Policy-based Access to Powerful Models

Summary

As machine learning models get more powerful, restricting query access based on a safety policy becomes more important. Given a setting where a model is stored securely in a hardware-isolated environment, access to the model can be restricted based on cryptographic signatures. Policy-based signatures allow signing messages that satisfy a pre-decided policy. There are many reasons why policy enforcement should be done cryptographically, including insider threats, tamper resistance and auditability. This project leverages existing cryptographic techniques and existing discourse on AI/ML safety to come up with reasonable policies and a consequent policy-based access model to powerful models.

The non-summary

Recent work [1] has shown that safety training atop LLMs can be disabled for a fine-tuning budget of $200. One simple way to prevent this is by not open-sourcing model weights. However, even allowing the fine-tuning of closed-source models such as GPT 3.5 results [2] in serious safety degradation. Given this, there is a clear motivation to come up with a framework where there is a provable guarantee that parties with malicious intent can not access powerful models in an unrestricted manner. The idea stated simply is to use the provable, cryptographic guarantees of policy-based signatures (PBS). In the PBS framework, a signer can only sign messages conforming to some authority-specified policy. Thus, one can restrict access to models based on pre-specified safety policies and any adversary would have to break the underlying cryptography to overcome the policy. Each query to a powerful model has to be sent along with a policy-based signature on the query using the user's key. Queries are only answered if and only if the query q satisfies the policy predicate P, i.e., $P(q) = 1$.

A natural question one might ask is the following: *why not enforce these policies through a simple backend system that restricts model access?* In a world where the model provider and their infrastructure are trustworthy, cryptographic enforcement is less useful. However, this is not a comprehensive threat model. If indeed these models are extremely powerful, factors such as insider threats, auditability, etc., become crucial to consider as part of the threat model. Furthermore, a compliance agency may desire to keep certain aspects of compliance secret from both the users and the model providers. Cryptographic enforcement of policy can have the following benefits:

- **Mitigation of Insider Threats**: If you rely solely on backend checks, you're implicitly trusting everyone who has access to the backend system. Cryptographic measures can mitigate risks from rogue insiders who might want to bypass internal policy checks.

- **Auditability**: Cryptographically signed requests provide a robust and secure method for auditing. Every access request can be stored with its signature, providing an immutable trail of who accessed what and when.

- **Fine-grain Access**: Policy-based signatures can allow fine-grained access control to powerful models. Different parties, such as research labs, could be given access to more unrestricted model capabilities.

- **Tamper Resistance**: Cryptographic techniques ensure that any tampering or alteration of a request/message is detectable. If an adversary tries to modify a request, the cryptographic signature will not match, and the system can easily identify and reject tampered requests.

- **Secrecy**: By employing cryptographic methods, you can ensure that sensitive policy checks, conditions, or other data remain confidential. This can be crucial if you don't want to expose certain aspects of your policy checks to potential attackers. And a compliance agency can certain policy aspects secret from the model providers.

- **End-to-end Security**: While a backend policy check ensures security at the server end, cryptographic signatures can offer end-to-end security, guaranteeing the authenticity of a request from the moment it is made until it reaches its final destination.

Impact

- *If you find a good method and use case, do you have an estimate for how likely it is that AI companies will be interested?*

So, in my opinion, a great use case would be government enforcing some auditability and policy-based access regulations on model providers. This is the next technological step forward after the policy documents that have been coming out of the recent UK AI Safety Summit. However, this is also a great way for companies to adopt transparency and auditability practices. I am happy to reach out and discuss the possibilities for this at the right time with the right people at these companies.

- *Do you know if other high-security industries use cryptography internally? What is the path from a good cryptographic method being discovered/designed to it being used?*

Cryptographic applications are quite ubiquitous today. From digital transactions, credit cards, ATMs, blockchains, TLS for all internet connections, encrypted communication via Signal, etc. As an example of how new cryptographic innovations get adopted to protect critical infrastructure, please see this recent initiative by the US Government's Cybersecurity & Infrastructure Security Agency (CISA) that describes how and why post-quantum cryptography (PQC) is a priority and will become a requirement after the set plan is executed.

Following is a rough plan for executing this project:

1. **Background**: (1-2 weeks)
    - Review current literature on AI/ML safety policies
    - Study existing policy-based signature schemes
    - Familiarize with hardware-isolated/trusted execution environments
2. **Define Policy Scope**: (2-3 weeks)
    - Hold brainstorming sessions with the team to identify potential safety concerns with access to powerful models.
    - Enumerate a list of possible policies that can address these safety concerns.
    - Consult with experts in the field of AI/ML safety to refine the policies.
    - Come up with a list of policies and classify them as achievable, probably achievable, and unlikely to be achievable with current policy-based signatures. Similar classification for formalization of policies as boolean circuits.
3. **Research** (7-9 weeks total)
    a. **Formalization:**(2-3 weeks * 2 team members)
        - How do we formalize the circuit/program that checks if a policy is satisfied?
    b. **Cryptography implementation**: (1-2 weeks* 2 team members)
        - Continue research to come up with policy-based signatures that work for our selected policies and implement resulting policy-based signatures
    c. **Testing**: (2-3 weeks *1 team member)
        - Control access to a powerful open-source model (safety disabled) using these signatures.
        - Test the effectiveness and security of the signature system against various attacks.
    d. **Documentation**: (3-4 weeks * 2 team members)
        - Create comprehensive documentation detailing the policies, cryptographic techniques used, integration methods, and access model design. Write it as an academic paper.

Output

An academic paper that measures the efficacy of this approach and a GitHub repository implementing the policy-based signatures for accessing powerful open-source models that allow replication of our tests.

Risks and downsides

One risk is that potential policies that we can implement could be limited.

Acknowledgements

This proposal is inspired by work of Max Tegmark on Provably safe AGI.

Team

Team size
Looking for 2-3 other team members. 2 engineering-heavy researchers (preferably with ML background, ideally with experience in experimenting with open-source models) + 1 other researcher (strong theory/ML with some eng./formalization/policy background). I will do most of the cryptography research, supervise the engineering work, and a little bit of everything else.

Research Lead
Pratyush Ranjan Tiwari

pratyush@cs.jhu.edu.

I am a 4th year Ph.D. candidate at Johns Hopkins University. My research in the past has primarily been on applied cryptography, complexity theory, and zero-knowledge proofs. I have had a productive last two years as a researcher due to clear research goals, planning, and execution. The next goal is to work towards provable AI safety guarantees.

I will spend 10-15 h/w on this project.

Team Coordinator
I prefer someone else to take this role

Skill requirements

For either of the roles below, no experience in cryptography is required. Interest in AI safety policy and a broad math/theoretical CS background is beneficial.

Research Eng. Roles (2)

- Experience prototyping ideas to code is required
- Background in experimenting with powerful models/LLMs is extremely useful
- Experience reading research papers is essential

Researcher (1)

- Background in ML research would be prioritized: Similarity-driven NLP classification, Semantic Hashing, and general NLP techniques are what we will probably end up using (probably)
- It would help to have a Swiss-army knife mindset toward problem-solving: you can have one area of expertise but interest and inclination towards learning new, cross-disciplinary techniques will go a long way
- Some experience in writing research papers/technical documentation is prioritized