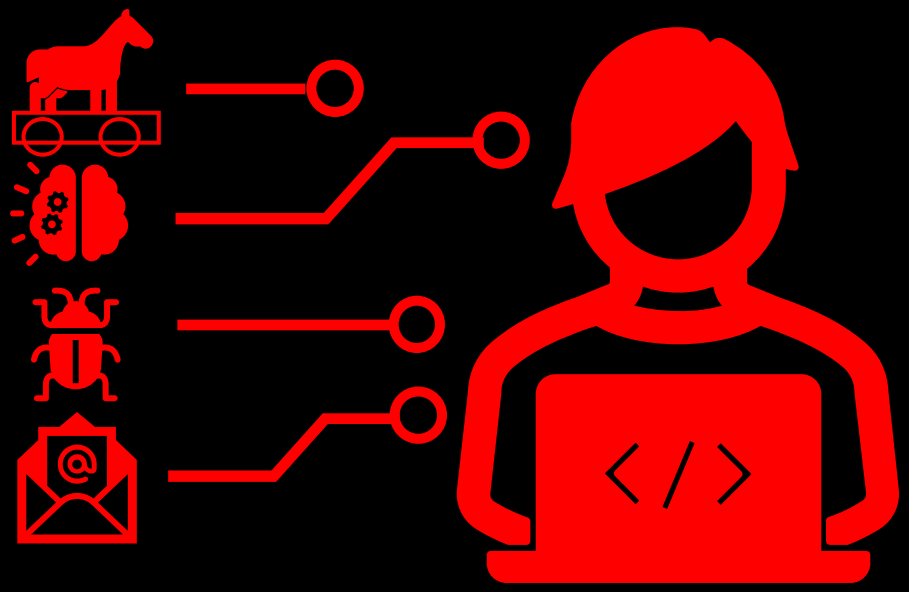
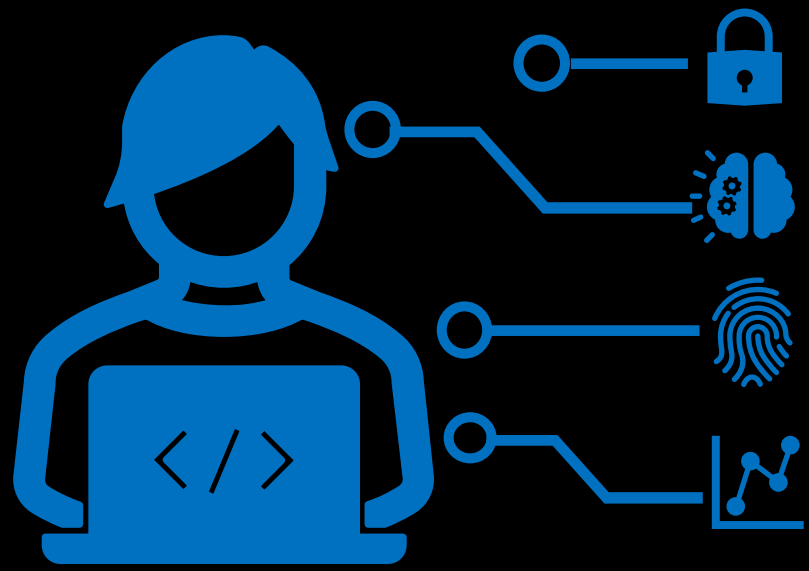
DeepRed: A Deep Learning–Powered Command and Control Framework for Multi-Stage Red Teaming Against ML-based Network Intrusion Detection Systems

Mehrdad Hajizadeh, Pegah Golchin, Ehsan Nowroozi, Maria Rigaki,

Veronica Valeros, Sebastian Garcia, Mauro Conti, Thomas Bauschert

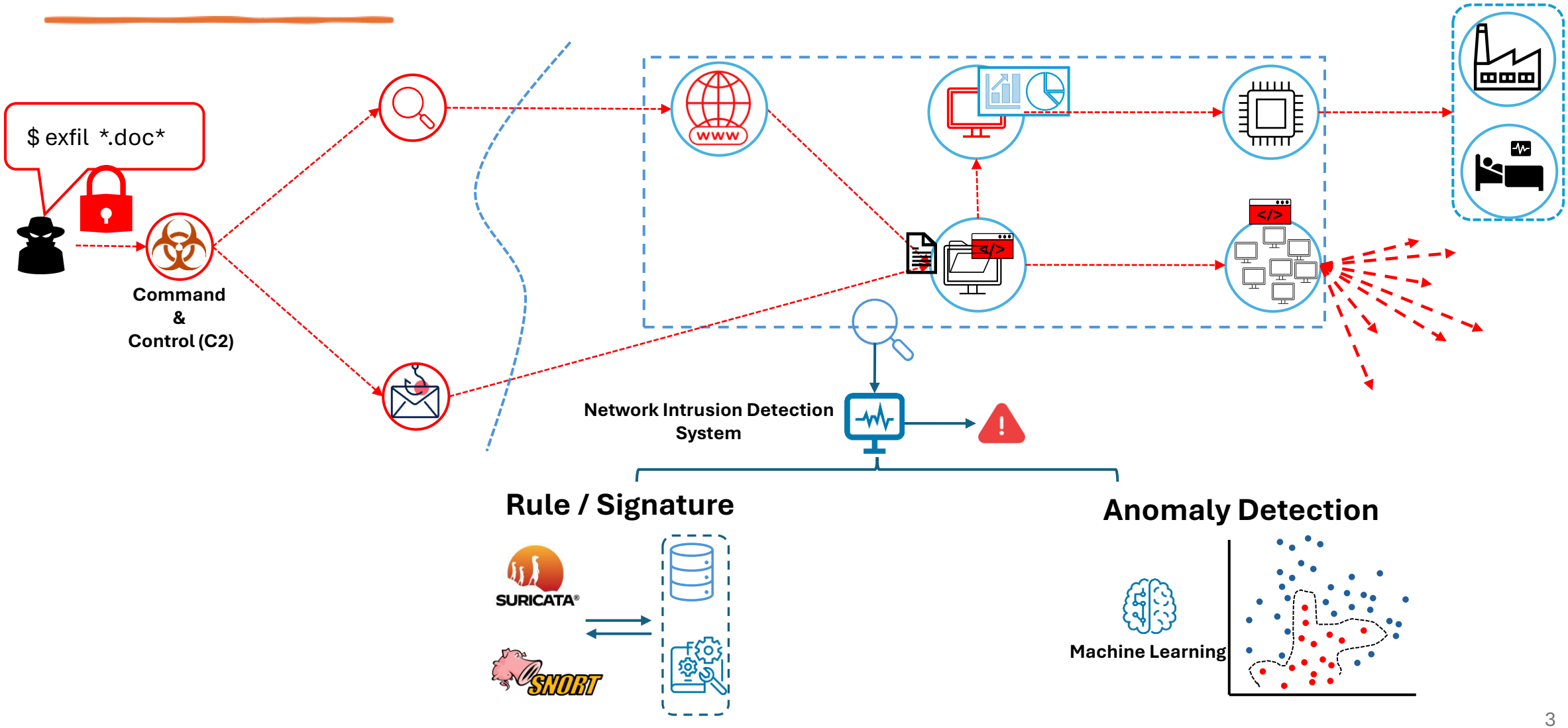
“To Know Your Enemy, You Must Become Your Enemy”



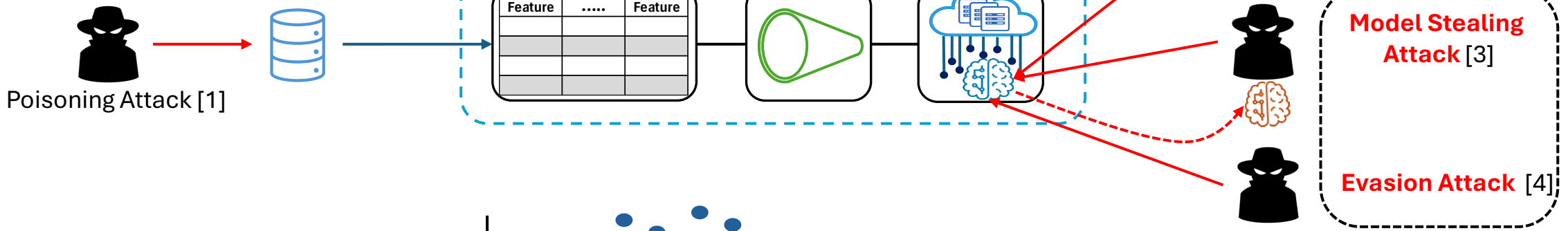


[1] The art of war, Sun Tzu

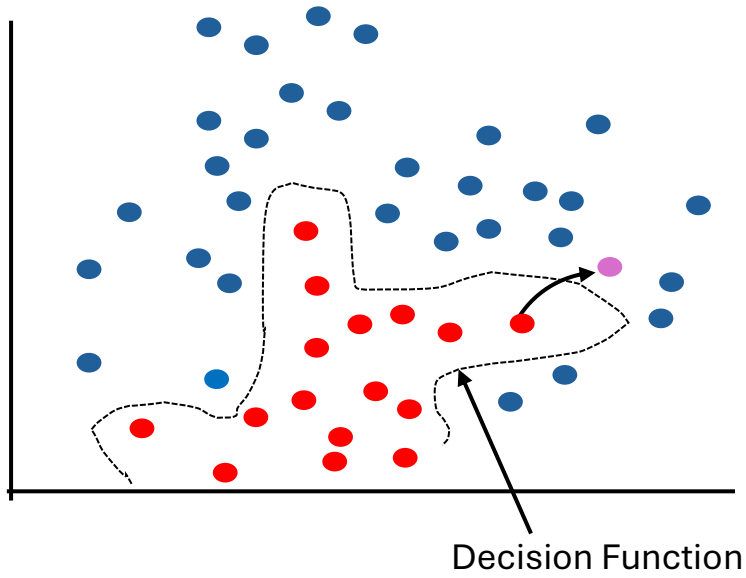
“Think Like An Adversary”

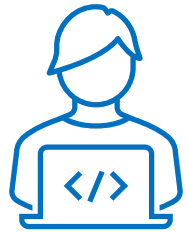


Adversarial Machine Learning: CIA



$$\max_{\delta \in \Delta, y=0} Loss\ (F_\theta(x+\delta), y)$$

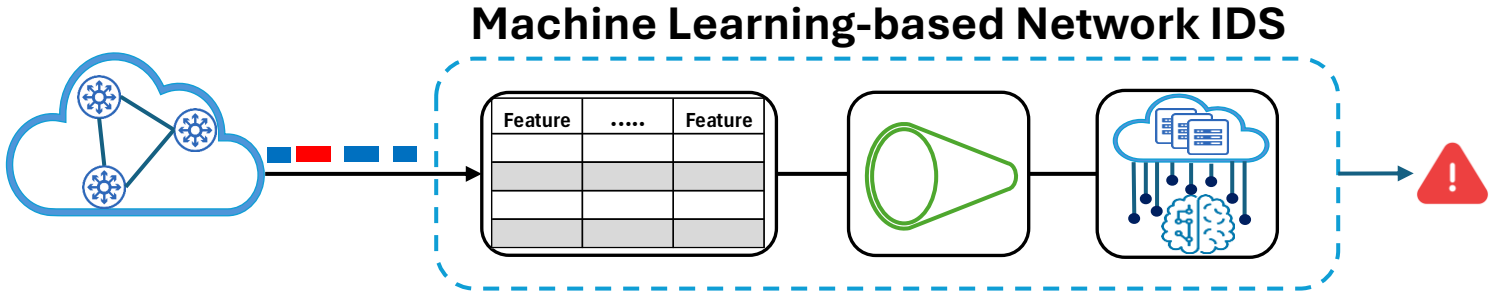




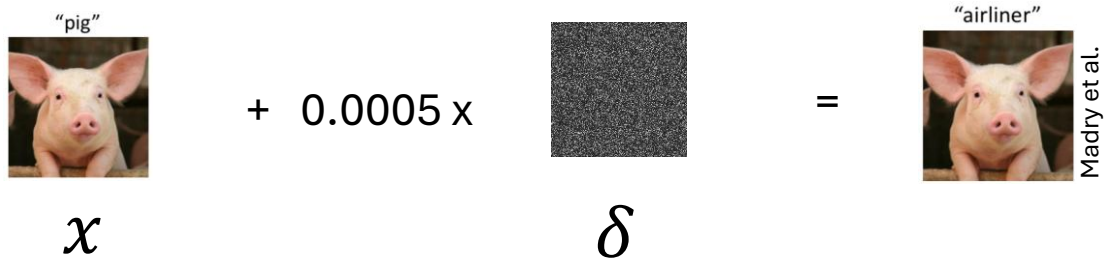
$$\min Loss\ (F_\theta(x), y)$$
$$y \in \{0,1\}$$

[1] Kravchik et al, Practical evaluation of poisoning attacks on online anomaly detectors in industrial control systems
[2] Shumailov´et al,"Sponge examples: Energy-latency attacks on neural networks,"
[3] Carlini et al, Stealing Part of a Production Language Model
[4] Goodfellow et al, Explaining and harnessing adversarial examples

Network Traffic Constraints for Finding δ



- ❑ Domain Constraint (TCP/IP)
- ❑ Attack Functionality Preservation
- ❑ Threat Model

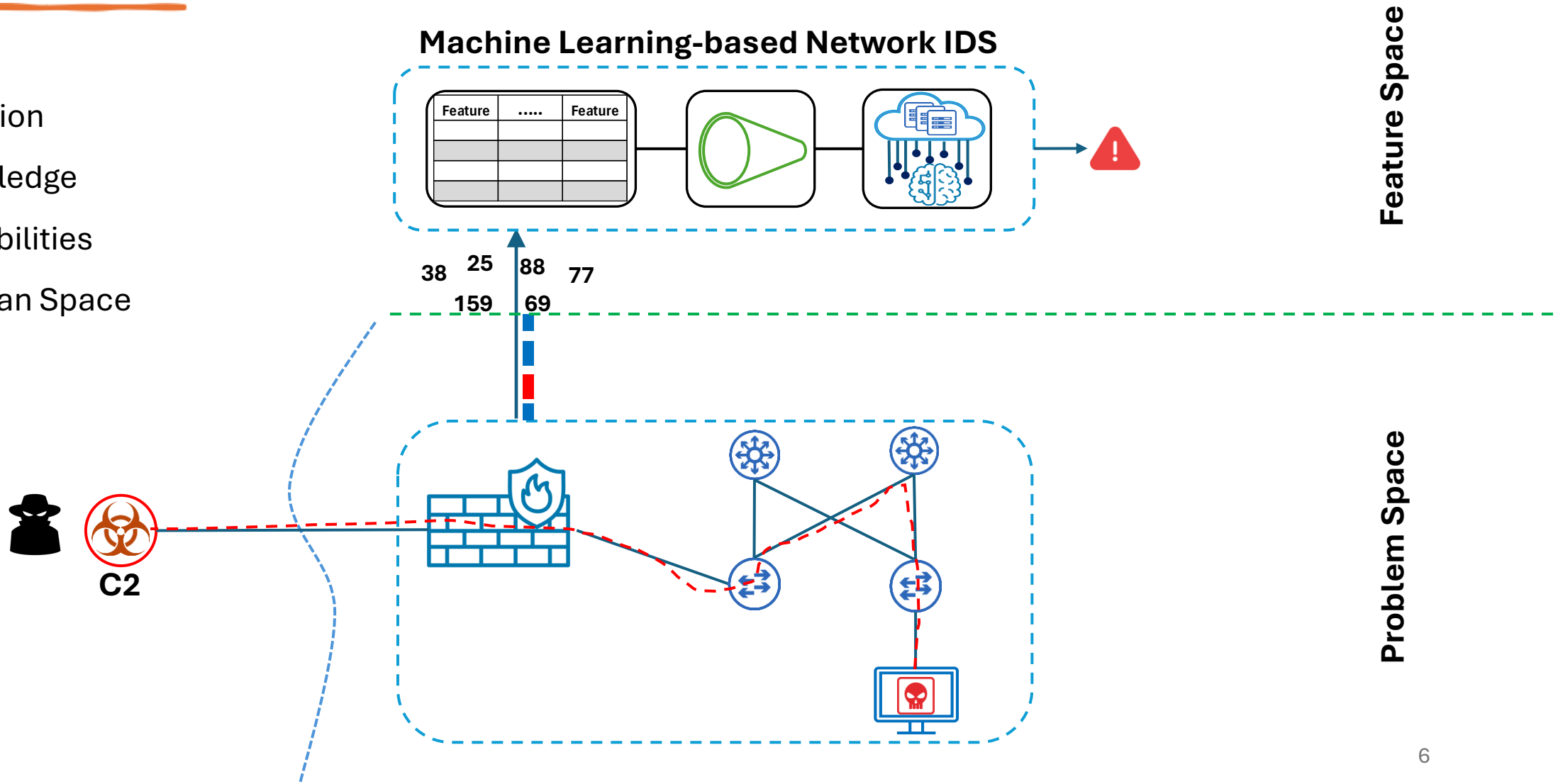


x δ

Threat Modelling

Attacker's

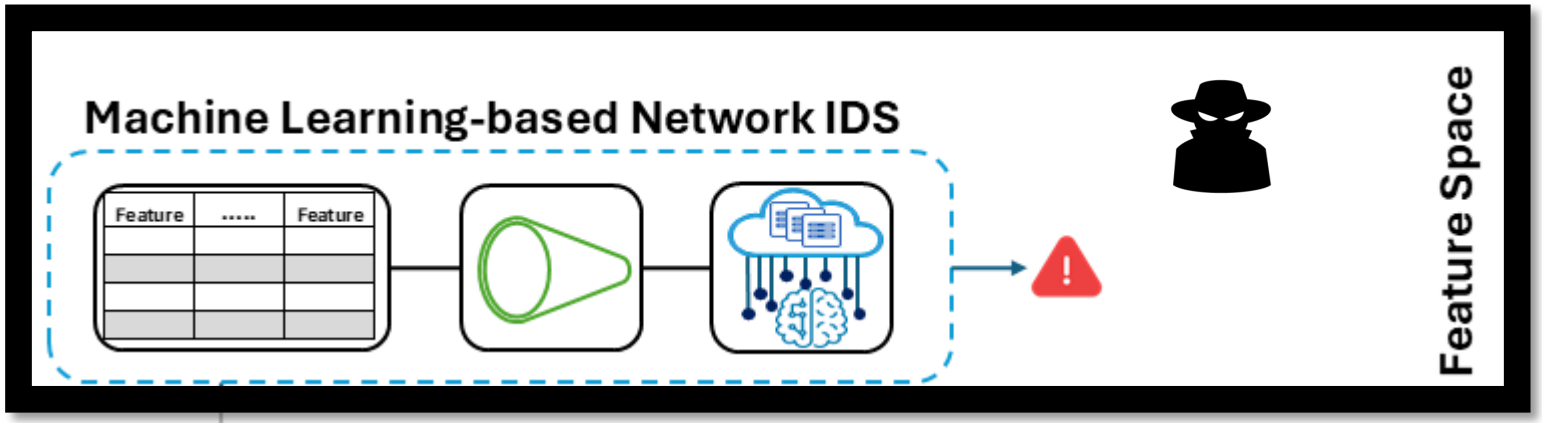
- ❑ Location
- ❑ Knowledge
- ❑ Capabilities
- ❑ Domian Space
- ❑ Etc.

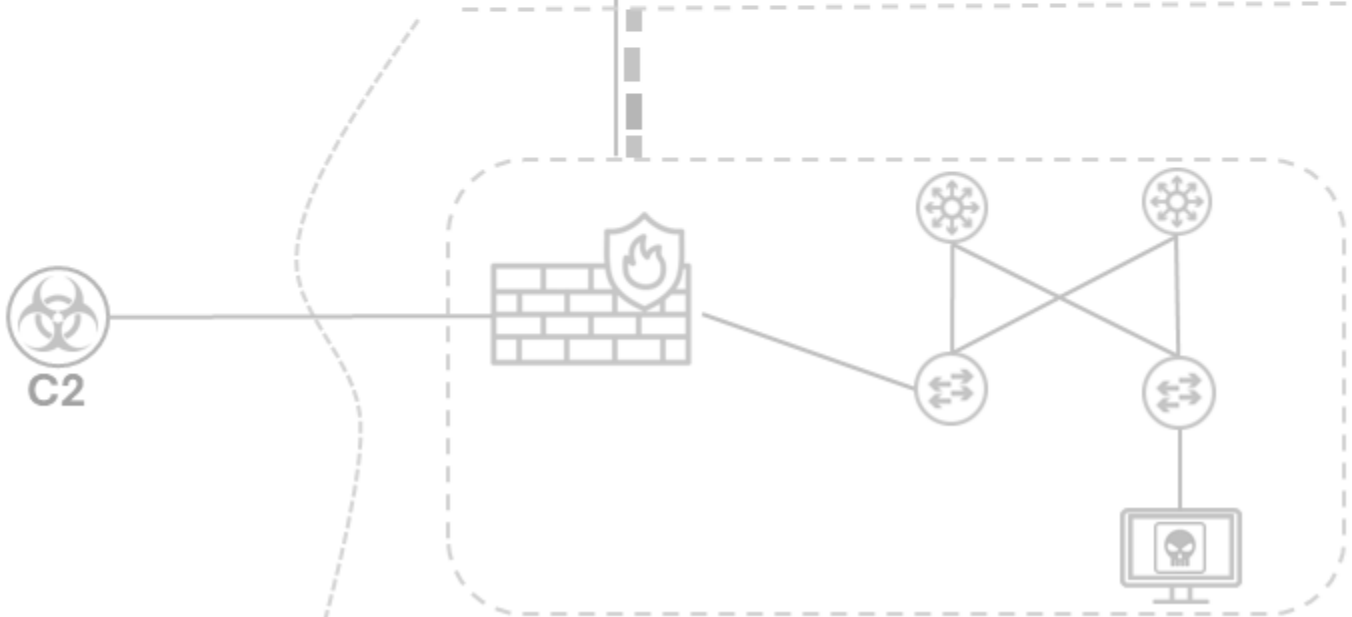


Common Gaps in Existing Threat Modelling

Hypothetical Threat!!

- ❑ **Access Level**
- ❑ Misclassification
- ❑ Capabilities
- ❑ Domian Space
- ❑ Attack Functionality Preservation



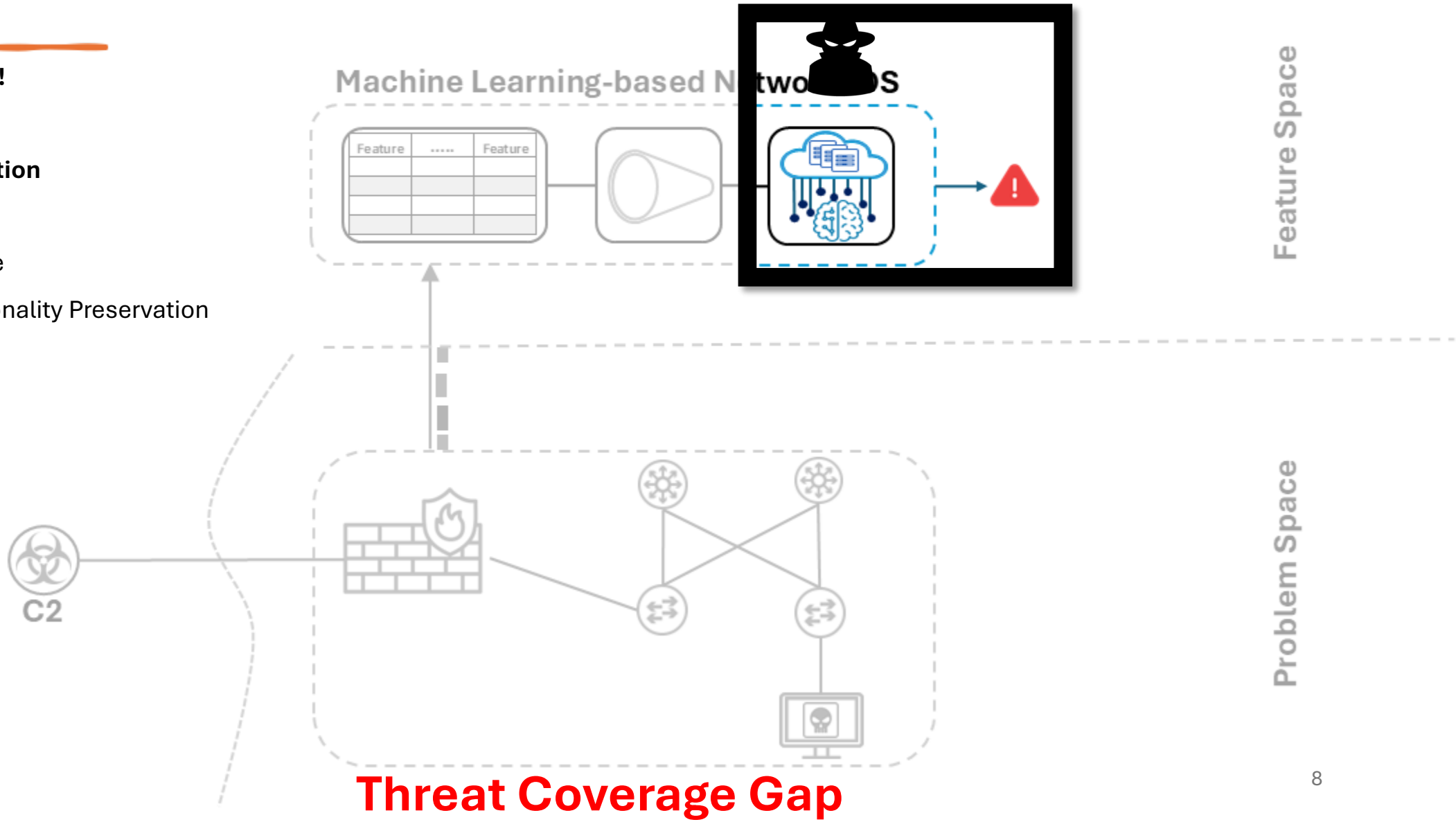


Threat Coverage Gap

Common Gaps in Existing Threat Modelling

Hypothetical Threat!!

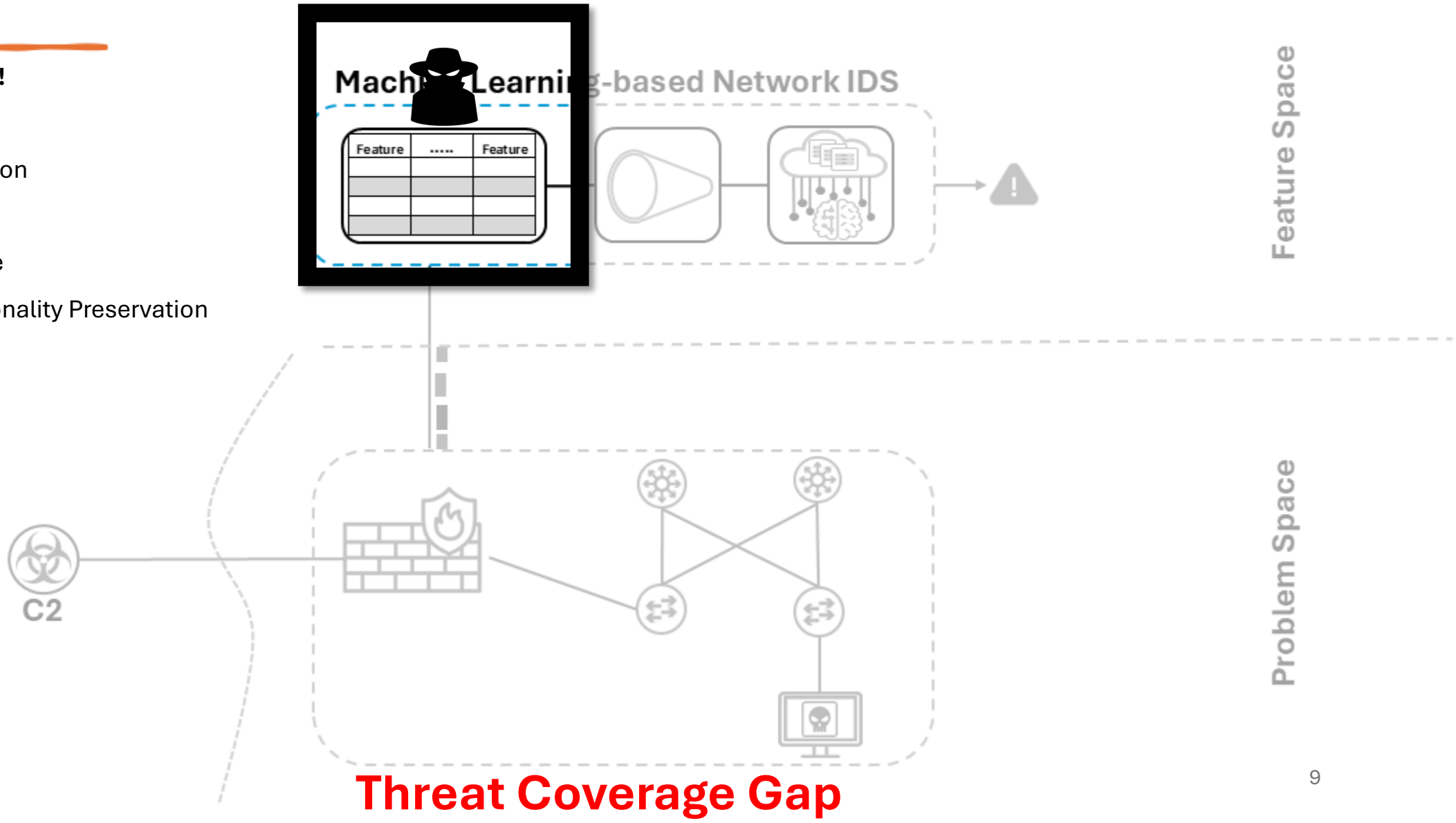
- ❑ Access Level
- ❑ **Misclassification**
- ❑ Capabilities
- ❑ Domian Space
- ❑ Attack Functionality Preservation



Common Gaps in Existing Threat Modelling

Hypothetical Threat!!

- ❑ Access Level
- ❑ Misclassification
- ❑ **Capabilities**
- ❑ Domian Space
- ❑ Attack Functionality Preservation

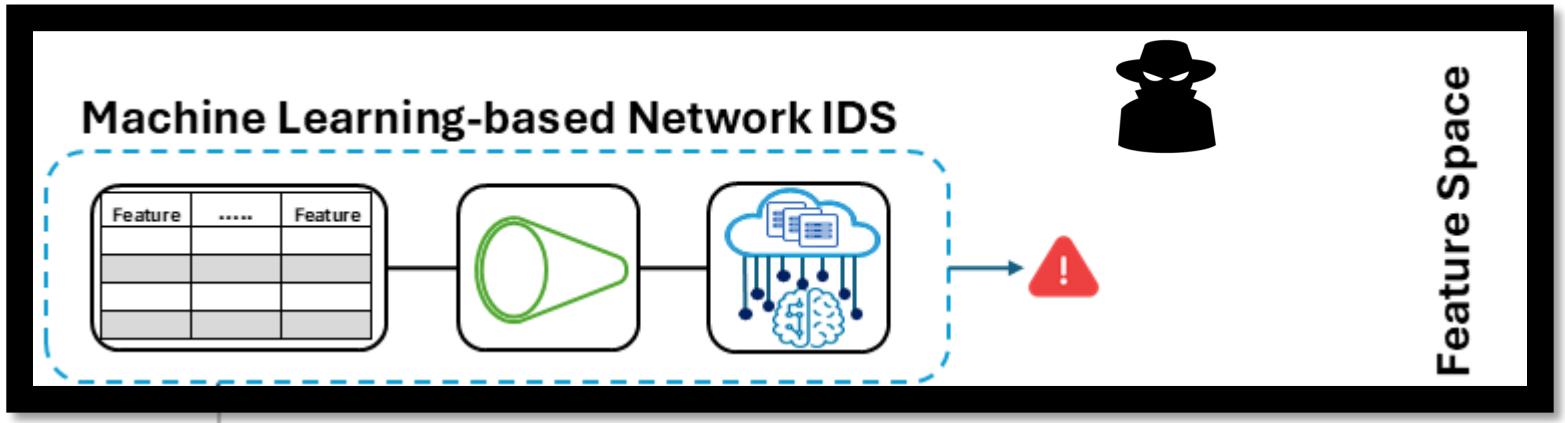


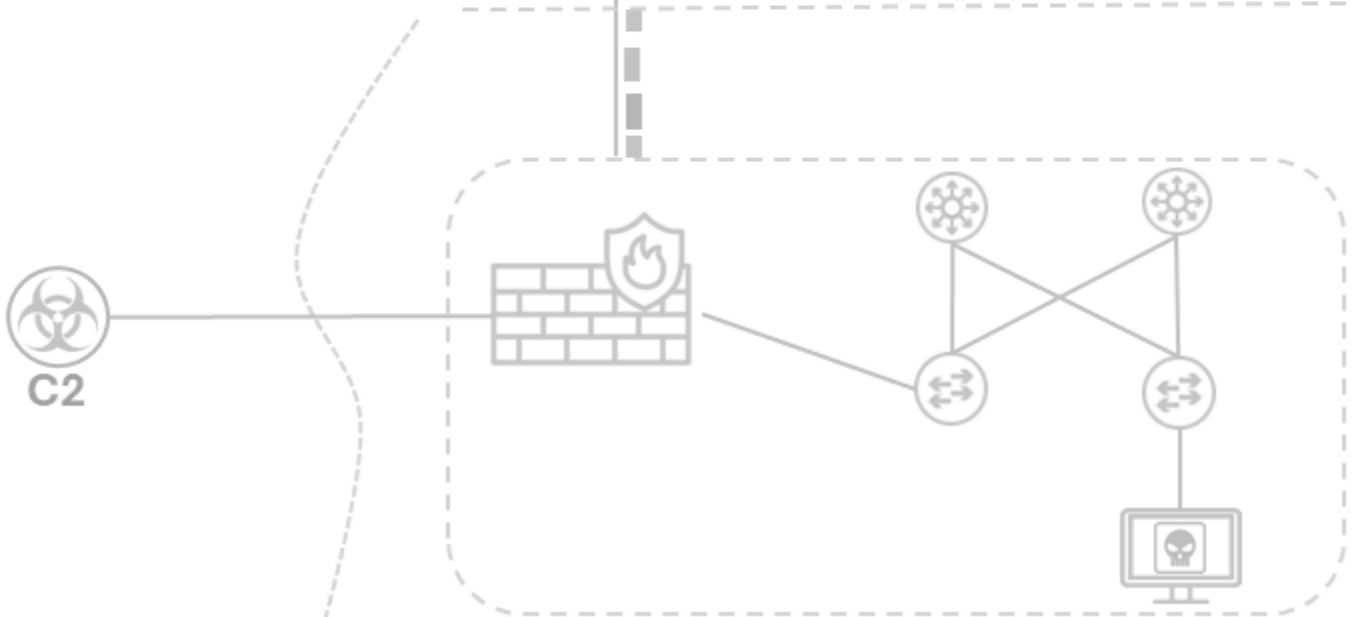
Threat Coverage Gap

Common Gaps in Existing Threat Modelling

Hypothetical Threat!!

- ❑ Access Level
- ❑ Misclassification
- ❑ Capabilities
- ❑ **Domian Space**
- ❑ Attack Functionality Preservation



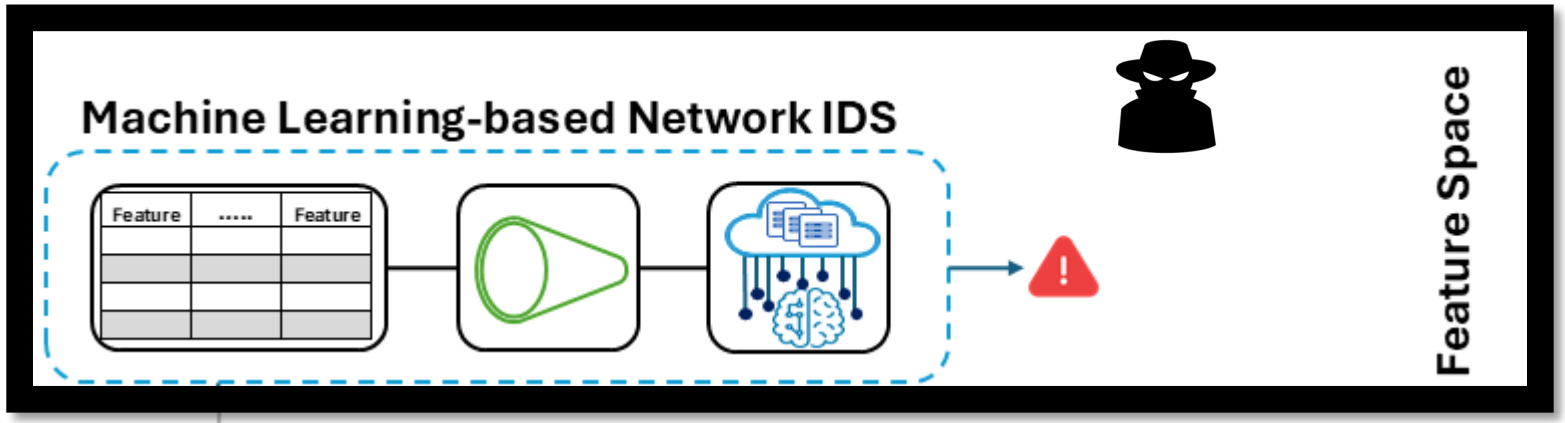


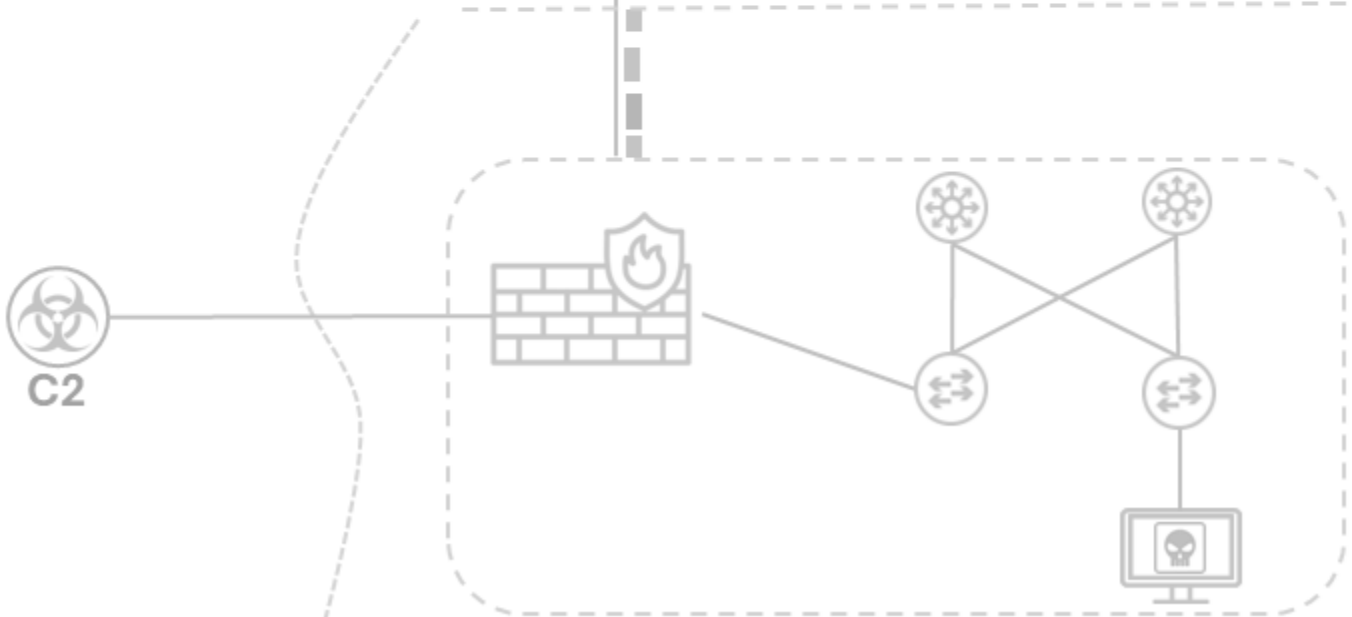
Threat Coverage Gap

Common Gaps in Existing Threat Modelling

Hypothetical Threat!!

- ❑ Access Level
- ❑ Misclassification
- ❑ Capabilities
- ❑ Domian Space
- ❑ **Attack Functionality Preservation**

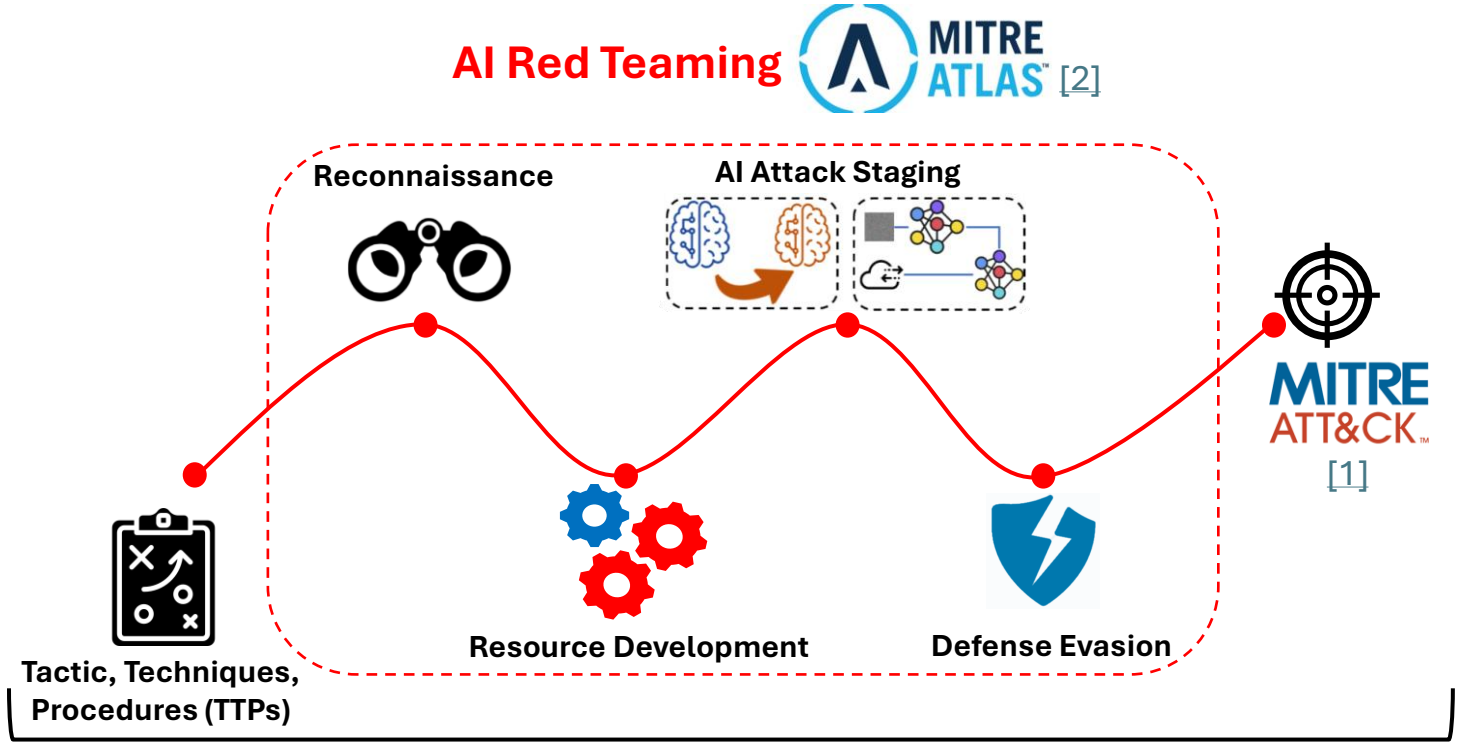




Threat Coverage Gap

Proposed Attack Methodology

Multi-Stage Red Team Operation

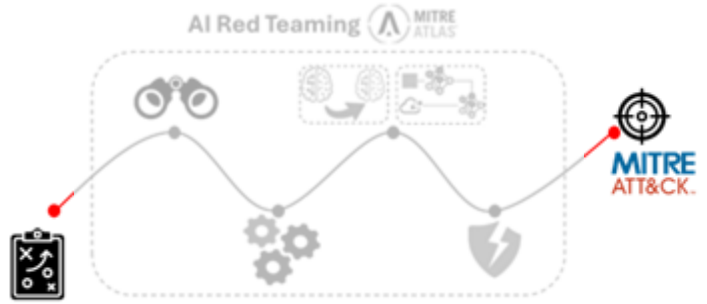


[1] https://attack.mitre.org/
[2] https://atlas.mitre.org/

Adversary‘s Objectives

MITRE ATT&CK Techniques, Tactic and Procedure (TTPs)

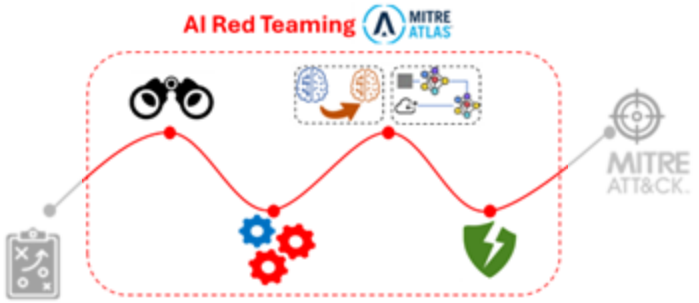
- ❑ Command & Control (T1071)
  - ❑ Remote Command Execution (T1059)
  - ❑ Remote Host/System Discovery (T1082, T1018, T1049, T1083, T1057)
  - ❑ Data Exfiltration (T1041, T1567)

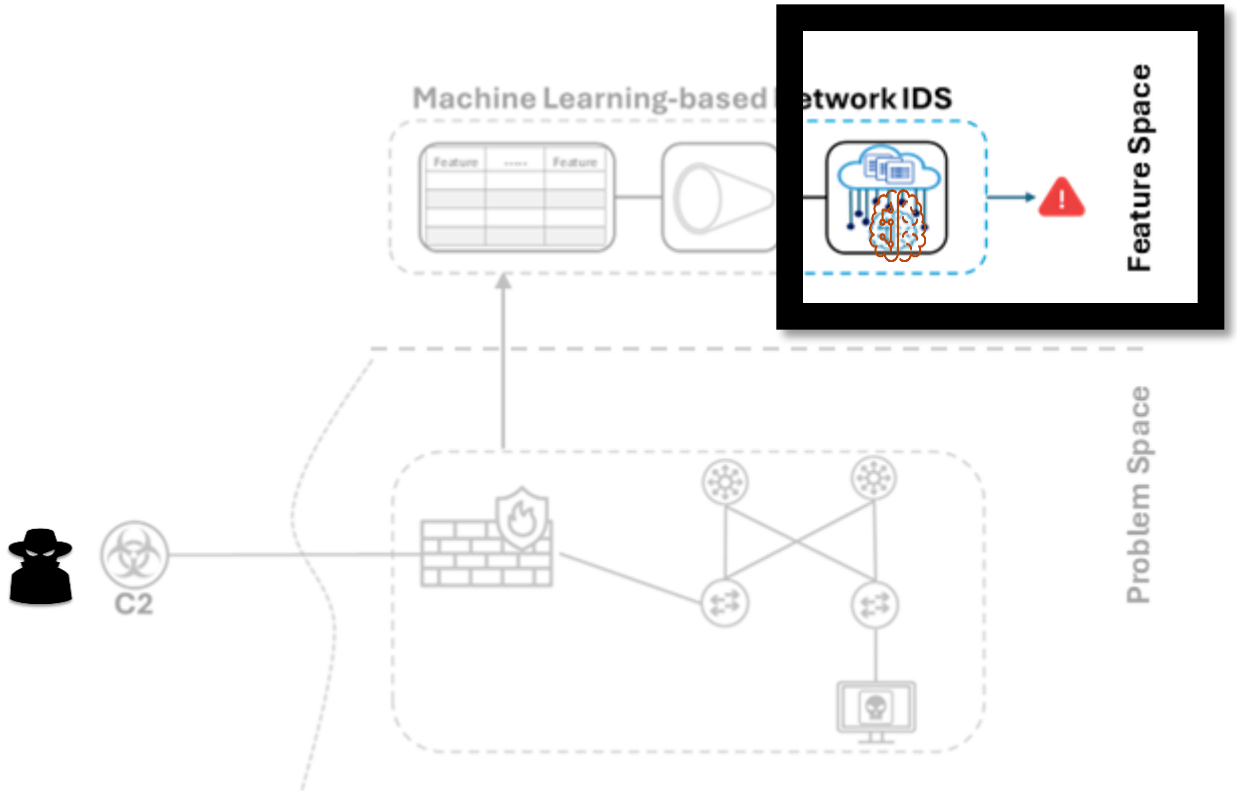


AI Red Teaming Against ML-NIDS

Overall attack strategy

- ❑ **Perform model stealing to build a substitute model**
- ❑ Generate adversarial examples in feature space to identify δ
- ❑ Apply successful perturbations in problem space through real C2 operations

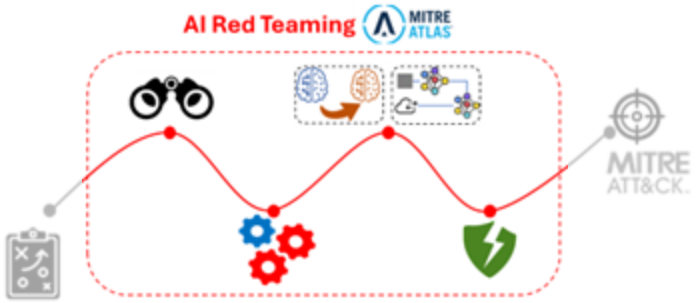


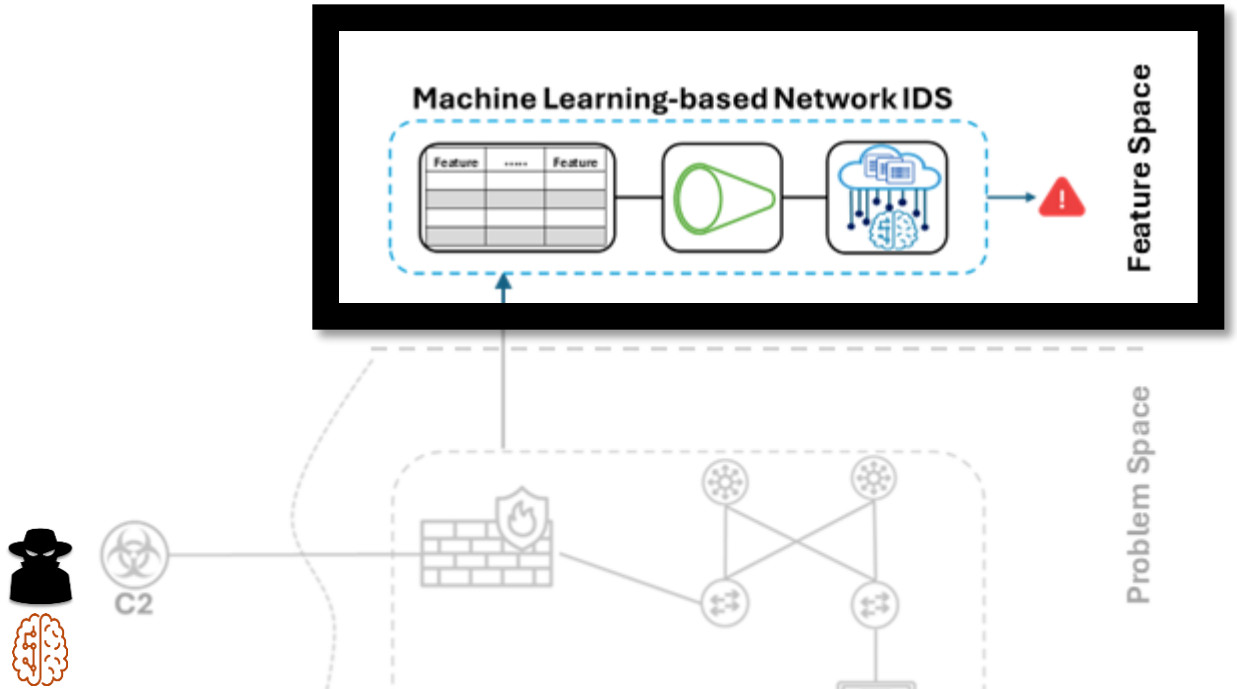


AI Red Teaming Against ML-NIDS

Overall attack strategy

- ❑ Perform model stealing to build a substitute model
- ❑ **Generate adversarial examples in feature space to identify δ**
- ❑ Apply successful perturbations in problem space through C2 operations

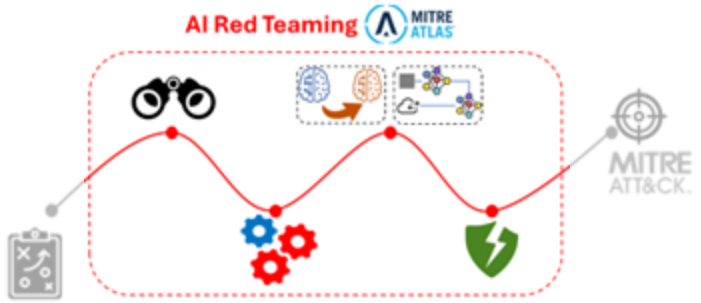


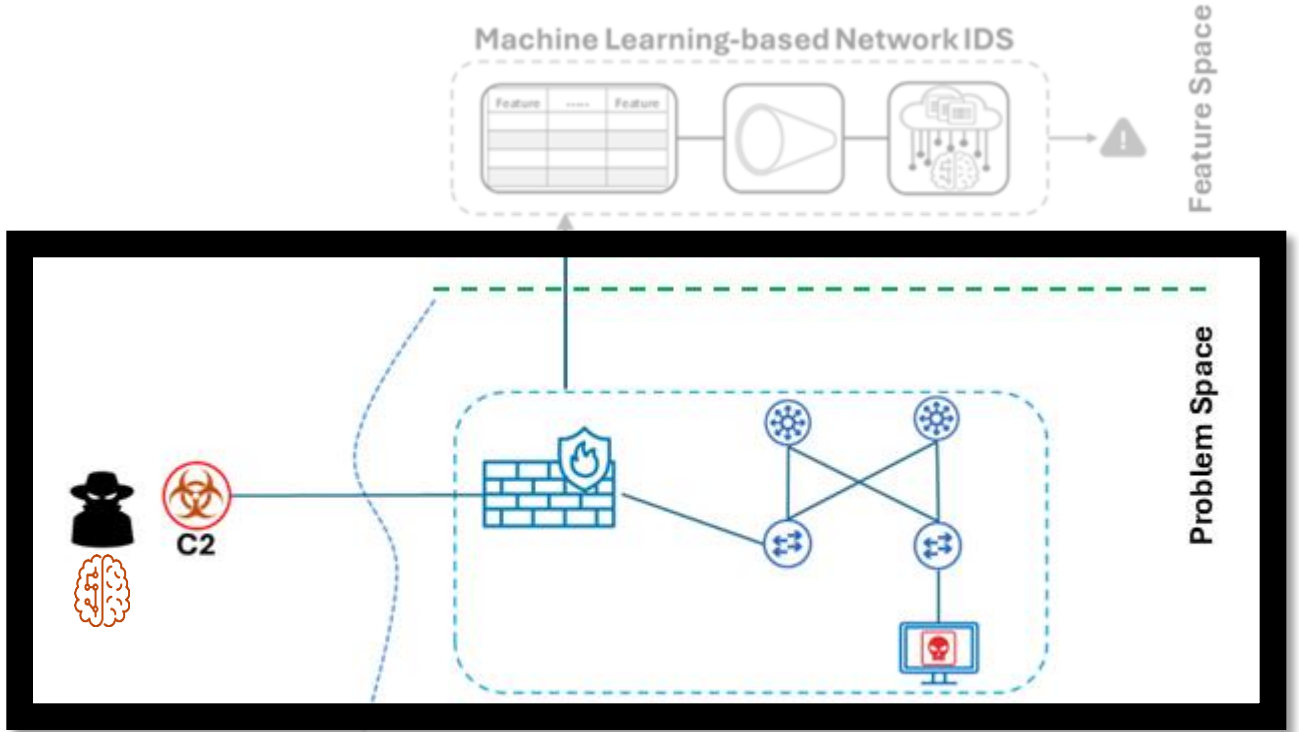


AI Red Teaming Against ML-NIDS

Overall attack strategy

- ❑ Perform model stealing to build a substitute model
- ❑ Generate adversarial examples in feature space to identify δ
- ❑ **Apply successful perturbations in problem space through C2 operations**

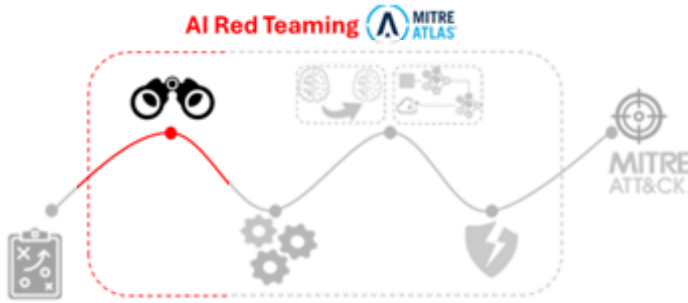


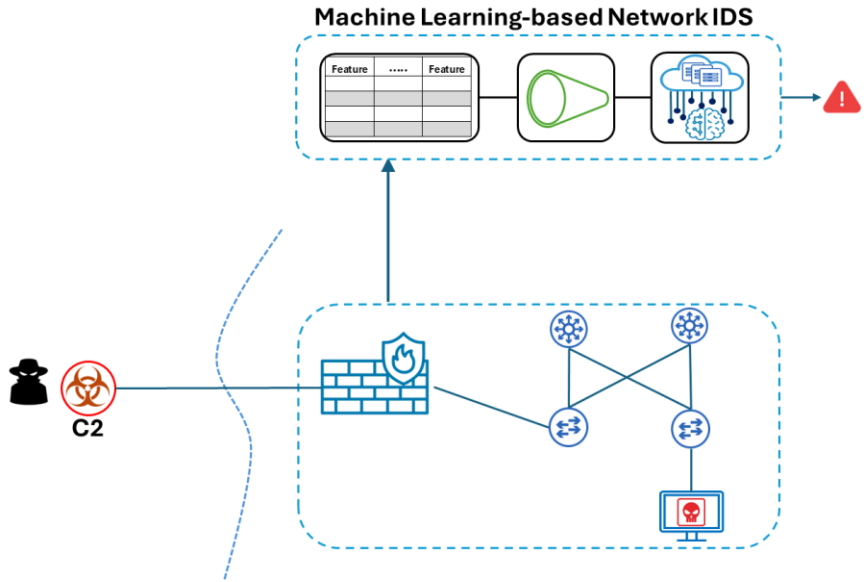


Reconnaissance

Gather information about the target **AI** system (passive/active AML.TA0002)

- E.g., to avoid detection in active interactions to the target
  - Purchasing similar ML-NIDS
  - Subscribe to get access to APIs (cloud-based solutions)
  - Leverage infected machine

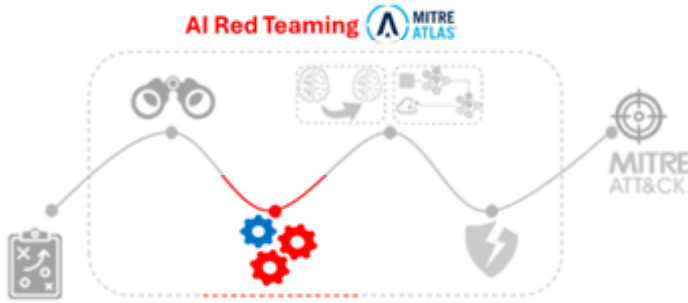


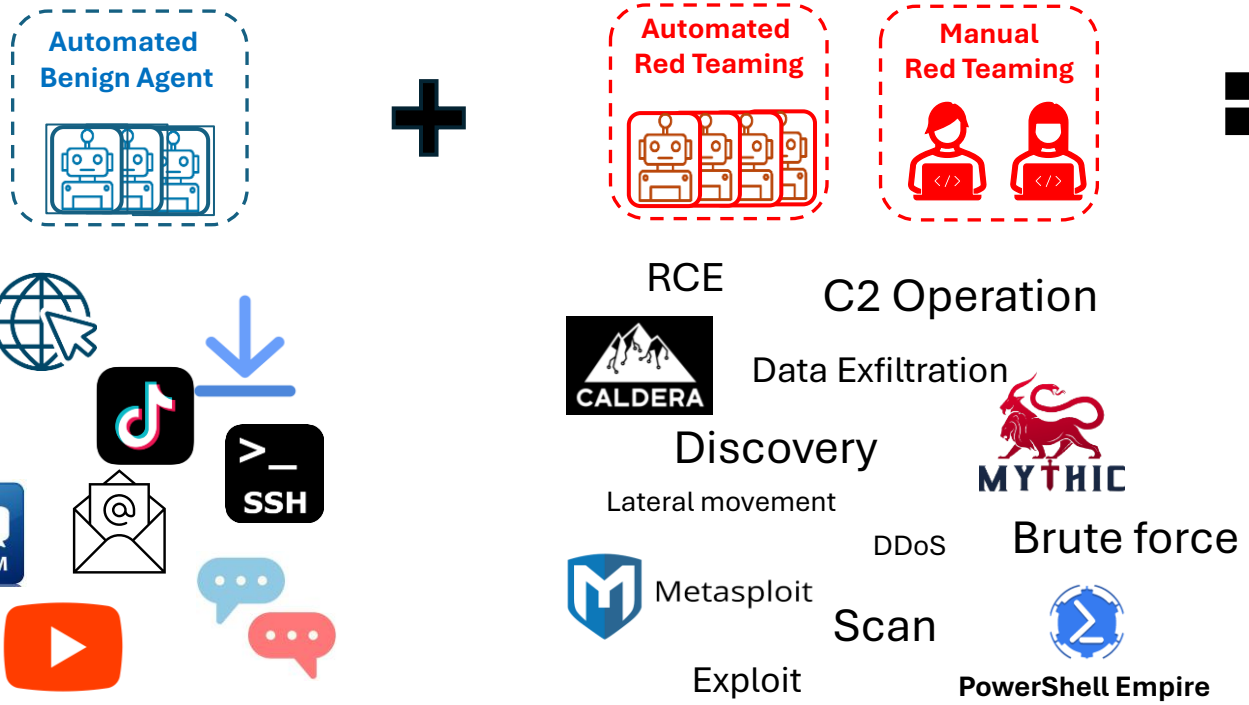


Resource Development

Establish resources they can use to support operations (AML.TA0003)

- Data, infrastructure, tools, etc.
  - E.g., Network traffic (Benign & Malicious)





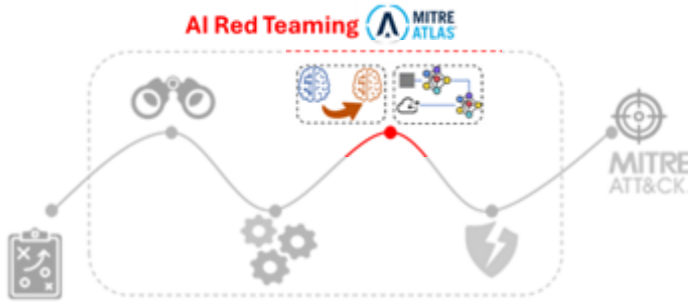
Open-Source Network Traffic Dataset

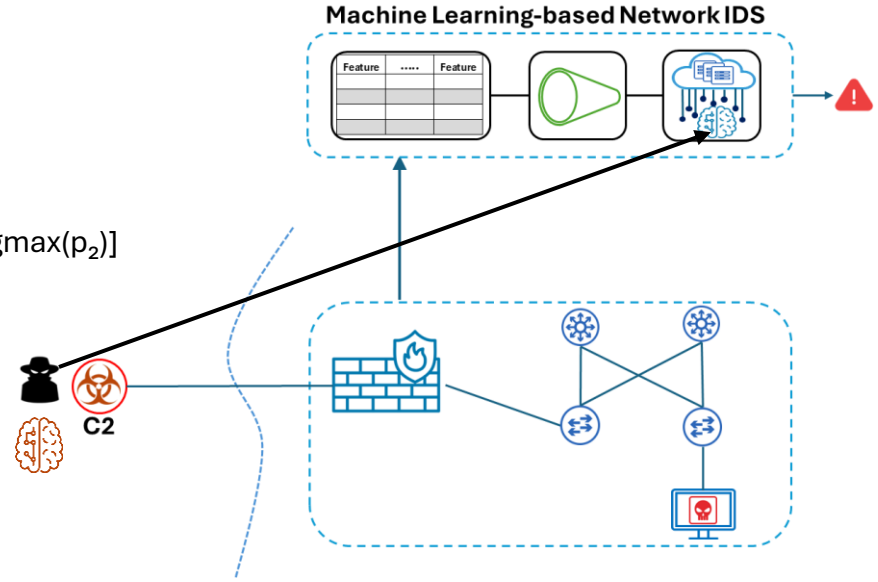
Technical University of Chemnitz RedTeam30 (TUC-RedTeam30)

AI Attack Staging

Access to the target system to tailor the attack (AML.TA0003)

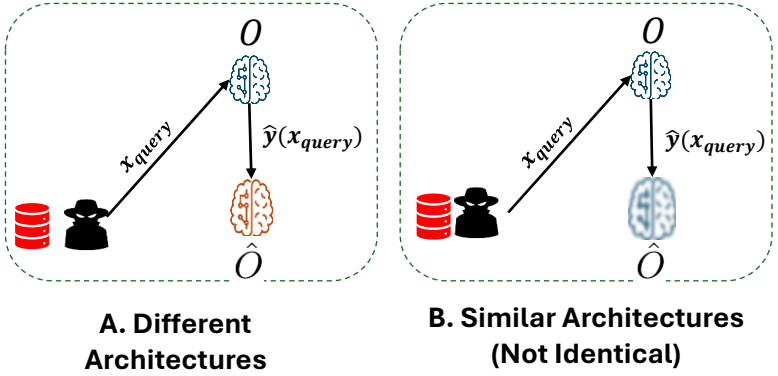
- ❑ **Create Proxy AI Model (AML.T0005)**
- ❑ Craft Adversarial Data in Feature Space (AML.T0043)
- ❑ Verify Attack (AML.T0042)





Agreement Score
$S(p_1, p_2) = \mathbb{1}[\text{argmax}(p_1) = \text{argmax}(p_2)]$

Query-based model stealing [1] is a **top-3** confidentiality threat, according to a survey of 28 organizations [2].



A. Different Architectures

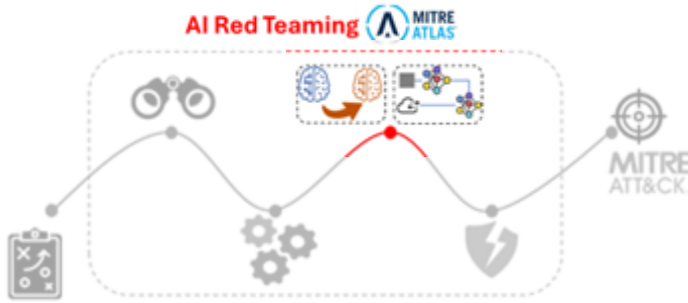
B. Similar Architectures (Not Identical)

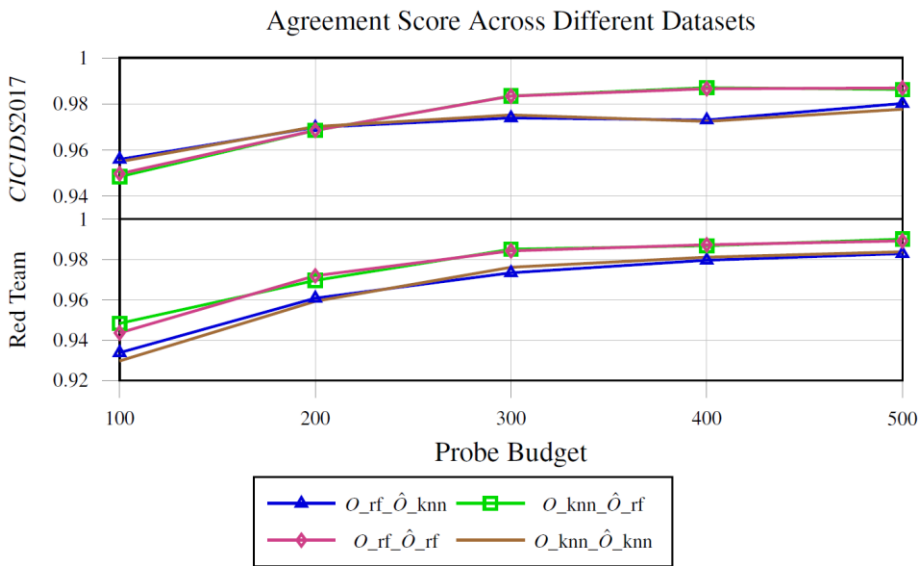
[1] Jagielski et al., High accuracy and high fidelity extraction of neural networks
[2] Shankar et al. , Adversarial machine learning-industry perspectives

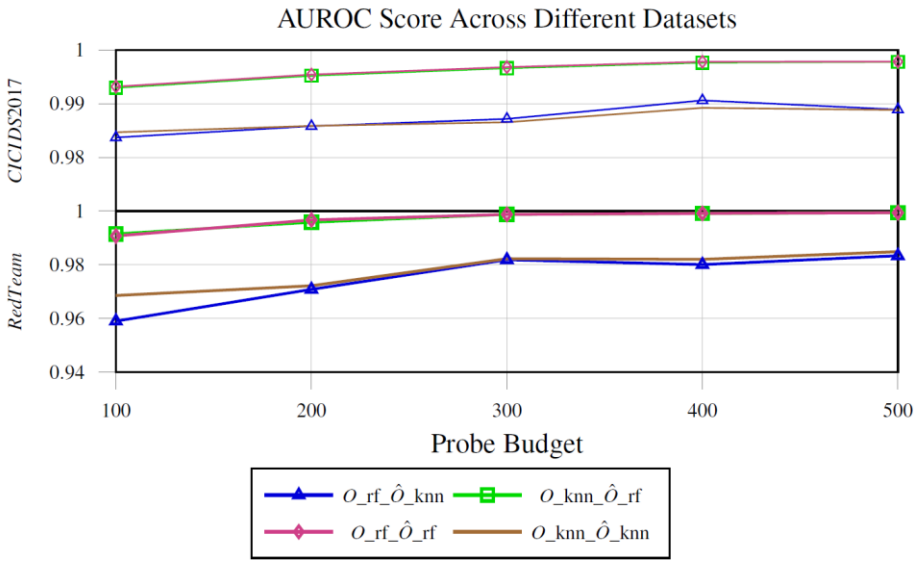
AI Attack Staging

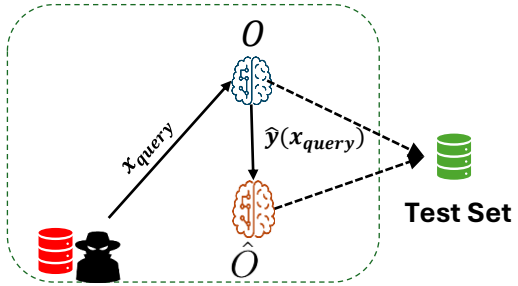
Access to the target system to tailor the attack (AML.TA0003)

- ❑ **Create Proxy AI Model (AML.T0005)**
- ❑ Craft Adversarial Data in Feature Space (AML.T0043)
- ❑ Verify Attack (AML.T0042)







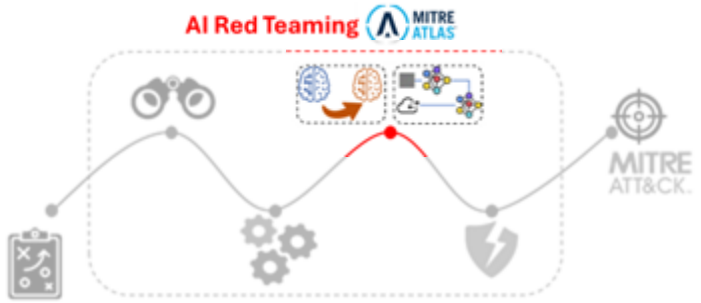


[1] Sharafaldin et al., Toward generating a new intrusion detection dataset and intrusion traffic characterization
[2] Liu et al. Error prevalence in nids datasets: A case study on cic-ids-2017 and cse-cic-ids-2018.

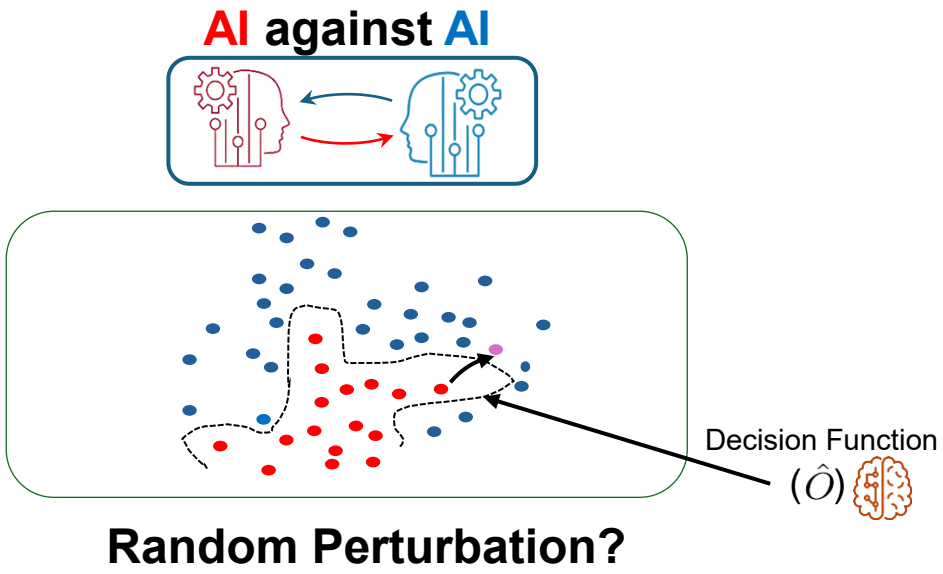
AI Attack Staging

Access to the target system to tailor the attack (AML.TA0003)

- ❑ Create Proxy AI Model (AML.T0005)
- ❑ **Craft Adversarial Data in Feature Space (AML.T0043)**
- ❑ Verify Attack (AML.T0042)



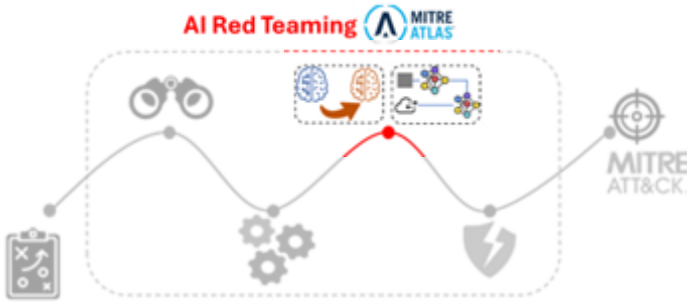
$$\max Loss\ (F_\theta(x+\delta), y)$$
$$\delta \in \Delta\ , y = 0$$

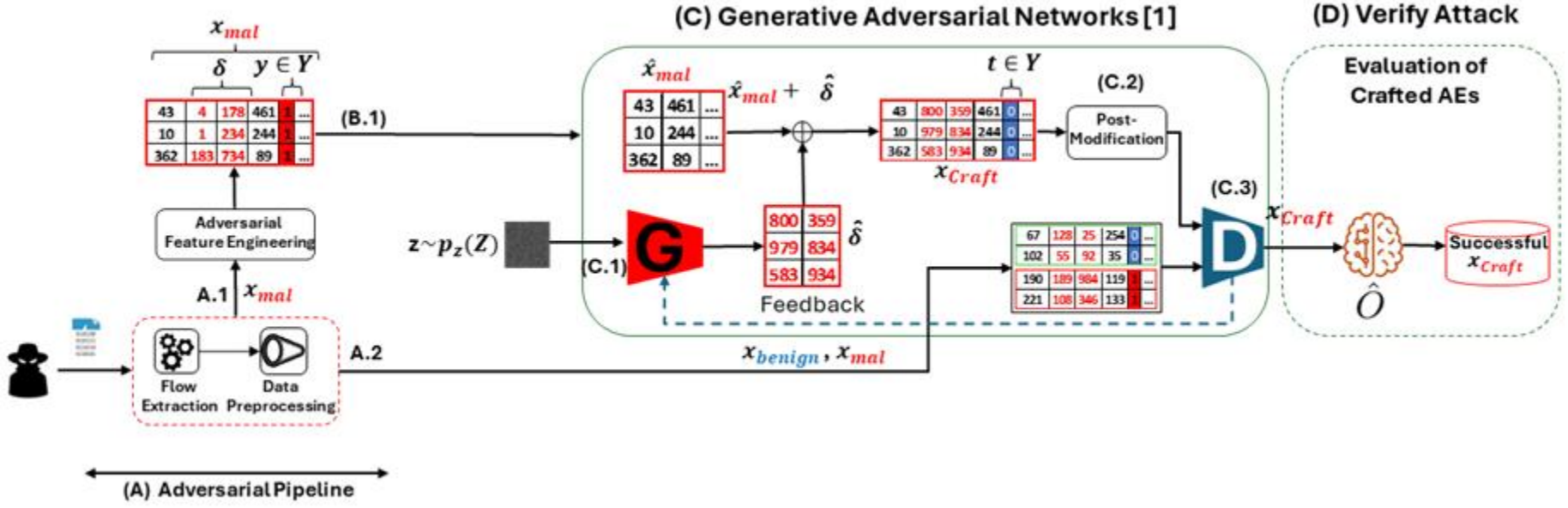


AI Attack Staging

Access to the target system to tailor the attack (AML.TA0003)

- ❑ Create Proxy AI Model (AML.T0005)
- ❑ **Craft Adversarial Data in Feature Space (AML.T0043)**
- ❑ Verify Attack (AML.T0042)



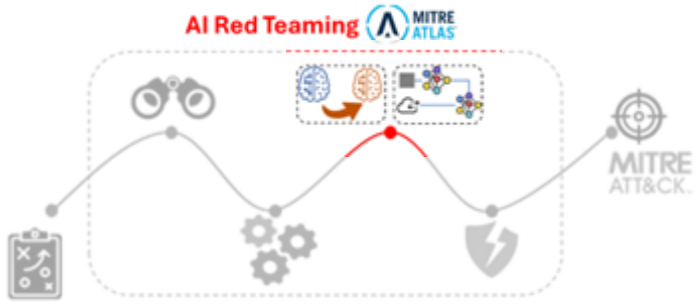


[1] Ian Goodfellow et al., Generative adversarial nets

AI Attack Staging

Access to the target system to tailor the attack (AML.TA0003)

- ❑ Create Proxy AI Model (AML.T0005)
- ❑ **Craft Adversarial Data in Feature Space (AML.T0043)**
- ❑ Verify Attack (AML.T0042)



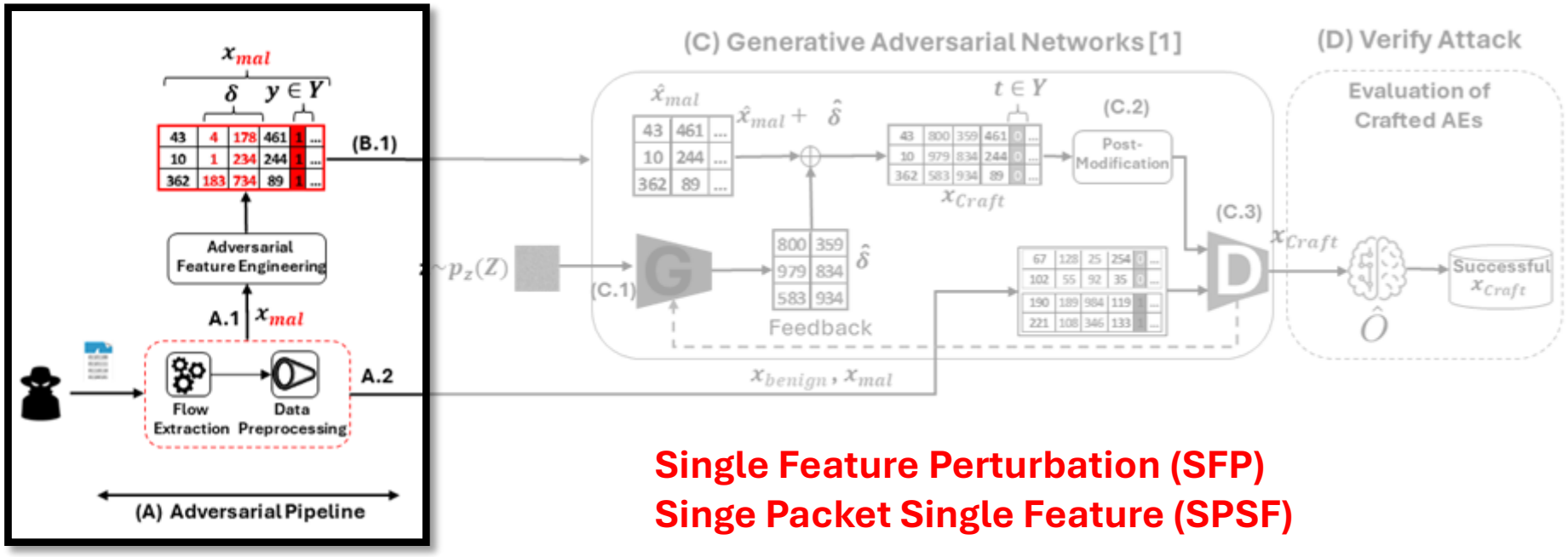
dst2src_max_ps

src2dst_packets

dst2src_packets

dst2src_bytes

src2dst_bytes



Single Feature Perturbation (SFP)

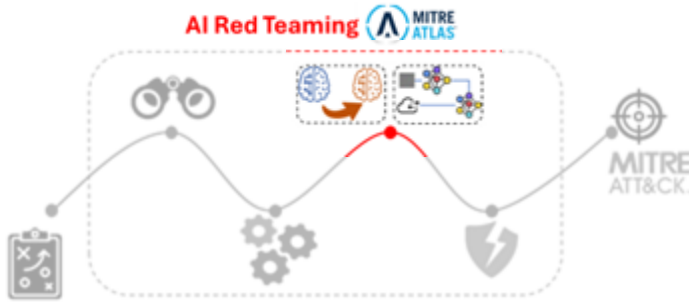
Singe Packet Single Feature (SPSF)

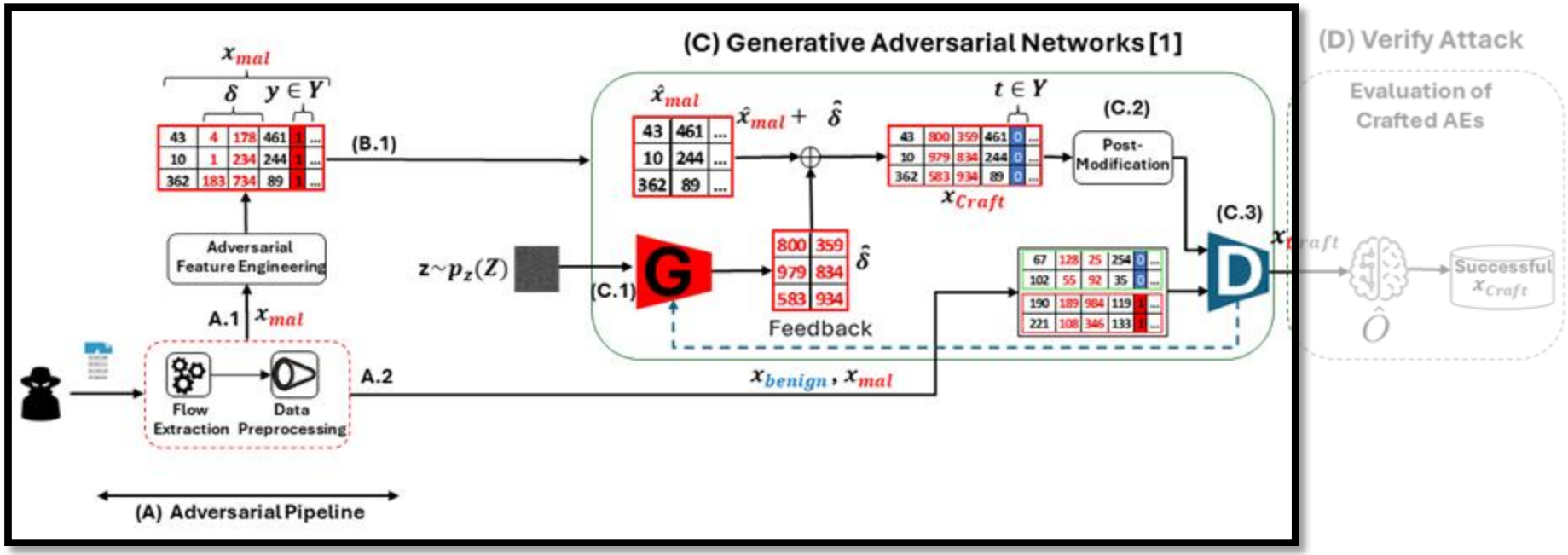
[1] Ian Goodfellow et al., Generative adversarial nets

AI Attack Staging

Access to the target system to tailor the attack (AML.TA0003)

- ❑ Create Proxy AI Model (AML.T0005)
- ❑ **Craft Adversarial Data in Feature Space (AML.T0043)**
- ❑ Verify Attack (AML.T0042)

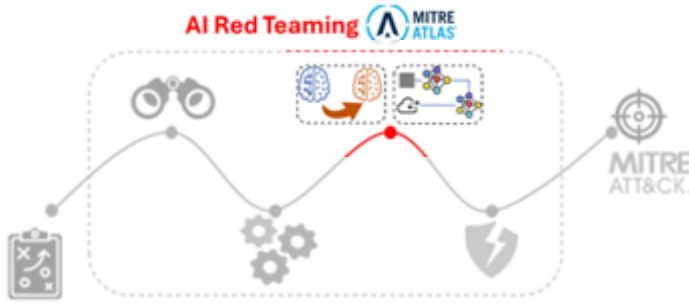


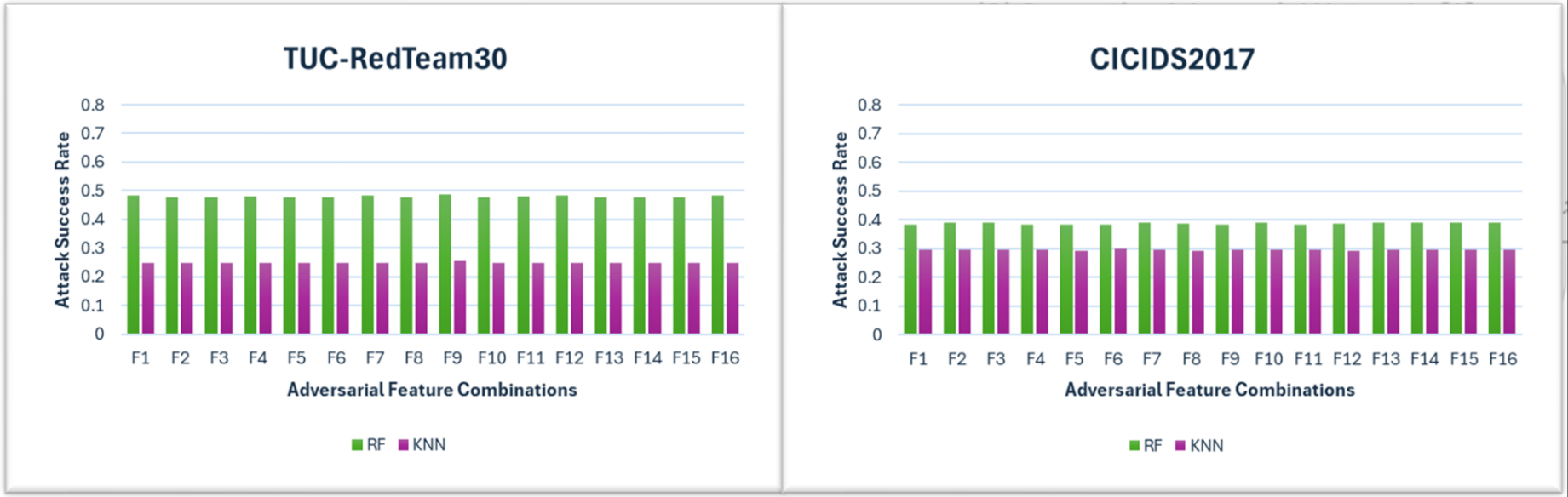


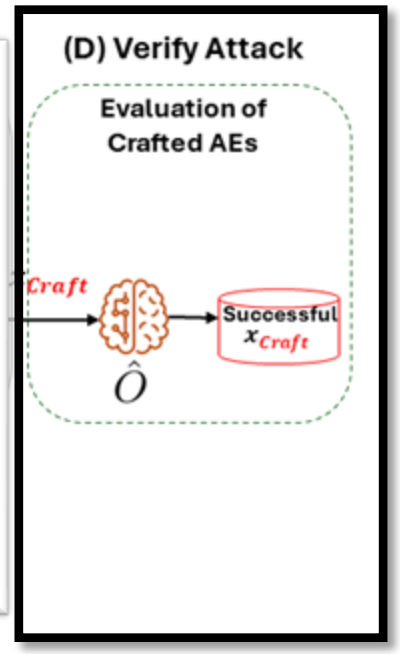
AI Attack Staging

Access to the target system to tailor the attack (AML.TA0003)

- ❑ Create Proxy AI Model (AML.T0005)
- ❑ Craft Adversarial Data in Feature Space (AML.T0043)
- ❑ **Verify Attack (AML.T0042)**







What Feature	What Values
δ	$\hat{\delta}$

801 801 934 946 359 583

Defense Evasion

Requirements

- ❑ Realistic Threat Model
- ❑ Adherence to TCP/IP
- ❑ Attack Functionality Preservation
- ❑ Pipeline-Independent Transferability

Threat Model

Attacker Goal: C2 Operations (e.g., RCE, Exfiltration)

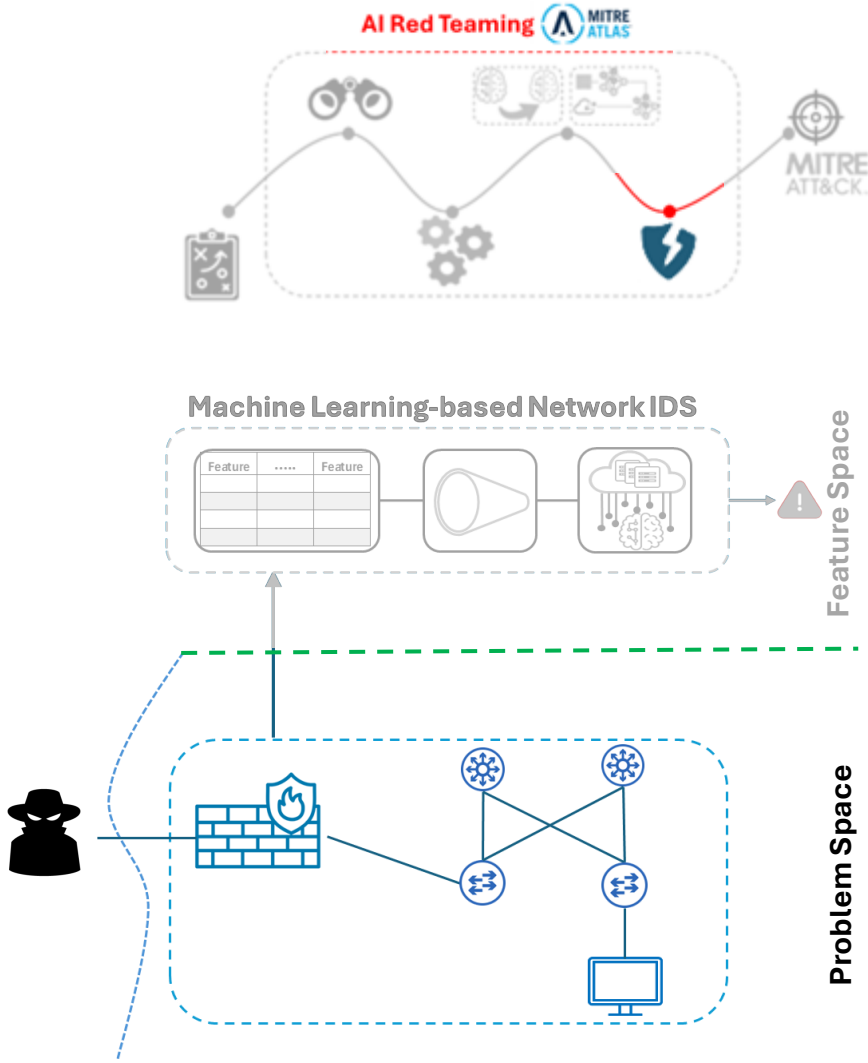
Attacker Location: Outside

Target Model: FlowTransformer [1], SSCL-IDS [2], CNN, RF, KNN

Attacker Knowledge

- ❑ Feature: No
- ❑ Training data: No
- ❑ Model Architecture: No

Domain Space: Problem Space

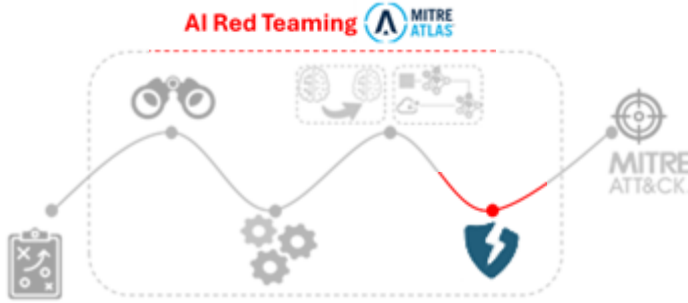


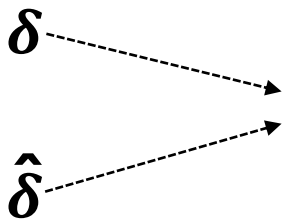
[1] Manocchio et al. , A transformer framework for flow-based network intrusion detection systems
[2] Golchin et al., Sscl-ids: Enhancing generalization of intrusion detection with self-supervised contrastive learning

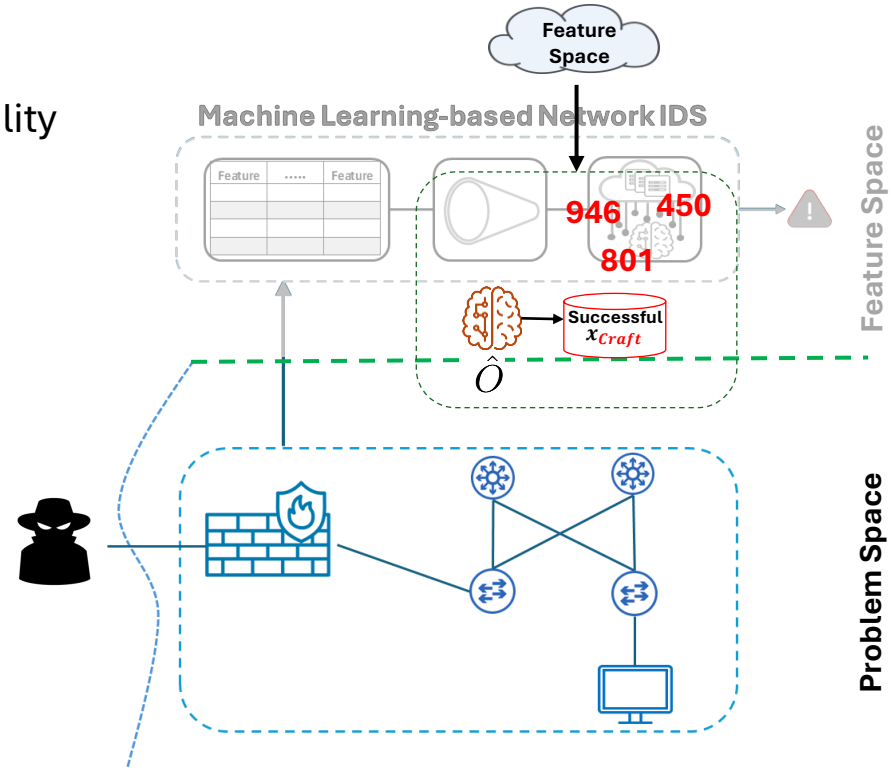
Defense Evasion

Requirements

- ❑ Realistic Threat Model
- ❑ Adherence to TCP/IP
- ❑ Attack Functionality Preservation
- ❑ Pipeline-Independent Transferability





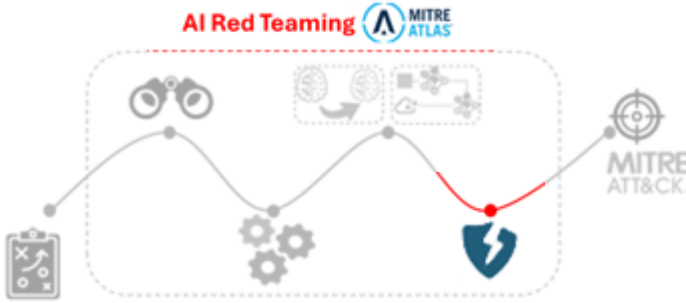


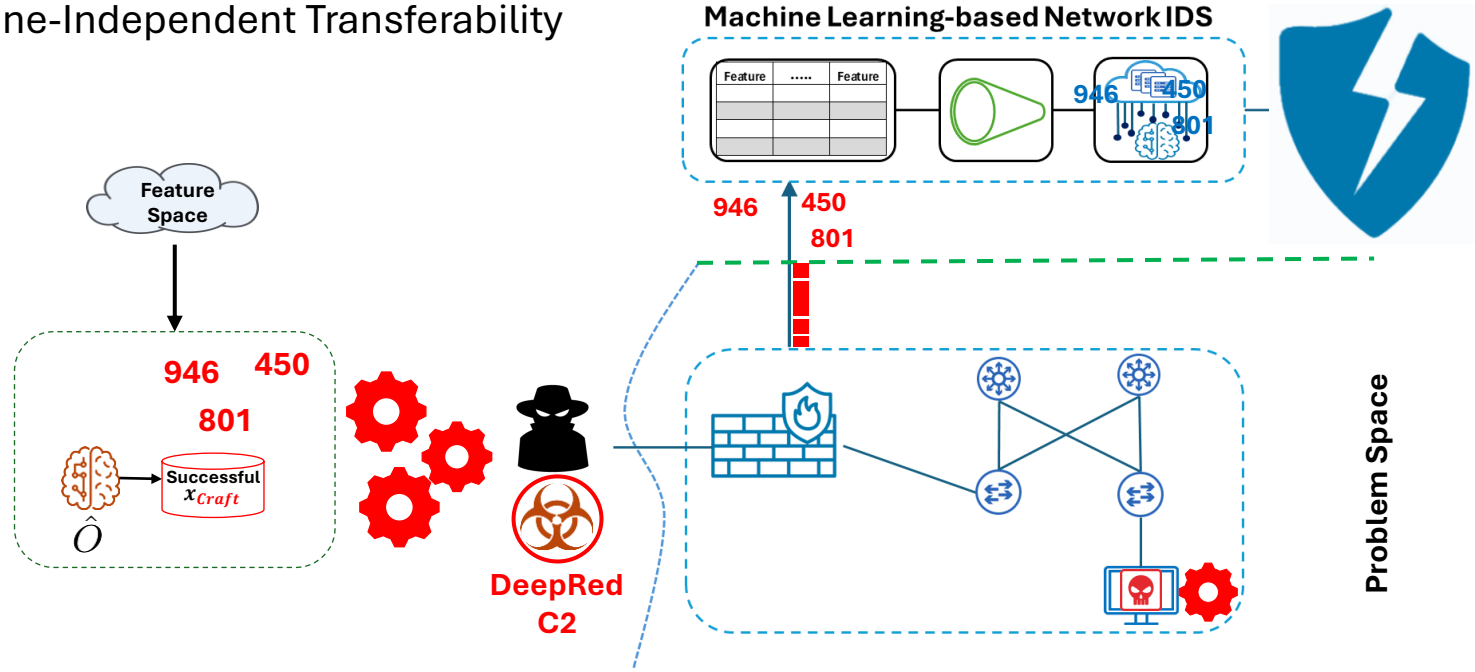
[1] Manocchio et al. , A transformer framework for flow-based network intrusion detection systems
[2] Golchin et al., Sscl-ids: Enhancing generalization of intrusion detection with self-supervised contrastive learning

Defense Evasion: DeepRed C2

Requirements

- ❑ Realistic Threat Model
- ❑ Adherence to TCP/IP
- ❑ Attack Functionality Preservation
- ❑ Pipeline-Independent Transferability



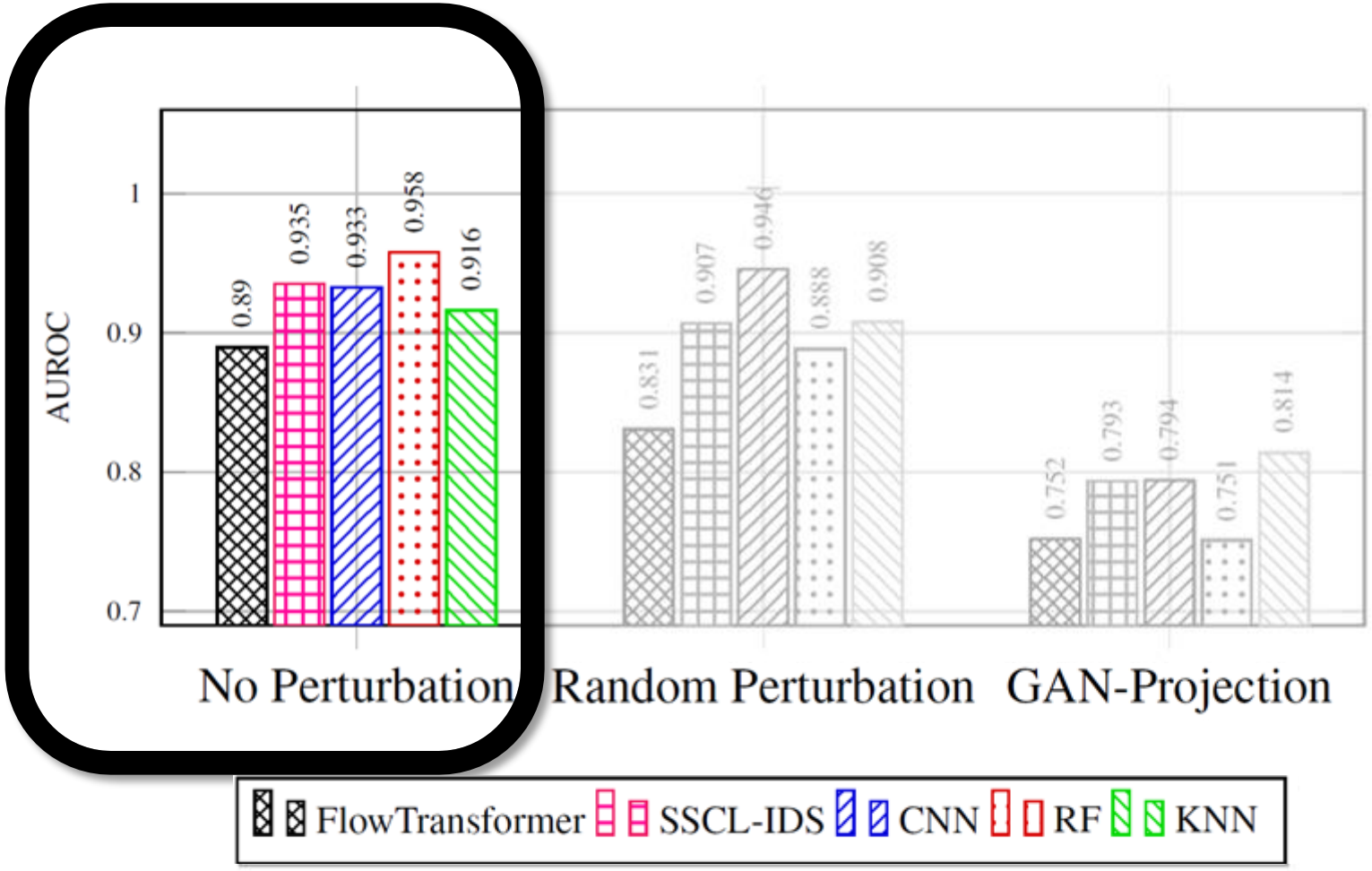


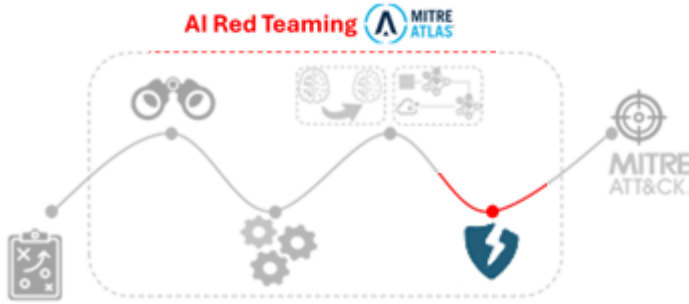
DeepRed C2 Framework

- ❑ HTTP and WebSocket
- ❑ Traffic Perturbation
  - ❑ Packet Injection
  - ❑ Padding

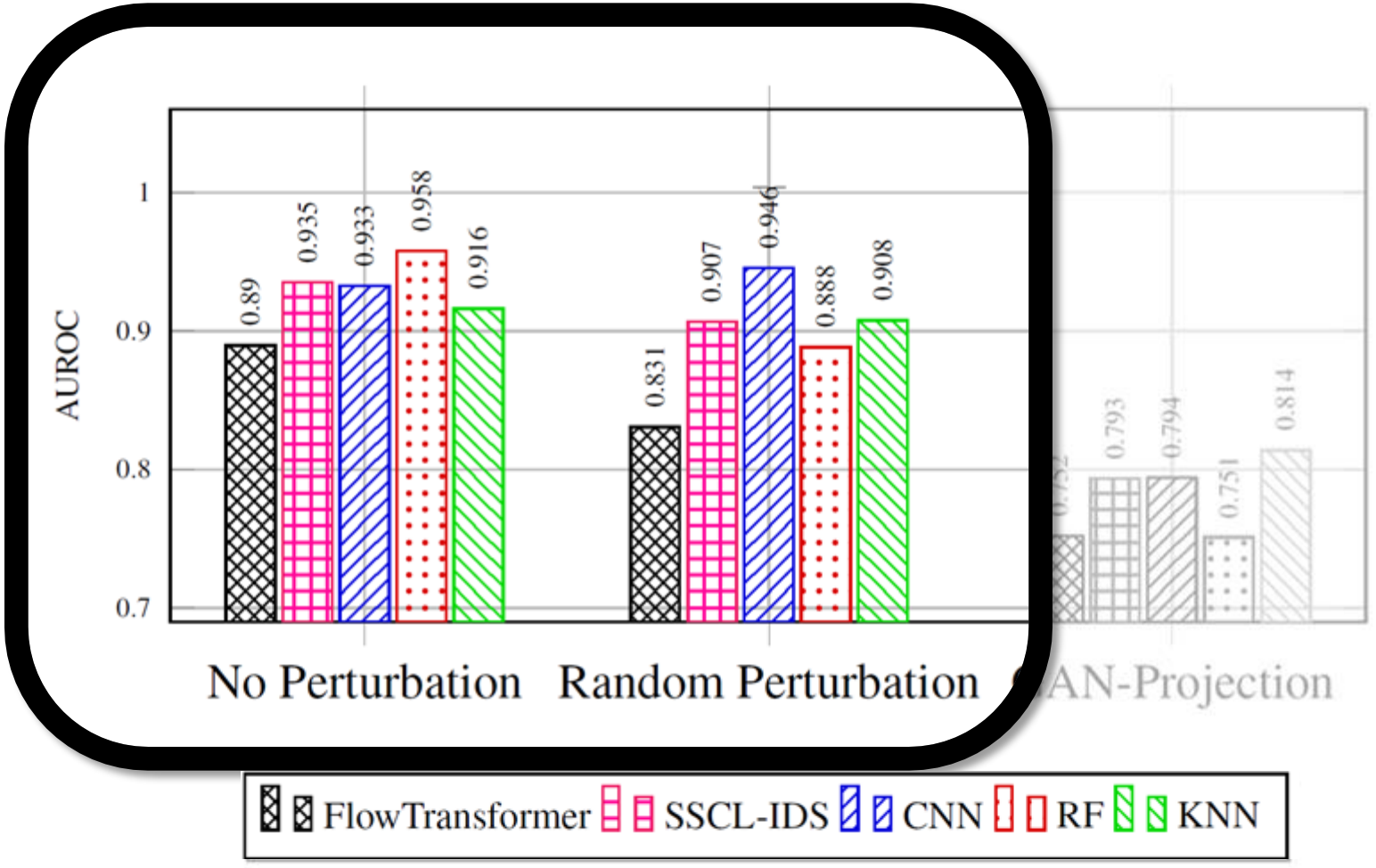
[1] Manocchio et al. , A transformer framework for flow-based network intrusion detection systems
[2] Golchin et al., Sscl-ids: Enhancing generalization of intrusion detection with self-supervised contrastive learning

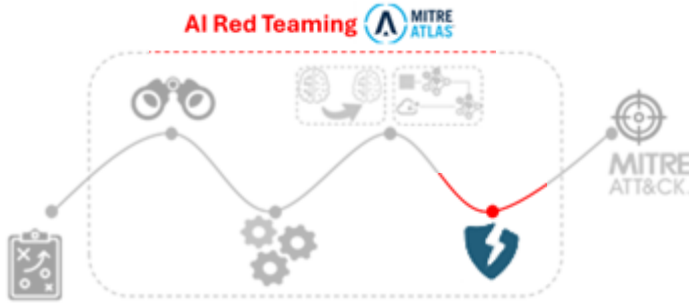
Defense Evasion: DeepRed C2



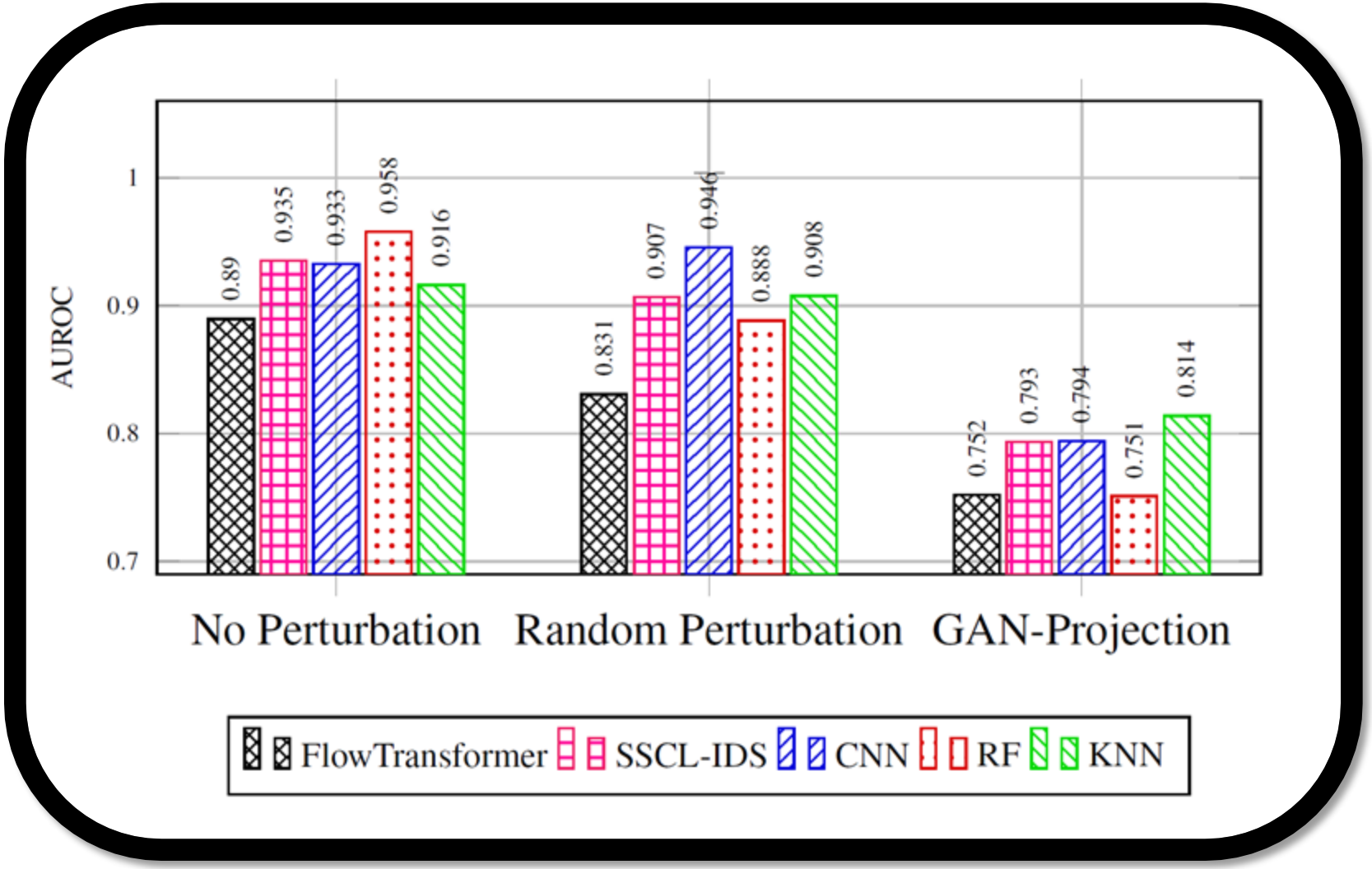


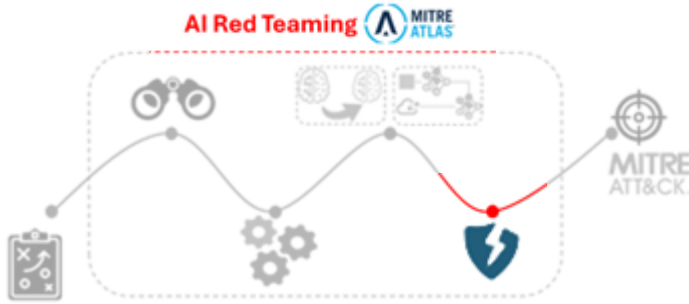
Defense Evasion: DeepRed C2



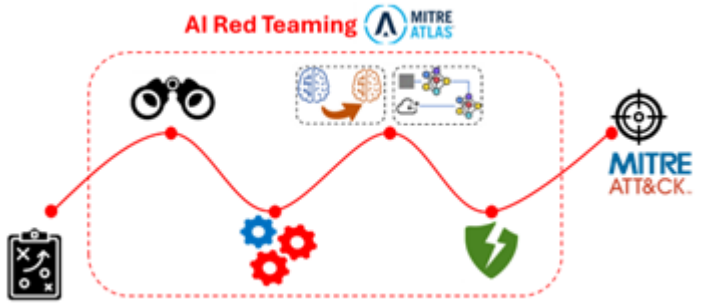


Defense Evasion: DeepRed C2





Summary & Conclusion



Summary

- ✓ Conducted an end-2-end multi-stage red teaming of ML-NIDS
- ✓ Proposed two novel attack strategies: **SFP** and **SPSF**
- ✓ Introduced **DeepRed** to bridges feature and problem spaces
- ✓ Released the **TUC-RedTeam30** NIDS benchmarking dataset

Conclusion

- ✓ Comprehensive threat modeling is the key to exposing diverse practical attack vectors
- ✓ Preserving attack functionality is essential to validate intended malicious behavior
- ✓ Pipeline-independent transferability evaluation is essential due to variability among ML-NIDS

DeepRed C2

TUC-RedTeam30 Dataset

Thanks For Your Attentions