THREAT MODELING CONNECT
GLOBAL MEETUP

Threat Modeling in the Age of AI

November 14, 2024
11am ET

THREAT MODELING CONNECT | POWERED BY IRIUSRISK

Roadmap

- **This talk is about:** *understanding the **centrality of data** to AI/ML system security, and applying that knowledge, through the lenses of 3 methodologies.*

- **Tool-agnostic:** We are focusing on understanding how **the role of data** changes threat vectors for AI vs traditional systems

- **Goal:** To become prepared to **think critically about data** & model threats in any AI/ML-driven environment

Threat Modeling Manifesto

“Threat modeling is analyzing representations of a system to highlight concerns about security and privacy characteristics.”

“At the highest levels, when we threat model, we ask four key questions:

- ***What are we working on?***
- ***What can go wrong?***
- ***What are we going to do about it?***
- ***Did we do a good enough job?”***

Source: Threat Modeling Manifesto

Data: From New Oil to New Attack Vector

2010s: Data is the **“new oil”**

- Tremendous potential value
- AI viewed as low-to-no security risk, high reward

2020s: Data is the new **attack vector**

- Tremendous potential threats
- As more systems include AI, risk spreads

AI Is About Scale

- Requires operationalization

- MLSecOps as a related discipline to DevSecOps

- Operationalization gives many (but not all!) of the answers to:

  > What are we working on

  > What can go wrong

  > What we are going to do about it

  > Whether we did a good enough job

Three Approaches to Understanding AI-Specific Threats

- NIST AI 100-2e2023
  - *CIA model, applied to AI*
- OWASP AI Exchange
  - *Dev/Deployment phases, MLSecOps*
- Boolean path threat model + OODA Loop
  - *Game theoretic, boolean, OODA*

CIA Model In The New AI Era

- Traditional CIA: ***Confidentiality, Integrity, Availability***
- What does this mean for AI?
- NIST AI 100-2e2023 Taxonomy refers to CIA model ++

CIA Model In The New AI Era

Availability Breakdown

"*An AVAILABILITY ATTACK is an indiscriminate attack against ML in which the attacker attempts to break down the performance of the model* ***at deployment time****.*"

- Data poisoning: when the attacker controls a fraction of the training set
- Model poisoning: when the attacker controls the model parameters
- Energy-latency attacks via query access
  - Energy-latency attacks "*exploit the performance dependency on hardware and model optimizations to negate the effects of hardware optimizations, increase computation latency, increase hardware temperature and massively increase the amount of energy consumed.*"

Source: NIST AI 100-2e2023

CIA Model In The New AI Era

Integrity Violations

“An INTEGRITY ATTACK targets the integrity of an ML model’s output, resulting in incorrect predictions performed by an ML model.”

- **Evasion attack** ***at deployment time***
  - Modifying testing samples to create adversarial examples which are misclassified by the model to a different class, while remaining imperceptible to humans

- **Poisoning attack** ***at training time***
  - Targeted poisoning: to violate the integrity of a few targeted samples; assumes that the attacker has training data control to insert poisoned samples
  - Backdoor poisoning: requires the generation of a Backdoor Pattern, which is added to both the poisoned samples and the testing samples to cause misclassification. Backdoor attacks are the only attacks in the literature that require both training and testing data control.
  - Model poisoning: could result in either targeted or backdoor attacks; attacker modifies model parameters to cause an integrity violation

Source: NIST AI 100-2e2023

CIA Model In The New AI Era

Privacy Compromise at Deployment Time

"Attackers might be interested in learning information about the training data (resulting in DATA PRIVACY attacks) or about the ML model (resulting in MODEL PRIVACY attacks)."

Attacker objectives: compromising the privacy of training data, such as

- **Data Reconstruction**: *inferring content or features of training data*
- **Membership-Inference Attacks**: *inferring the presence of data in the training set*
- **Data Extraction:** *ability to extract training data from generative models*
- **Property Inference**: *inferring properties about the training data distribution*
- **Model Extraction:** *a model privacy attack in which attackers aim to extract information about the model*

Source: NIST AI 100-2e2023

An AI-Tailored Approach: Understanding Threats in Their Lifecycle Phases

Many AI-specific vulnerabilities occur during ***key phases*** *in the* ***AI development lifecycle:***

> *Training time*

- Poisoning

> *Deployment time*

- Evasion & Privacy
- Model theft

AIML Lifecycle Phases: Data Roles & Risks

> Training Time

Risk: **Poisoning** - *Manipulating data that the model uses to learn, in order to affect the algorithm's behavior*

- Example: "*an attacker breaks into a training set database to add images of houses and labels them as 'fighter plane', to mislead the camera system of an autonomous missile. The missile is then manipulated to attack houses.*"
- Mitigations: Protect data, increase or alter data so poisoning is less effective, build models resilient to poisoned data (possibly through adversarial training), and monitor data for poisoning attacks

Source: Owasp AI Exchange, 3.1.1. Data poisoning

AIML Lifecycle Phases: Data Roles & Risks

> Deployment Time

Risk: **Evasion** - *creating input which maliciously misleads a model into performing its task incorrectly*

- Example: "*slightly changing traffic signs so that self-driving cars may be fooled.*"
  - **Note**: these changes are often mathematically optimized so as to be imperceptible to humans
- Mitigations: Monitor data for unusual or known malicious inputs, distort inputs in order to make adversarial changes ineffective, design systems which are more robust to adversarial examples (possibly through adversarial training)

Source: Owasp AI Exchange, 2.1. Evasion

AIML Lifecycle Phases: Data Roles & Risks

> *Deployment Time*

Risk: **Sensitive data disclosure**

- Example: "*The output of the model may contain sensitive data from the training set, for example a large language model (GenAI) generating output including personal data that was part of its training set. Furthermore, GenAI can output other types of sensitive data, such as copyrighted text or images...*"
  - **Note:** Data from the training set can be disclosed by either malicious activity or normal use
- Mitigations: Filter sensitive output, obscure confidence indications, keep models small to prevent overfitting/memorization

Source: Owasp AI Exchange, 2.3. Sensitive data disclosure through use

Game Theoretic Modeling for AI Security

Boolean path threat model + OODA Loop: Attacker’s ***costs*** & ***benefits*** required for success

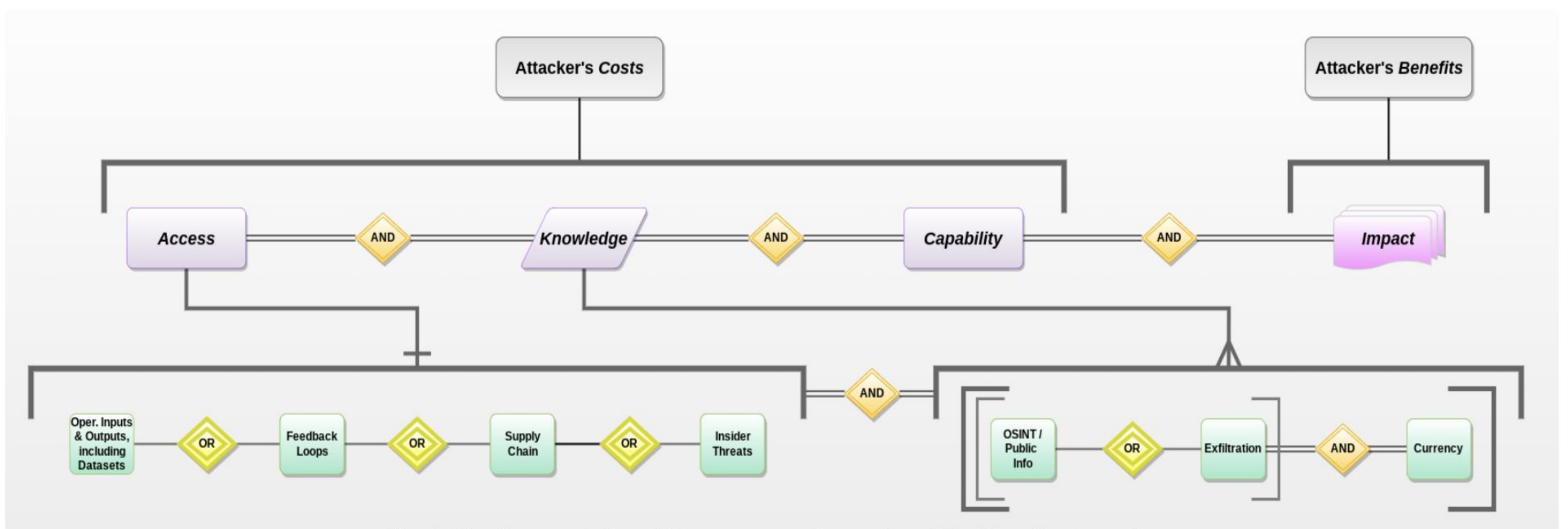


Fig 1: Boolean relationships among elements of Red’s attack vectors

Source: Securing AIML Systems in the Age of Information Warfare

Game Theoretic Modeling for AI Security

*At the highest level, Red [the adversary] must do **four things:***

1. ***Access** the system in question*
2. ***Know** enough to conduct an attack*
3. *Have **resources & capability** to carry out*
4. *Create **negative mission repercussions** for defenders*

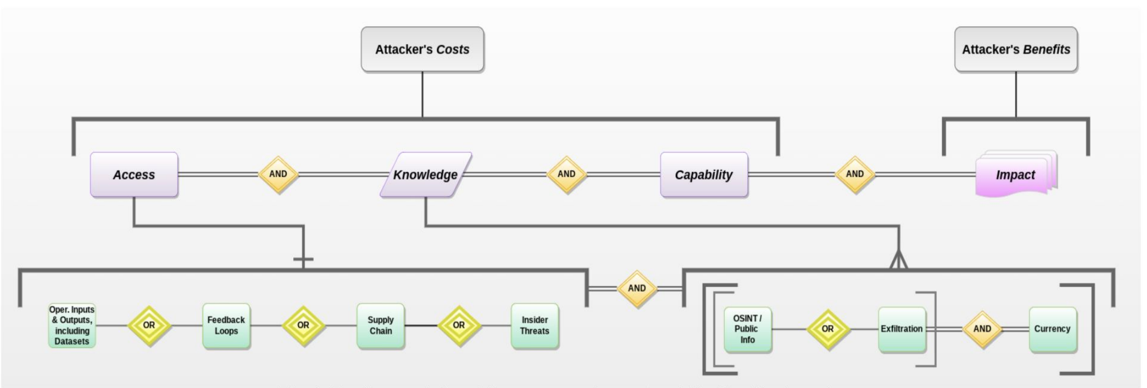


Fig 1: Boolean relationships among elements of Red's attack vectors

Source: *Securing AIML Systems in the Age of Information Warfare*

Game Theoretic Modeling for AI Security

In AIML systems, an attacker might:

- Gain **knowledge** of model behavior or specs via probing the system
- Gain **access** to training data via user-created intake, feedback loops, supply chain, or IT security breach

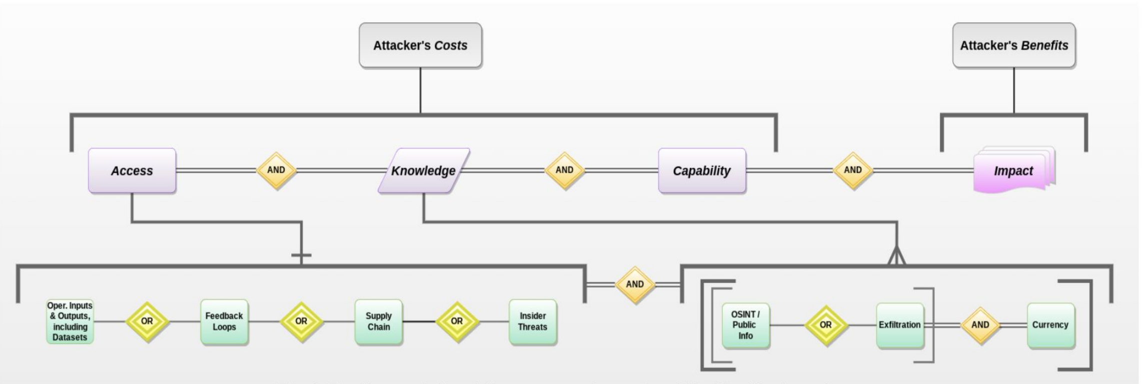


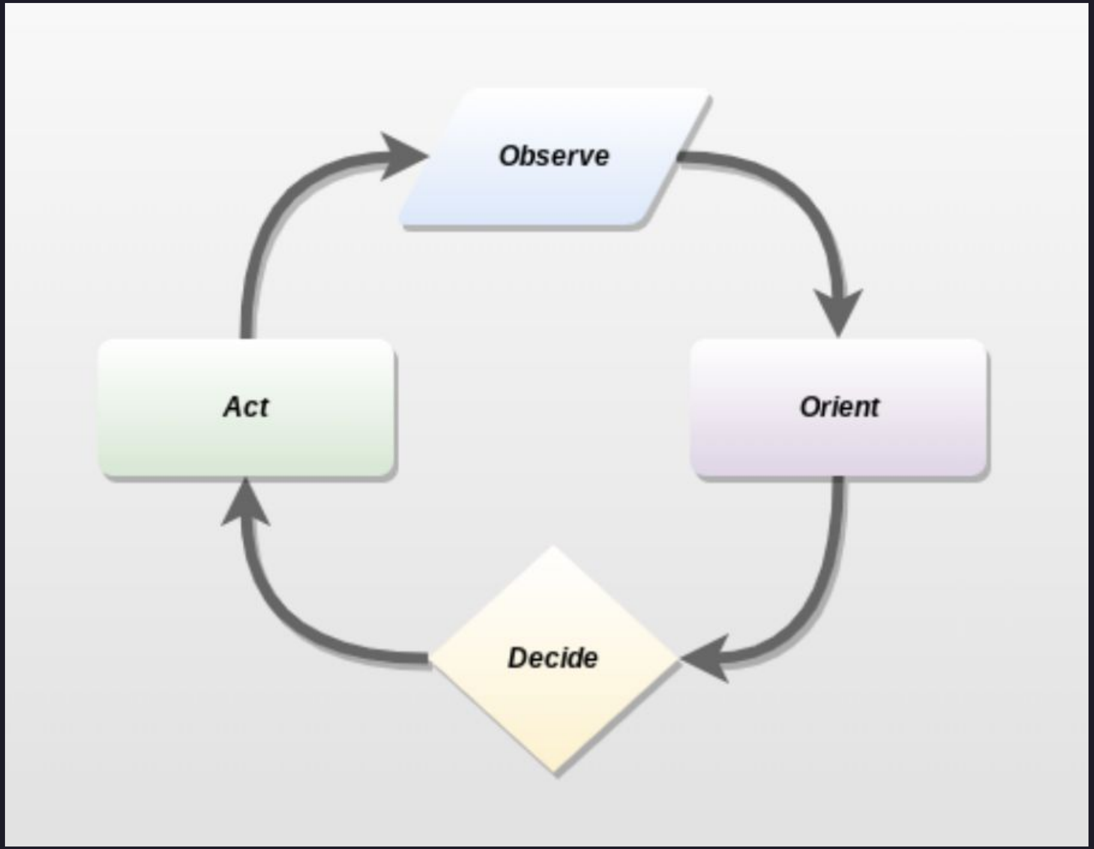
Fig 1: Boolean relationships among elements of Red's attack vectors

Source: Securing AIML Systems in the Age of Information Warfare

Game Theoretic Modeling for AI Security

The **OODA Loop** is:

- An information processing & decision framework used in tactical ops
- Game theoretic modeling for Information Warfare
- Modeling how adversaries & defenders must gather information, filter, make decisions, and act to create impact

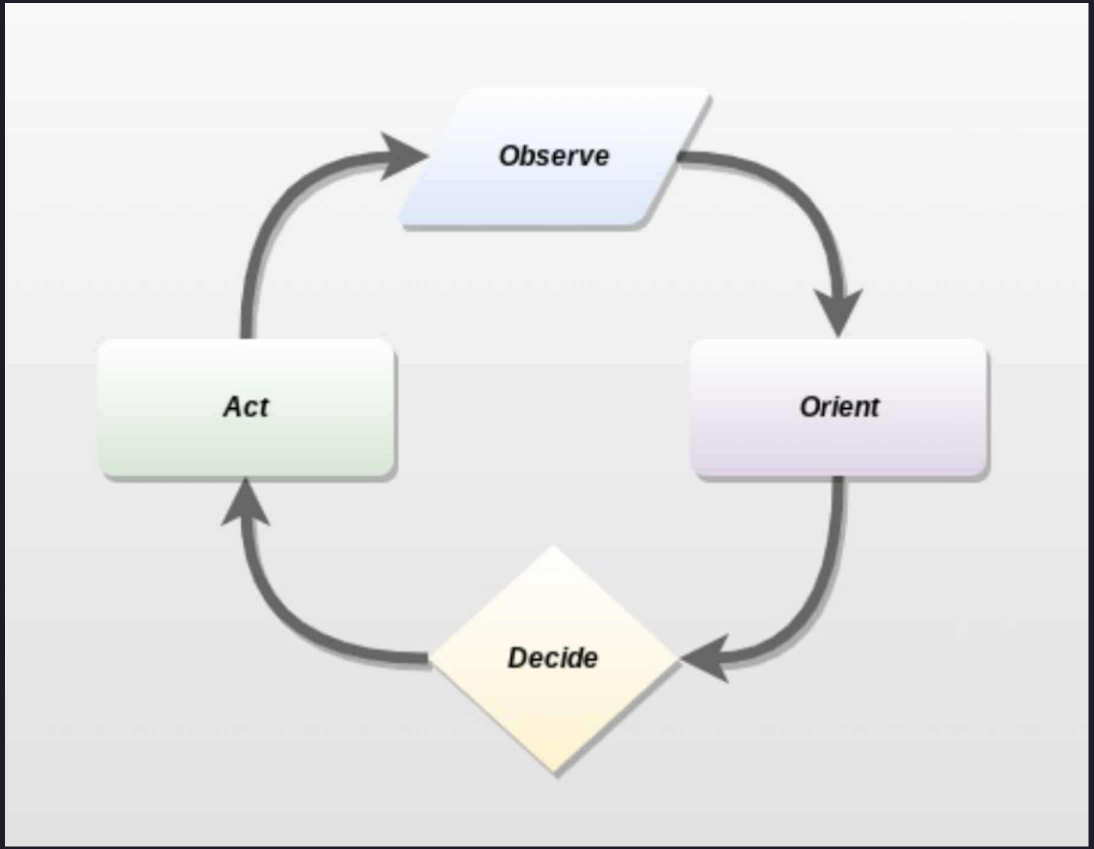


Source: [Securing AIML Systems in the Age of Information Warfare]

Game Theoretic Modeling for AI Security

In AIML systems, the OODA Loop changes:

- AI can help Red to gather, filter, make decisions & execute faster
- Access to Blue's data & feedback loops tighten Red's OODA loop
- Blue may have increased difficulty remaining inside Red's loop



Source: Securing AIML Systems in the Age of Information Warfare

Mapping & Securing the Attack Surface: Operationalization & Data Intelligence

Three steps to understanding your AIML system attack surfaces:

1. Know your data ***flows***
2. Know your data ***provenance***
3. Know your data ***governance***

Three questions to ask:

1. Is it **secure**?
2. Can we **operationalize**?
3. Does it **scale**?

Resources

- OWASP AI Exchange
- NIST AI 100-2e2023
- Threat Modeling Manifesto
- Threat Modeling Capabilities
- Securing AIML Systems in the Age of Information Warfare
- anglesofattack.io

Take home message from Susanna

> *In AIML systems, data is key.*

Know where your data comes from, know how it flows, and use operationalized controls in the appropriate development phases to protect, mitigate potential events, and return to trusted states.