UMBC

Evading Malware Classifiers via Monte Carlo Mutant Feature Discovery

John Boutsikas, Maksim E. Eren, Charles Varga, Edward Raff, Cynthia Matuszek, and Charles Nicholas
(iboutsi1, meren1, cvarga1, cmat, nicholas)@umbc.edu; raff_edward@bah.com

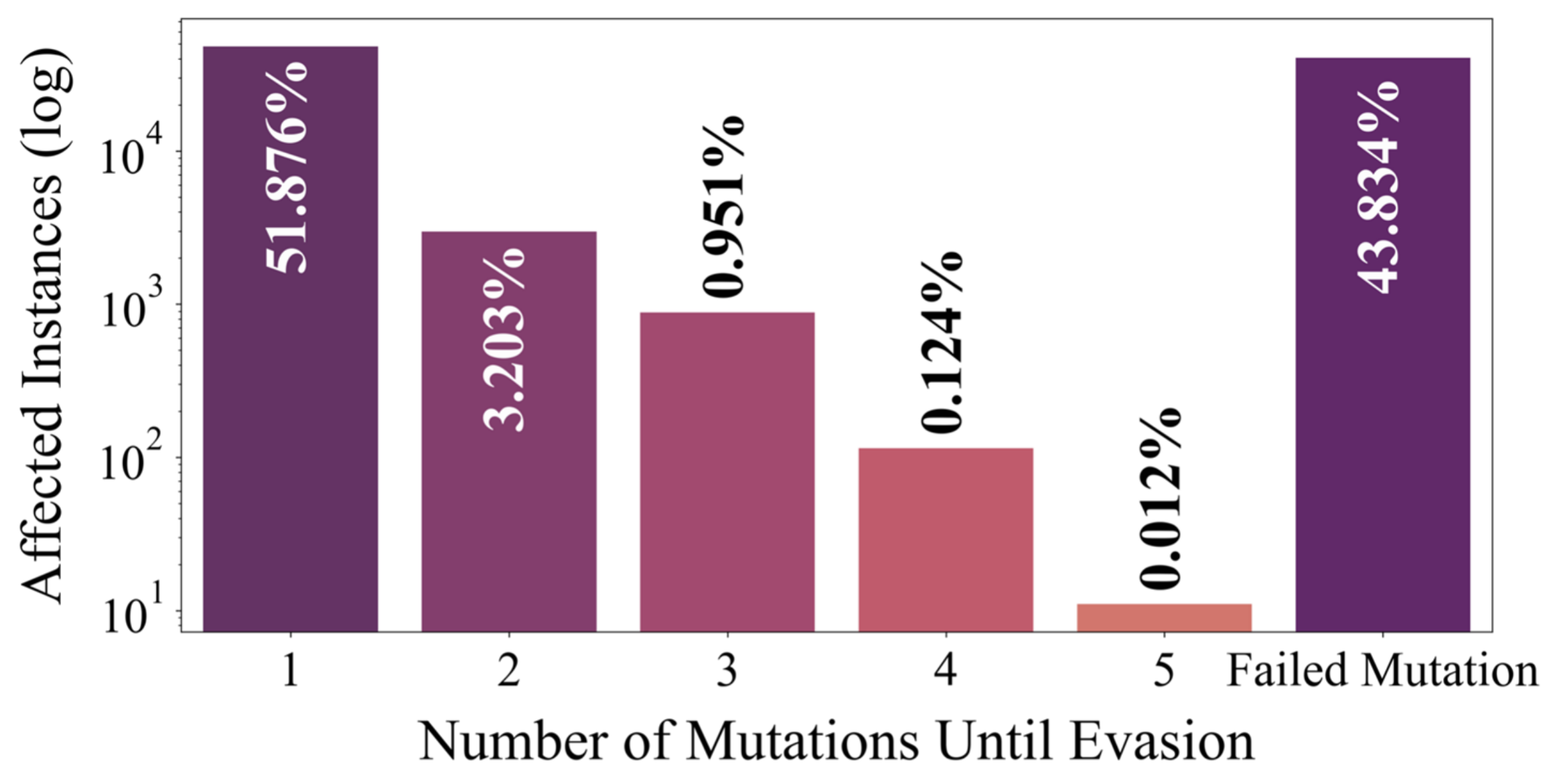
DREAM LAB

Anti-virus (AV) vendors use machine learning (ML) for malware detection,[1,2] and ML based intrusion detection influences the cyber kill chain.[3] **Studying classifier evasion strategies dictates cyber defense against malice.**[5] We stage a grey-box setup to analyze a scenario where a malicious actor trains a model to discover the mutations that misclassify an instance using Monte Carlo Tree Search (MCTS).

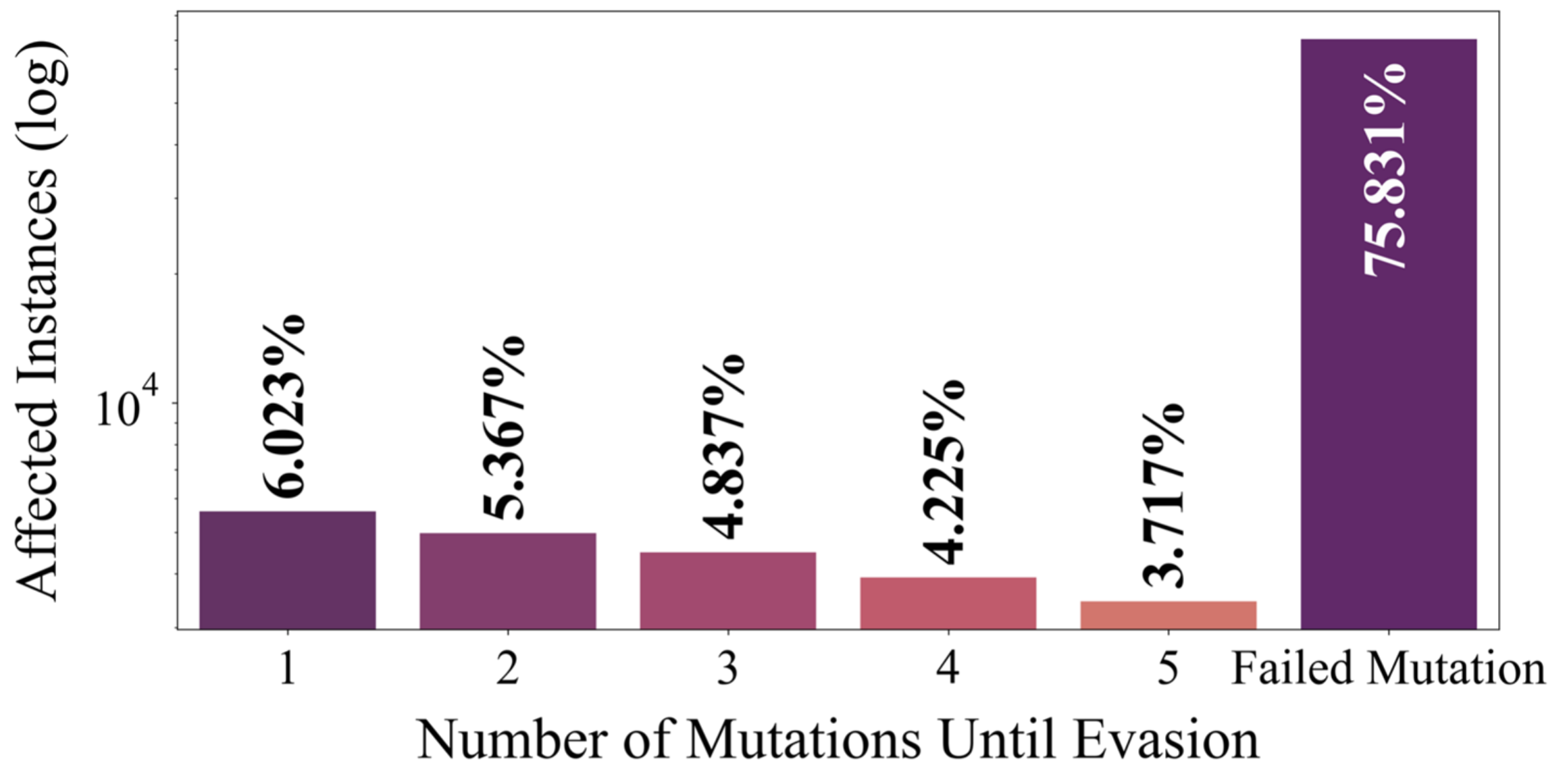
Approach: Monte Carlo Tree Search

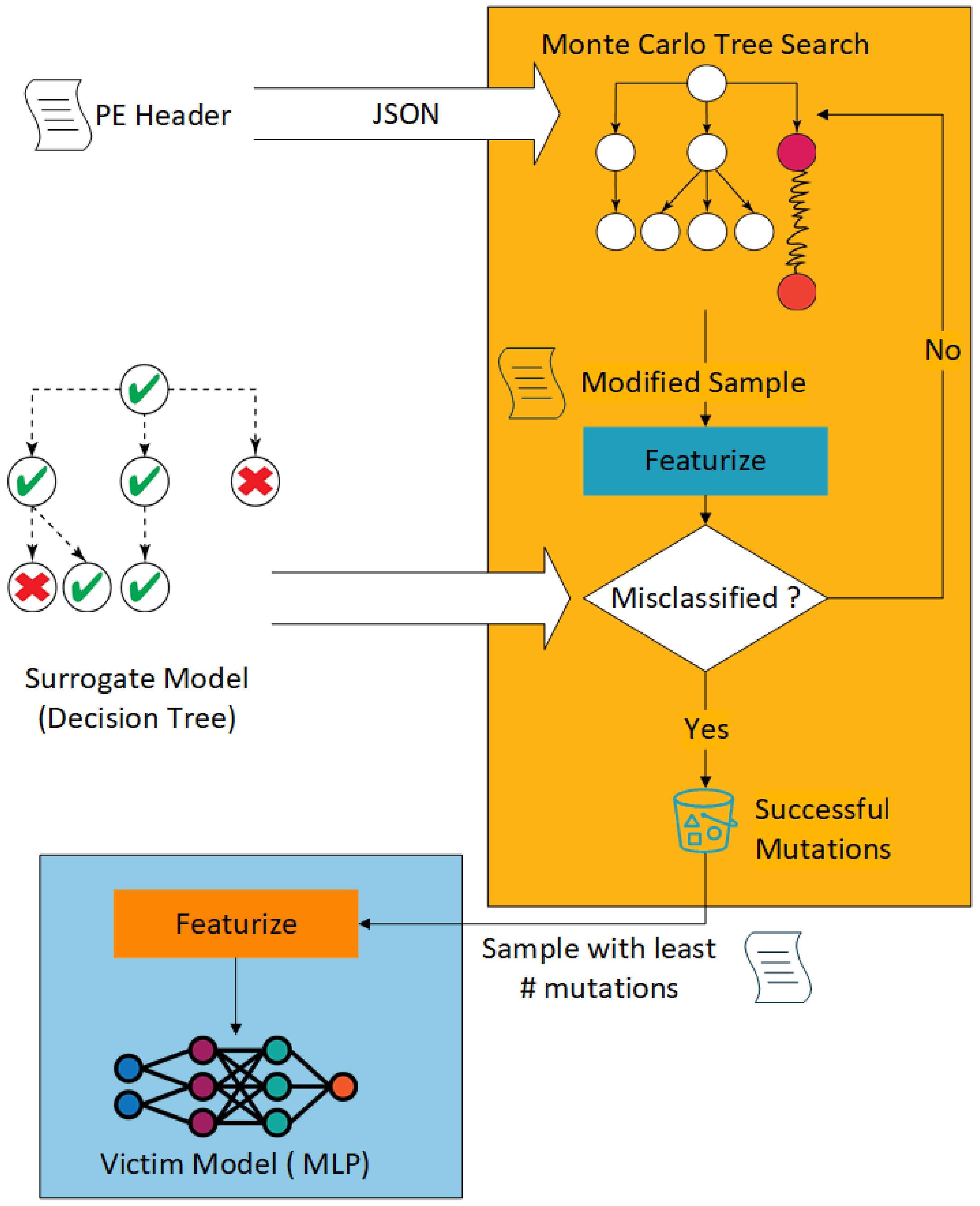
- **Classifier evasion is like a chess game** between the adversary and the victim
- The winning 'board' is a successful mutation that makes the malware undetectable
- MCTS **examines mutations without computing all possible permutations** of malware feature changes
- Empirical evaluation searches a comprehensive set of mutations with minimal divergence from the true results

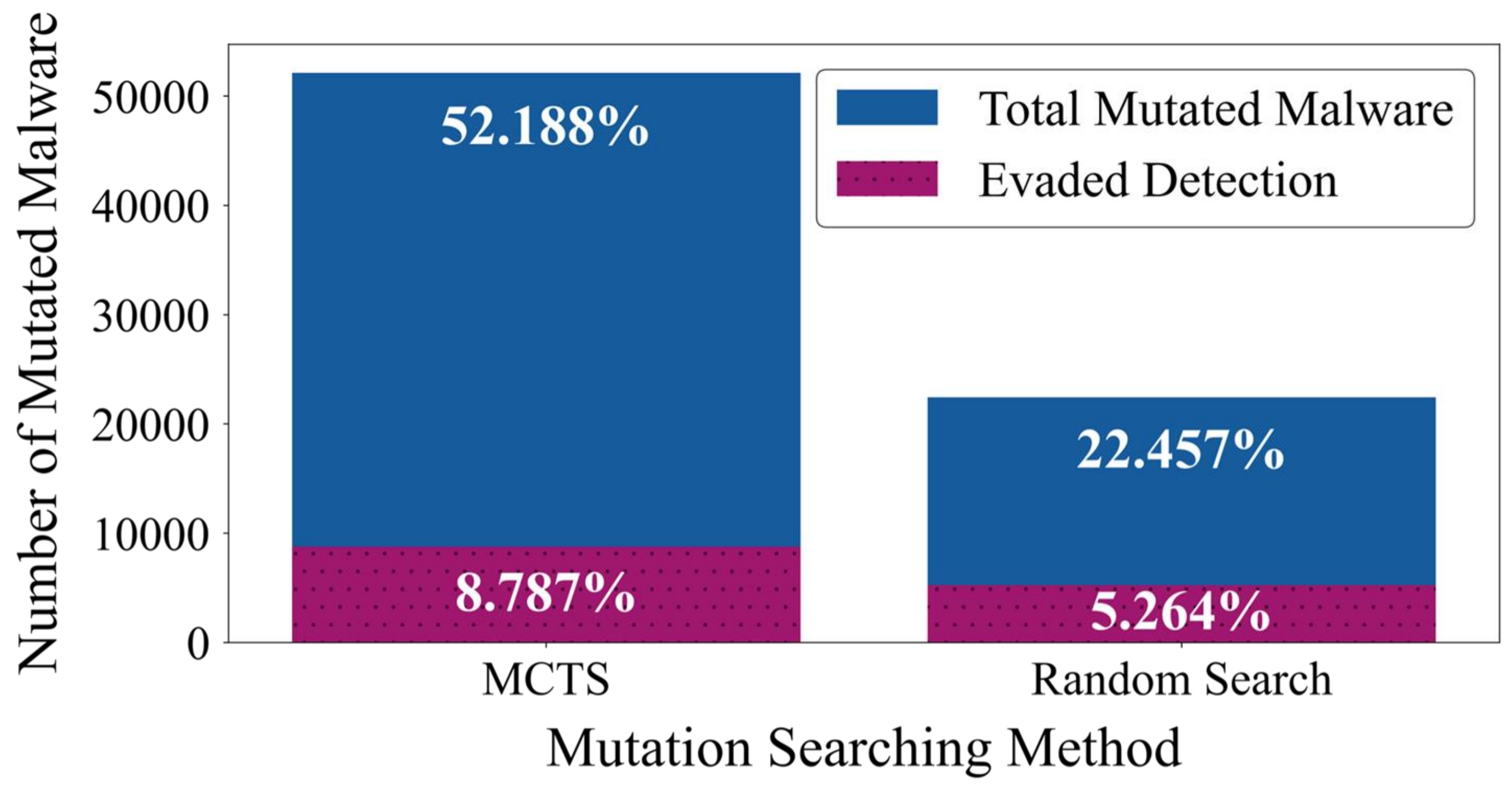
MCTS



Random Search







Adversarial Setup: Gray Box

- Use different subsets of EMBER-2018 dataset [4] to train victim and surrogate models
- Attacker trains a surrogate Decision Tree
- MCTS confirms the evasive feature modifications using the surrogate model
- Organization AV systems are not public
- **Attacker does not need to query AV APIs**
- Mutations are then evaluated against the victim Multi-layer Perceptron (MLP) that takes the place of the target AV
- Scenario is feasible for a **malicious actor avoiding attention**

Cornell University
arXiv.org

SCAN ME

https://arxiv.org/abs/2106.07860

DREAM LAB

SCAN ME

umbc-dream-lab.github.io

[1] Microsoft 365 Defender Threat Intelligence Team. Microsoft researchers work with Intel labs to explore new deep learning approaches for malware classification, www.microsoft.com/security/blog. 2020.

[2] B. Quintero. Virustotal += sangfor engine zero, 2019; Virustotal += bitdefender theta. 2019.

[3] T. N. Nguyen. Attacking machine learning models as part of a cyber kill chain. Arxiv. 2017.

[4] H. Anderson, P. Roth. Ember: An open dataset for training static PE malware machine learning models. ArXiv. 2018.

[5] W. Song et al. *Automatic generation of adversarial examples for interpreting malware classifiers. ArXiv. 2020.*

Presented at the *Malware Technical Exchange Meeting* (MTEM), Online, July 13-15, 2021.