Disseminate Incident Information, Establish Information Collection Plan.
- **Controlled Process:** Information Collection and IAP Development Process.
- **Feedback:** Field Observations from Operations, Resource Status from Logistics, External Data Feeds (e.g., weather forecasts, GIS data).

This level of modeling reveals that functions often considered "support," such as Planning and Logistics, are in fact critical controllers of their own processes. A traditional view of the ICS might focus almost exclusively on the command and control aspects of the Operations section. However, the control structure diagram makes it visually explicit that the Planning Section controls the vital process of information gathering and plan creation, while the Logistics Section controls the equally vital process of resource mobilization. This demonstrates that an attack on the Logistics control loop (e.g., submitting fraudulent resource requests or tampering with inventory data) could cripple the entire response without ever directly affecting an operational unit. Similarly, an attack that feeds disinformation into the Planning section's control loop could result in a fundamentally flawed IAP that misdirects the entire operational effort. The diagram forces the realization that security resources and controls must be applied with equal rigor to these "support" functions, as they represent critical and vulnerable control points in the overall system.

3.4 Level 3: The Tactical Control Loops

Further refinement involves drilling down from a mid-level controller, such as the Operations Section Chief, to the tactical level where directives are executed.

- **Controller:** Division Supervisor (responsible for a specific geographic area).
- **Control Actions:** Direct Strike Team Actions (e.g., "Construct fireline from point A to

point B"), Report Progress to Operations Chief, Request Immediate Air Support.
- **Controlled Process:** A specific tactical objective, such as Fire Suppression in Sector A.
- **Feedback:** Visual Observation of Fire Behavior, Radio Reports from Team Leaders, Updates on Team Status (e.g., fatigue, water supply).

This level of detail is where specific human-machine interactions and communication protocols can be analyzed for vulnerabilities. For example, the feedback loop of "Radio Reports from Team Leaders" is a critical channel that is highly vulnerable to spoofing, jamming, or simple misinterpretation under stress.

3.5 The Integrated Hierarchical Control Diagram

The final step in modeling is to assemble these different levels into a single, integrated diagram. This diagram visually represents the flow of control and authority down through the hierarchy (from IC to Section Chief to Division Supervisor) and the flow of feedback and information up through the hierarchy. This comprehensive model serves as the essential foundation for the subsequent steps of the STPA-Sec analysis: identifying insecure control actions and their causal scenarios. The diagram makes it clear how a single human, such as a Section Chief, often serves as a critical nexus for multiple control loops. They act as a controller for the tactical process below them while simultaneously being part of the controlled process from the perspective of the Incident Commander above them. This dual role makes them a single point of failure and a high-value target for an adversary seeking to cause disruption through manipulation, disinformation, or a denial-of-service attack (e.g., overwhelming them with false reports). This visual representation underscores that personnel security, training in disinformation resilience, and robust procedures for succession are not merely administrative functions but are critical security controls for the system.

To facilitate the understanding and application of this modeling approach, the following table provides a direct translation between the familiar terminology of the Incident Command System and the formal components of the STPA control structure.

ICS Element	STPA Component	Example
Incident Commander	High-Level Controller	Makes strategic decisions based on an overall picture of the incident.
Operations Section Chief	Mid-Level Controller	Manages all tactical operations to achieve the objectives set by the IC.
Tactical Field Operations	Controlled Process	The set of physical actions

		being performed by responders on the ground (e.g., firefighting, search and rescue).
Incident Action Plan (IAP)	Control Action (High-Level)	The set of formal directives from Command to the rest of the organization for an operational period.
Radio Command to a Team	Control Action (Low-Level)	A specific, real-time directive to perform a task (e.g., "Engine 5, advance to the third floor").
Situation Report	Feedback	Information from a lower level to a higher level about the state of the controlled process (e.g., "Third floor is clear").
IC's understanding of the event	Process Model	The IC's mental model, formed from all available feedback, which drives strategic control actions.

Section 4: Identifying Insecure Control Actions and Loss Scenarios within the ICS

With a robust model of the ICS control structure in place, the analysis proceeds to Steps 3 and 4: systematically identifying potential insecurities in the system's control actions and then developing plausible scenarios that could cause them. This is where the abstract model is used to uncover concrete, actionable vulnerabilities.

4.1 Step 3: Identifying Insecure Control Actions (ICAs)

The objective of this step is to examine each critical control action identified in the control structure diagram and determine how it could lead to a hazardous or insecure state. This is accomplished by applying four specific guide words or types of control flaws to each action.[7]

This process provides a comprehensive taxonomy for control-related vulnerabilities that goes far beyond simple categories like "data breach" or "denial of service." It offers a functional or behavioral classification of failure. For instance, a technical "Denial of Service" attack is a cause, but its effect on the control system might be that a required control action is provided too late (Type 3) or not at all (Type 1). This shift in taxonomy allows analysts to rigorously trace the impact of a technical attack all the way to a system-level loss, effectively bridging the cyber and physical domains.[5]

Let us analyze the control action **Operations Chief Assigns Tactical Mission** from the Level 2 control loop:

- **ICA Type 1 (Not Provided when required):** The Operations Chief fails to assign a team to a critical, rapidly escalating sector of a wildfire. This omission of a necessary control action could be due to information overload, a communications failure preventing a request from being received, or a deliberate disruption. This directly leads to Hazard H-2 (Incident stabilization efforts are inadequate).
- **ICA Type 2 (Provided, but causes a hazard):** The Operations Chief assigns a search and rescue team to enter a building that is structurally unsound. This hazardous control action is provided based on the belief (process model) that the building is safe. This leads directly to Hazard H-1 (Responders are in a state of uncontrolled exposure to physical harm).
- **ICA Type 3 (Provided too early, too late, or in the wrong order):** During a wildfire, the Operations Chief assigns an aerial water-drop mission *before* confirming that all ground teams have cleared the target area. This incorrectly sequenced control action creates a direct threat to the safety of the ground crews, leading to Hazard H-1.
- **ICA Type 4 (Stopped too soon or applied too long):** The Operations Chief prematurely reassigns a team from a sector of the fire that has not yet been fully secured and mopped up. The control action (suppression in that sector) is stopped too soon, allowing for a flare-up that threatens the entire fireline, leading to Hazard H-2.

4.2 Step 4: Developing Causal Loss Scenarios

After identifying a list of potential ICAs, the next step is to move from "what" could go wrong to "why" it might go wrong. This involves developing causal loss scenarios for each ICA. A scenario explains the sequence of events and causal factors that could lead a controller to issue an insecure control action. This is the stage where both unintentional system flaws and deliberate, malicious attacks are considered.[7]

Let us expand on the ICA identified above: **Operations Chief assigns a team to enter a structurally unsound building** (ICA Type 2).

- **Underlying Causal Factor:** The core reason for this ICA is a **Process Model Flaw**. The Operations Chief's process model is incorrect; they believe the structure is safe when, in reality, it is not. The scenarios below explore different ways this process model could become corrupted.

- **Scenario 4.2.1 (Unintentional Cause - Flawed Feedback):** A reconnaissance team assesses the building and reports back via radio. However, the radio message is garbled due to interference, and "structurally... unsound" is misinterpreted by the communications unit as "structurally sound." The incorrect information is passed to the Operations Chief, who updates their process model accordingly. The flaw lies in the low integrity of the feedback channel.
- **Scenario 4.2.2 (Intentional/Malicious Cause - Spoofing):** A malicious actor, equipped with a cloned radio and knowledge of ICS procedures, monitors the reconnaissance team's frequency. After the real team is out of communication, the attacker transmits a fraudulent "all clear, structure is safe for entry" message, perfectly mimicking the voice and call sign of the recon team leader. The Operations Chief receives this spoofed feedback, has no reason to doubt its authenticity, and updates their process model with deliberately false information. This leads directly to the hazardous control action.
- **Scenario 4.2.3 (Intentional/Malicious Cause - Tampering):** The incident command post uses a digital mapping system that displays building status. An insider threat (e.g., a disgruntled IT support contractor) or an external attacker who has gained network access alters the database entry for the building, changing its status flag from "UNSAFE - DO NOT ENTER" to "ASSESSED - CLEAR." The Operations Chief consults the map (a form of feedback), sees the "CLEAR" status, and their process model becomes flawed. The hazardous assignment is then made.

These scenarios reveal that the most dangerous threats are often not those that "break" the system (e.g., crashing a server), but those that manipulate the system into using its own legitimate rules and procedures to behave in a hazardous way. In Scenarios 4.2.2 and 4.2.3, the Operations Chief acted perfectly rationally and in full compliance with ICS procedure based on the information they possessed. The system itself, with its trusted communication channels and data sources, was turned into the weapon. This demonstrates how STPA-Sec uncovers emergent, systemic insecurities that arise from the interactions between components, which are often missed by traditional security analyses that focus on the failure or compromise of a single component.[3]

4.3 Analyzing the Human Controller and Process Model Flaws

A critical part of the STPA-Sec analysis, especially for a human-centric system like the ICS, is a deeper examination of the vulnerabilities of the human controller's process model. The human mind is not a computer; its ability to maintain an accurate process model is influenced by a wide range of factors that an intelligent adversary can seek to exploit.

- **Cognitive and Environmental Factors:** Stress, fatigue, and information overload are inherent in any major incident response. These conditions degrade cognitive performance, making it harder for a human controller to process complex information, spot inconsistencies in feedback, and maintain an accurate mental model. A poorly designed user interface on a command-and-control software platform can exacerbate this by presenting information in a confusing or ambiguous way.

- **Adversarial Exploitation:** A sophisticated adversary understands these human factors and will design attacks to exploit them. For example, a denial-of-service attack against an Incident Commander might not aim to crash their computer, but rather to flood them with a high volume of low-priority but plausible-sounding alerts and notifications. The goal is to induce a state of information overload and cognitive tunneling, degrading their ability to maintain an accurate high-level process model and making them more susceptible to a subsequent, more targeted piece of disinformation. The attack targets the human's cognitive limits, using the system's own information channels as the vector.

Section 5: Deriving Security Requirements and Strategic Recommendations

The ultimate purpose of analysis is to guide design and improve the system. The final phase of the STPA-Sec process involves translating the analytical findings from the identified insecure control actions and loss scenarios into concrete, actionable outputs. This closes the loop, demonstrating how STPA-Sec functions not just as an analytical tool but as a powerful engine for generating robust, security-guided design requirements.

5.1 From Loss Scenarios to Enforceable Constraints

Every loss scenario identified in the analysis points to a missing or inadequately enforced constraint within the system's control structure. The goal is to define new constraints or refine existing ones to prevent the scenario from occurring in the future. This process ensures clear traceability from a potential real-world failure back to a specific design requirement.[9]

Let us revisit **Scenario 4.2.2**, where a malicious actor uses a spoofed radio message to deceive the Operations Chief into issuing a hazardous command.

- **Identified Flaw:** The analysis reveals that the control loop between the field-based Reconnaissance Team (sensor) and the command-post-based Operations Chief (controller) lacks a mechanism to ensure the *authenticity* of the feedback provided. The controller implicitly trusts the information without a reliable means of verification.
- **New Security Constraint (SC-4.1):** The Operations Chief must be able to verify the authenticity of all operational reports from field units before using that information to update the process model for safety-critical tactical decisions.
- **New Security Constraint (SC-4.2):** The communications system must provide a mechanism for authenticating the identity of the sender for all transmissions containing operational reports or directives.

5.2 Developing Specific Security Requirements

High-level constraints must be translated into specific, testable requirements that can be implemented across the different layers of the system: people, processes, and technology.

This ensures that the abstract safety or security goal is realized through concrete measures.

Continuing with the constraints derived from the spoofing scenario:

- **Procedural Requirement (for SC-4.1):** All field units must use a pre-briefed, non-trivial challenge-response authentication code word when reporting safety-critical information (e.g., a building is "all clear"). This code word should be changed for each operational period. This is a low-tech but effective control that can be implemented immediately.
- **Technical Requirement (for SC-4.2):** All future radio systems and data communication platforms procured for incident response must implement strong, end-to-end encryption with digital signature capabilities for non-repudiation and sender authentication. This provides a high-assurance technical control for future systems.
- **Training Requirement (for SC-4.1):** All personnel designated for command and general staff roles must receive regular training on adversary tactics, including social engineering, disinformation, and communication spoofing. This training must include drills on procedures for verifying suspicious or unexpected communications before acting on them.

The security requirements derived from STPA-Sec are inherently more resilient than those from many traditional methods. This is because they are tied to the system's fundamental control functions, not to a specific technology or threat that will inevitably become obsolete. A traditional requirement might be "Install antivirus software version X," which becomes useless against a new threat. The STPA-Sec-derived requirement, "The controller must verify the integrity of feedback before issuing a control action," is a timeless control principle. This principle can be implemented in various ways, and if one implementation (e.g., a specific encryption algorithm) becomes obsolete, the underlying requirement to enforce the constraint remains valid and drives the search for a new implementation. This focus on fundamental principles helps build systems that are "secure by design" by promoting resilience over simple, brittle defenses.[6]

5.3 Strategic Recommendations for Integrating STPA-Sec into the ICS Lifecycle

To realize its full potential, STPA-Sec should not be treated as a one-time analytical activity. Instead, it should be integrated into the entire lifecycle of emergency management systems and doctrine.[2]

- **Preparedness and Planning:** STPA-Sec should be used proactively during the planning phase to design more resilient incident management protocols, communication plans, and technological systems. The loss scenarios identified during the analysis can become powerful inputs for designing realistic training exercises and drills, allowing responders to practice identifying and responding to sophisticated systemic threats.
- **Incident Response:** While a full, formal STPA-Sec analysis is not feasible in the midst of a real-time incident, the *principles* of systems thinking it teaches are invaluable.

Decision-makers trained in STPA are more likely to constantly question the validity of their own process models, actively seek out disconfirming evidence, and think critically about the integrity of their feedback channels, leading to more robust real-time decision-making.
- **Post-Incident Analysis (CAST):** For incidents that do occur, a complementary methodology called Causal Analysis based on STAMP (CAST) can be used.[6] CAST is the STAMP-based tool for accident/incident investigation. It guides investigators to analyze the entire socio-technical system to understand why the existing controls were ineffective, moving beyond a superficial root cause to identify systemic flaws in design, procedures, and organizational culture.
- **Continuous Improvement:** The findings from proactive STPA-Sec analyses and reactive CAST investigations should be fed into a continuous improvement loop. This ensures that lessons learned are used to systematically refine ICS doctrine, update training curricula, and inform requirements for new technology acquisition.

5.4 Conclusion: The Value of a Systems-Theoretic Approach

This analysis has demonstrated that it is not only possible but highly valuable to model the Incident Command System using the STPA-Sec framework. By reframing security as a control problem, this approach provides a structured, top-down methodology for analyzing the complex interactions between people, processes, and technology that define modern emergency response.

The application of STPA-Sec to the ICS reveals deep, systemic vulnerabilities—such as those related to flawed process models, unauthenticated feedback, and the exploitation of human cognitive limits—that are often missed by traditional risk assessment methods focused on component failure. More importantly, the analysis generates effective, multi-layered security requirements that are traceable, testable, and inherently resilient because they are based on fundamental control principles.

However, fully embracing a systems-theoretic approach requires more than just learning a new analytical technique. It necessitates a significant cultural shift within an organization. Traditional incident reviews often conclude by assigning blame to "human error" or "component failure".[1] STPA and CAST force deeper questions: "Why did that error make sense to the person at the time?" and "What aspects of the system's design allowed this component failure to propagate into a system-level loss?".[22] This shifts the focus from blaming individuals to understanding and improving the design of the system in which they operate—the training they receive, the tools they use, the procedures they must follow, and the information they are given. In an era of increasing operational complexity and sophisticated adversaries, ensuring the security and safety of our critical response systems demands that we move beyond component-level thinking and foster an organizational culture dedicated to the holistic, systems-theoretic control of risk.

Works cited

1. A Novel System-Theoretic Matrix-Based Approach to Analysing Safety and Security of Cyber-Physical Systems - Coventry University, accessed September 6, 2025, https://pureportal.coventry.ac.uk/files/45906792/Binder1.pdf
2. A Novel System-Theoretic Matrix-Based Approach to Analysing Safety and Security of Cyber-Physical Systems - MDPI, accessed September 6, 2025, https://www.mdpi.com/2673-4001/2/4/30
3. STPA Guide - VTT's Research Information Portal, accessed September 6, 2025, https://cris.vtt.fi/files/98296189/Complete_with_DocuSign_2024-1-2_STPA_guide_F.pdf
4. System-Theoretic Process Analysis (STPA) - Flight Test Safety ..., accessed September 6, 2025, https://flighttestsafety.org/images/STPA_intro.pdf
5. A Case Study in the Application of STPA-sec and CHASSIS for Socio-Technical Cyber Security Risk Management in Health Care from Developing Nations | Request PDF - ResearchGate, accessed September 6, 2025, https://www.researchgate.net/publication/366777694_A_Case_Study_in_the_Application_of_STPA-sec_and_CHASSIS_for_Socio-Technical_Cyber_Security_Risk_Management_in_Health_Care_from_Developing_Nations
6. Threat analysis using STRIDE with STAMP/STPA - CEUR-WS, accessed September 6, 2025, https://ceur-ws.org/Vol-2809/WESPr-18_02.pdf
7. Limitation and Improvement of STPA-Sec for Safety and ... - Zenodo, accessed September 6, 2025, https://zenodo.org/record/183288/files/STPA-SEC.pdf
8. Limitation and Improvement of STPA-Sec for Safety and Security Co-analysis | Request PDF, accessed September 6, 2025, https://www.researchgate.net/publication/307585878_Limitation_and_Improvement_of_STPA-Sec_for_Safety_and_Security_Co-analysis
9. Formal analysis of safety and security requirements of critical systems supported by an extended STPA methodology. - ePrints Soton - University of Southampton, accessed September 6, 2025, https://eprints.soton.ac.uk/406432/1/S4CIP17_paper_11.pdf
10. EMS Incident Command System - StatPearls - NCBI Bookshelf, accessed September 6, 2025, https://www.ncbi.nlm.nih.gov/books/NBK441863/
11. (PDF) TRANSLATING THE STPA-SEC SECURITY METHOD INTO A MODEL-BASED ENGINEERING APPROACH - ResearchGate, accessed September 6, 2025, https://www.researchgate.net/publication/383858664_TRANSLATING_THE_STPA-SEC_SECURITY_METHOD_INTO_A_MODEL-BASED_ENGINEERING_APPROACH
12. Data-flow-based adaption of the System-Theoretic Process Analysis for Security (STPA-Sec), accessed September 6, 2025, https://pmc.ncbi.nlm.nih.gov/articles/PMC7959614/
13. [2006.02930] Data-Flow-Based Extension of the System-Theoretic Process Analysis for Security (STPA-Sec) - arXiv, accessed September 6, 2025, https://arxiv.org/abs/2006.02930
14. BASIC INTRODUCTION TO STPA FOR SECURITY (STPA-SEC) July 22, 2020, accessed September 6, 2025,

http://psas.scripts.mit.edu/home/wp-content/uploads/2020/07/STPA-Sec-Tutorial.pdf
15. A comparative risk analysis on CyberShip system with STPA-Sec, STRIDE and CORAS - DTU Orbit, accessed September 6, 2025, https://orbit.dtu.dk/files/315004918/1_s2.0_S0167404823000895_main.pdf
16. A Comparative Risk Analysis on CyberShip System with STPA-Sec, STRIDE and CORAS - arXiv, accessed September 6, 2025, https://arxiv.org/pdf/2212.10830
17. Resource Center - VWAY ENG, accessed September 6, 2025, https://eng.vway.co.kr/vway-introduce/resource-center/?pageid=2&uid=5&mod=document
18. Enriching Systems Theory Based Cyber-Security and Safety Analysis Using Stakeholder Value Networks - SAE International, accessed September 6, 2025, https://www.sae.org/publications/technical-papers/content/2020-01-0143/
19. EVALUATING CYBERSECURITY TOOLS FOR USE IN AN EARLY STPA-SEC FLOW - NDIA Conference Proceedings, accessed September 6, 2025, https://ndia.dtic.mil/wp-content/uploads/2023/systems/Wed_1552546_Newport.pdf
20. Review of Systems-Theoretic Process Analysis (STPA) Method and Results to Support NextGen Concept Assessment and Validation - MIT Lincoln Laboratory, accessed September 6, 2025, https://www.ll.mit.edu/sites/default/files/publication/doc/2018-12/Harkleroad_2013_ATC-427.pdf
21. Introduction to STPA-Sec Method by Steven King on Prezi, accessed September 6, 2025, https://prezi.com/p/dwxcq22pqxhn/introduction-to-stpa-sec-method/
22. STAMP Workshop | MIT Partnership for Systems Approaches to Safety and Security (PSASS), accessed September 6, 2025, https://psas.scripts.mit.edu/home/stamp-workshops/