Using System Theoretic Process Analysis to Advance Safety in LLM-enabled Software Systems

JANUARY 17, 2024

David Schulker
Senior Data Scientist, CERT Division
LLM-enabled software systems require new approaches to make them more robust and secure

- Novel attack surfaces
- Unpredictable components
- Scope of inputs and model parameters
- Opaque software stacks for implementation

Start with a concrete LLM use case: *operational planning*

Military commanders and staffs convey directives through detailed planning orders

• Length of a single order = 100+ pages

Could we save valuable planning time with a question-answering (QA) system?
Abundant advice and tools for AI safety are available…

**Policy and Principles**
- DoD RAI strategy
- Private/public sector principles documents

**Metrics**
- HELM project
- BIG-bench

**Tools**
- CDAO RAI toolkit

**Software practices**
- Static/dynamic analysis
- Testing practices

…but what is relevant to safety and security *in my use case?*
System theoretic process analysis (STPA) could be an effective approach for designing safe systems

What is it?
A hazard analysis technique to prospectively evaluate risk in safety critical systems

Why is it relevant?
STPA was designed for...
• Complex systems
• Hardware + software + humans
• Losses result despite proper functioning of all components
The STPA process seeks to design constraints for components that prevent critical losses.

- Loss that must be prevented
- Hazardous state that could lead to loss
- Component behaviors that produce hazard
- Constraints that prevent behaviors
The STPA process seeks to design constraints for components that prevent critical losses

Loss that must be prevented

Planning decision based on faulty info

Hazardous state that could lead to loss

System returns wrong answer

Component behaviors that produce hazard

- Embedding model
- Context retrieval
- LLM response

Constraints that prevent behaviors

- Data curation
- Automated testing
- Retraining interval
STPA outputs help developers understand how safety constraints reduce hazards and losses.
By linking unsafe component behaviors to losses, STPA helps developers understand metrics and gaps.
By linking unsafe component behaviors to losses, STPA helps developers understand metrics and gaps.

Not every class of metrics is relevant to the losses that must be prevented.
By linking unsafe component behaviors to losses, STPA helps developers understand metrics and gaps.

LLM literature does not help us measure cyber and information security risks!
We are drawing on STPA to help LLM communities navigate cybersecurity (and other) unknowns.

Which metrics do we use, and how do we interpret them?

How do we include all dimensions of risk in our evaluation?

How do we design safety constraints?

How do we implement principles and tools in a maintainable way?