The AI RMF Does not Address Common Needs

I believe that the Risk Management Framework is lacking in actionable recommendations for model governance, and ignores the long history of model risk management that is employed by companies and organizations that deploy models in adversarial settings. It does address large expensive models well, but the majority of models are disposable models trained on shifting datasets. Additionally, there are recommendations that are contrary to model risk from a security standpoint which are not pointed out. A model that resists adversarial attacks is often less secure than a model that was designed without this consideration. Explainability gives attackers information that they can use to attack the model. There are major tradeoffs between security and these two recommendations that organizations need to consider.

Background

The oldest attack against a production AI was likely a word packing attack against Sophos' spam filter in 2003 (Graham-Cumming). This was against the original bag of words spam model, and the attack is classic (Graham). This started the oldest arms race in machine learning security, with attackers innovating to bypass models and defenders innovating new models. It moved to social media when facebook became prominent and is a required component of those platforms. The bypasses are inevitable, there are too many variables for spammers to tweak and the spam systems have to have an extremely low rate of false positives. Blocking users is very bad for businesses that rely on daily active user metrics. The response to these constant attacks is to constantly retrain & redeploy while developing new features for the model. The features need to be hard to manipulate. The content of a message on Facebook is easy to alter while the user behavior is not, therefore, in 2018, Facebook's spam model relied on user behavior and ignored content.

Malware detection used signatures and other brittle mechanisms that required constant maintenance, until the tide of malware got too large for those systems to work (Saxe and Sanders). Now they use machine learning models, and the same old cat and mouse game from spam is still being played. This time, a miss for the defenders could mean a hospital gets ransomed and patients die. To manage this risk the AI is constantly updated. A production malware model requires 99.99% accuracy to be deployable. Below that, the number of false positives becomes too large for security analysts and the model does more harm than good. The low false positives are maintained through the life of the model as benign data doesn't drift as much as malicious data, but the false negatives spike a few months after deployment indicating a bypass has been found and is widespread (Cattell).

Most data is dynamic and constantly changing. That's why a lot of companies originally turn to machine learning. Aside from security models we've detailed above, there are many ML systems that drift:

1. Recommendations systems drift as user preferences change.

2. Financial models drift as markets change with new regulations and products.
3. Business forecast models drift with new packaging, competition, and the weather.

Even large language models drift as new slang is introduced and facts change. BeRT thinks Trump is still the President of the US. Image models used in self driving cars drift as new cars and traffic patterns are introduced. Even developments in the lenses . The only difference is the speed. Malware models require retraining on a monthly basis, but spam models may need a weekly cadence. A model's dataset may move slowly enough . In all likelihood the majority of data scientists today are employed managing drift by retraining and improving existing models.

Issues

Issue 1: The Lifecycle Recommendations Ignore Known Best Practices

The metrics needed to manage the risks shared by models deployed in adversarial settings like spam and malware are time-to-bypass, and response time. A longer time-to-bypass means a lower likelihood of bypass occurring before the next scheduled retraining and redeployment. A short response time means that when a bypass does occur, the model vendor can respond quickly. This extends beyond models in adversarial settings, Microsoft and OpenAI's Codex recommended insecure code, and those companies took months to even respond to the notification that their model was behaving poorly (Anderson). This is unacceptable in the security industry, and should be unacceptable to the machine learning industry.

Other industries may have their own metrics that rely on drift. Recommendation systems may just track the model's performance and retrain when it falls below a certain level. Financial firms can do something similar. The response is usually to retrain and redeploy, either proactively, or with minimal delay. This affects all aspects of the model as metrics often rely on the assumption that the model is trained on data that is distributed identically to the test set. Models used to select which candidate to hire will perform worse when they are outdated. The closest mention of the oldest and most understood model risk management that we have is a on page 30:

> **Datasets used to train AI systems may become detached from their original and intended context, or may become stale or outdated relative to deployment context.**

For many orgs this isn't true, the dataset is continually updated because they know it drifts or they just want more data. There are several places in the playbook where the report mentions monitoring the AI system for potential issues, but few suggestions about addressing the problem. Drift and monitoring is almost non-existent in the main document.

Issue 2: Adversarial Robustness Has Significant Drawbacks in Current Models

Adversarial robustness as defined in the academic literature lowers accuracy (Carlini et al.) (Madry et al.). This report needs to make this clear to the executives it targets. In malware, employing techniques that increase model resilience to this not only hamstrings the model, it

can lower time-to-bypass for models in adversarial settings. Nearly all security models and data scientists have rejected this definition of robustness as they have found it counter productive. In medical situations, I would not like to have a mis-diagnosis due to a “secure” model. This report may say it is just voluntary recommendations, but it will be seen as best practices. Medical device manufacturers are already employing adversarial robustness to comply with this report and the EU law that is under development. There is a tradeoff and the model’s manufacturer will choose to comply with this report rather than do what’s best for patients if this trade off is not made clear.

Recommendations

The primary recommendation we have is to include mention of the known and understood risk management companies are already doing.

- The GOVERN section should include a recommendation that there be a guaranteed time to response built into the model plan when a stakeholder brings up an issue.
    - This would fit in well as GOVERN 5.3.
    - The time to response is common in security critical applications.
- The GOVERN section should include a strong suggestion for an external audit of the model and the monitoring framework.
    - Internal audits will include the bias of the company and team developing it. External auditors have fresh eyes and are not as influenced by the business decisions.
- MEASURE 2 section should not list everything but call back to the risks identified in the MAP section.
    - Listing all of this in this section indicates that these are universal risks. Companies will use this list as a list of risks instead of identifying their own.
    - A malware model does not suffer from privacy or bias issues, and should not be evaluated with this in mind. Also, its explainability is largely irrelevant.
- The MEASURE 3 section should mention tracking performance metrics over time.
    - Companies will use one time evaluations for 3.1 and 3.2 if this isn’t called out.
- The MANAGE 2.2 section should include a suggestion to create a scheduled lifecycle.
    - A scheduled lifecycle is uncommon of in software development, but essential for ML

Works Cited

Anderson, Ross. “The Dynamics of Industry-wide Disclosure.” *Light Blue Touchpaper*, 5 August 2022, https://www.lightbluetouchpaper.org/2022/08/05/the-dynamics-of-industry-wide-disclosure/. Accessed 29 September 2022.

Carlini, Nicholas, et al. *Certified! Adversarial Robustness for Free*. 2022. *arXiv*, https://arxiv.org/abs/2206.10550.

Cattell, Sven. “Online Dataset Drift in O(log N).” *RobustML Workshop@ICLR 2021*, vol. 1, no. 1, 2021, pp. 1-7. *RobustML Workshop@ICLR 2021*, https://sites.google.com/connect.hku.hk/robustml-2021/accepted-papers/paper-091.

Graham, Paul. “A Plan for Spam.” *Paul Graham*, 2002, http://www.paulgraham.com/spam.html. Accessed 29 September 2022.

Graham-Cumming, Paul. *How to beat an adaptive spam filter.* MIT Spam Conference, 2004.

Madry, Aleksander, et al. *Towards Deep Learning Models Resistant to Adversarial Attacks*. *arXiv*, https://arxiv.org/abs/1706.06083.

Saxe, Joshua, and Hillary Sanders. *Malware Data Science: Attack Detection and Attribution*. No Starch Press, 2018.