Mutual Monitoring Update

Progress Report 2

A.J. Stein

astein38@gatech.edu

Problem

- Cloud service providers are everywhere, but their security is complex.
- Continuous monitoring is complex, tedious multi-party process.
  - FedRAMP is a very public example.
- Everything is manual.
- Information flows in one direction, not to everyone who needs it.
- Is there a better way?

Solution

- Examine FedRAMP as a representative example.

- Re-imagine a new system with:
  - a new technological approach; and
  - change the process and how people interact.

- Estimate improved cost and performance.

Solution Deliverables

🎁 1️⃣ Critical analysis of FedRAMP

🎁 2️⃣ Architecture specification

🎁 3️⃣ prototype code for mutual monitoring transparency service

🎁 4️⃣ a quantitative cloud security measurement framework

Status

- 🎁 1️⃣ I completed more desk research and fleshed out my deliverables' outline.
- 🎁 2️⃣ I made substantial progress on the architecture specification draft.
  - I have still some work to do, but finishing beginning of Week 5.
  - I developed automation for publishing drafts and final specification.
  - I tried customizing popular tools used by IETF authors, but it wasn't a good.
  - I will leverage this for deploying prototype code, mentioned below.
- 🎁 3️⃣ I coded the initial prototype code and created initial deployment automation.

Status

Architecture specification draft: conmotion.netlify.app

Source code and containers for transparency service: github.com/aj-stein/conmotion

Challenges

- Implementing against draft specifications
  - Balancing ambiguous tradeoffs

  - Understanding under-documented algorithms foundational to the design

  - Filling in gaps outside of the specification's scope

- Balancing developing development; research; and writing on a schedule
  - Testing and ironing out kinks is frustrating, but necessary

Questions

1. How do I use polls and pseudonymonous content on social media? Should I?

2. How do I use self-hosted or hosted LLMs do sentiment analysis? Should I?

Next Weeks' Focus

1. 🎁 1️⃣ Complete outline of FedRAMP critical analysis.

2. 🎁 1️⃣ Start draft of FedRAMP critical analysis, get feedback from advisors.

3. 🎁 2️⃣ Complete the architecture specification, get feedback from advisors.

4. 🎁 3️⃣ Complete the submissions API, the core component of the prototype.

Feedback

- You can provide feedback in multiple ways.

- Post in the class discussion board in Canvas.

- Open issues in my GitHub repo at github.com/aj-stein/practicum/issues.

Conclusion

Et fin.

(See you all again in two weeks.)