Explanation-Guided Backdoor Poisoning Attacks Against Malware Classifiers

Giorgio Severi – Northeastern University

Jim Meyer - Xailient

Scott Coull - FireEye

Alina Oprea – Northeastern University

USENIX Security – August 2021

Machine Learning for Malware Detection

- Static ML models play key role in pre-execution malware prevention
- Volume and diversity of executables makes training challenging
- Crowdsourced threat feeds provide an ideal source for training data

SentinelOne blog

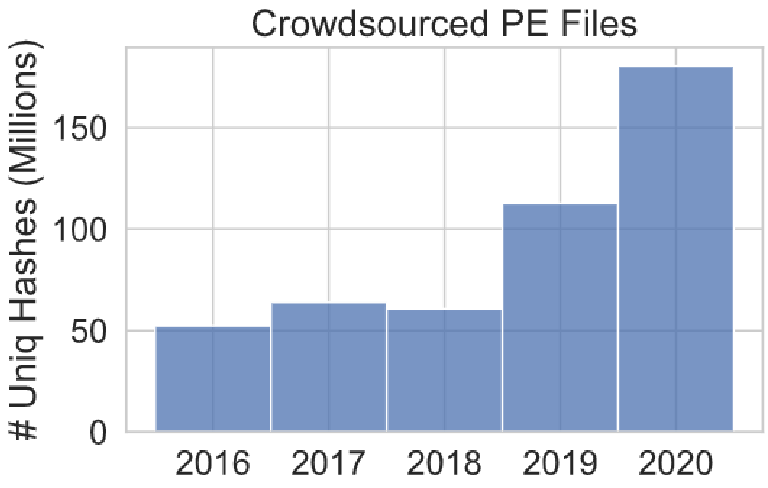
Detecting Malware Pre-execution with Static Analysis and Machine Learning

CROWDSTRIKE

Why Machine Learning Is a Critical Defense Against Malware

FIREEYE

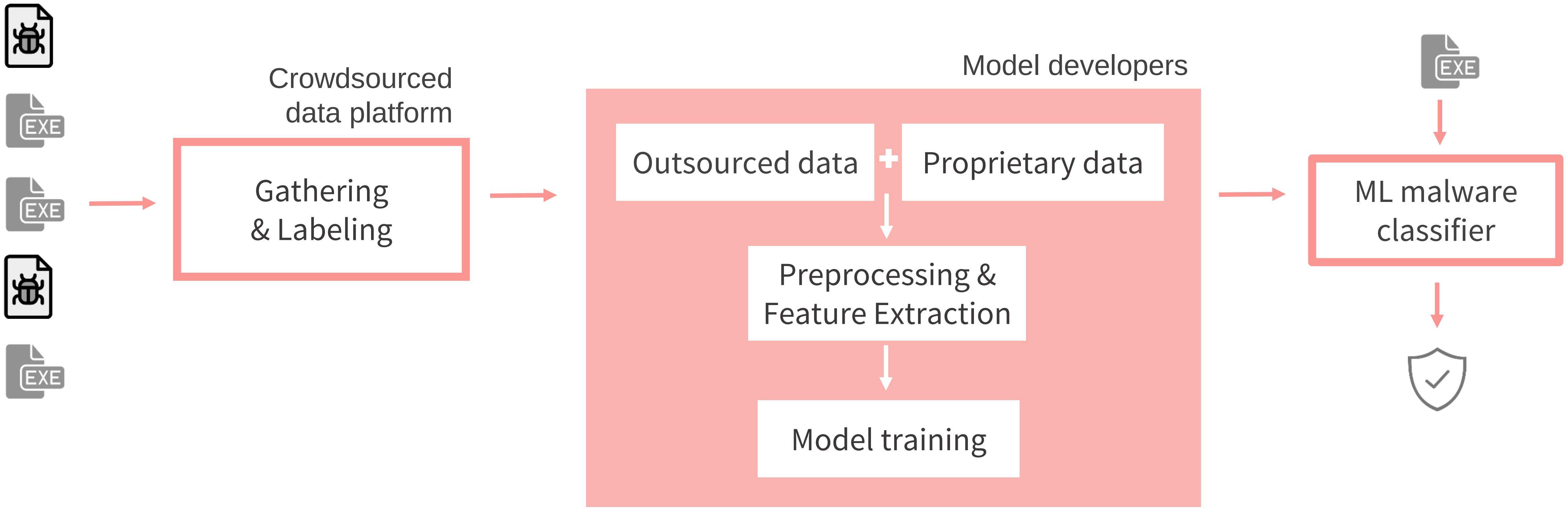
MalwareGuard: FireEye's Machine Learning Model to Detect and Prevent Malware



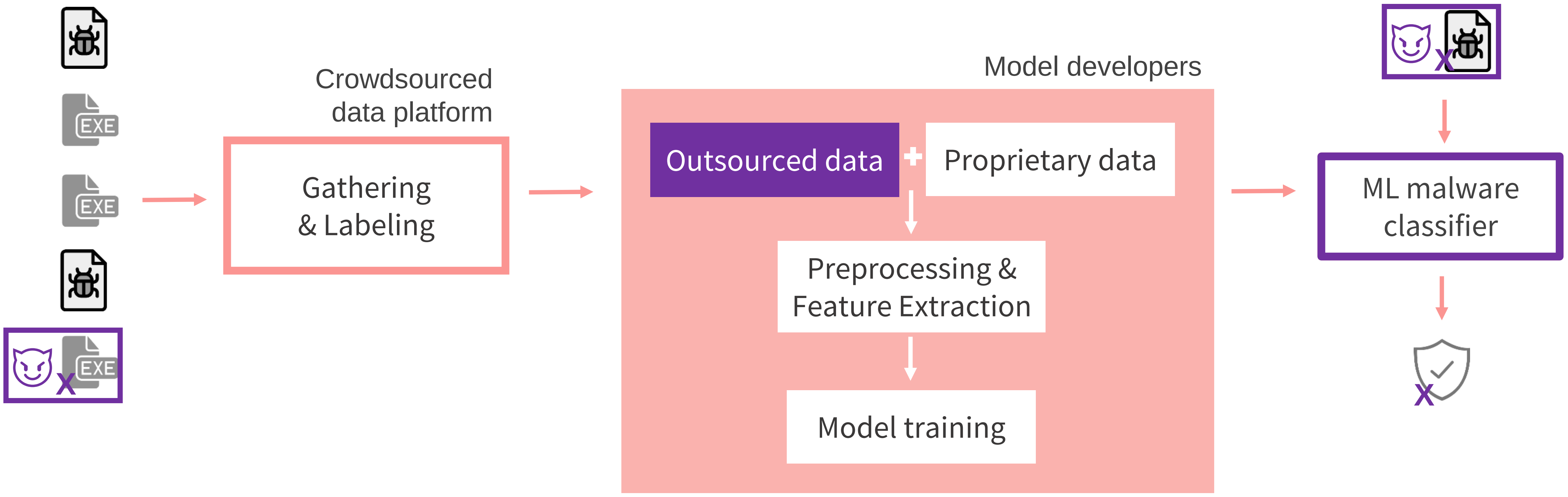
Our contributions

- New backdoor poisoning attacks targeting the supply chain of ML malware classifiers
- Model-agnostic methodology to generate backdoors using explainable ML techniques
- Functional poisoned binaries for multiple file types
- Attacks effective on a variety of models and difficult to mitigate using existing defensive strategies

System overview



System overview



Backdoor poisoning

Background

- Backdoor (Gu et al. 2017): associate a pattern (trigger) with a target class

Challenges:

- Attacker has no control over training labels - Clean-label (Shafahi et al. 2018)
- Must respect the constraints dictated by the data semantics

Image from Gu et al. 2017

Feature	LightGBM	EmberNN
major_image_version	1704	14
major_linker_version	15	13
major_operating_system_version	38078	8
minor_image_version	1506	12
minor_linker_version	15	6
minor_operating_system_version	5	4
minor_subsystem_version	5	20

Threat model

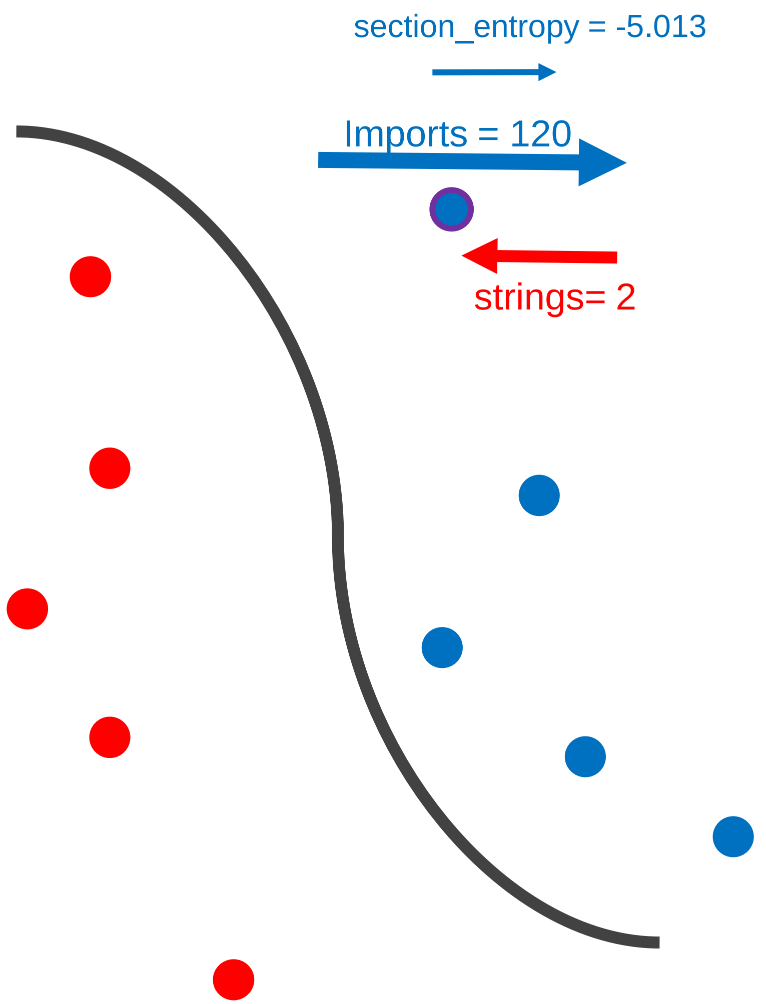
Attacker	Knowledge				Control	
	Feature Set	Model Architecture	Model Parameters	Training Data	Features	Labels
unrestricted	●	●	●	●	●	○
data_limited	●	●	●	◔	●	○
transfer	●	○	○	●	●	○
black_box	●	○	○	●	●	○
constrained	●	●	●	●	◔	○

Table 1: Summary of attacker scenarios. Fullness of the circle indicates relative level of knowledge or control.

Using model explanations

SHapley Additive exPlanations (SHAP) – Lundberg et al. 2017

- Model agnostic framework
- Local interpretability
  - Estimate influence of feature-value assignments on model decisions
- Global interpretability
  - Aggregate SHAP values over all the points for each feature
  - Provides intuition on feature importance and direction



Backdoor design strategies

Independent

Independently select high-leverage features and uncommon/weakly-aligned values

- Stronger effect
- Identifiable points

Combined

Greedily select coherent combinations of features and values aligned with target class

- Backdoor points close to real data
- Stealthier

Datasets

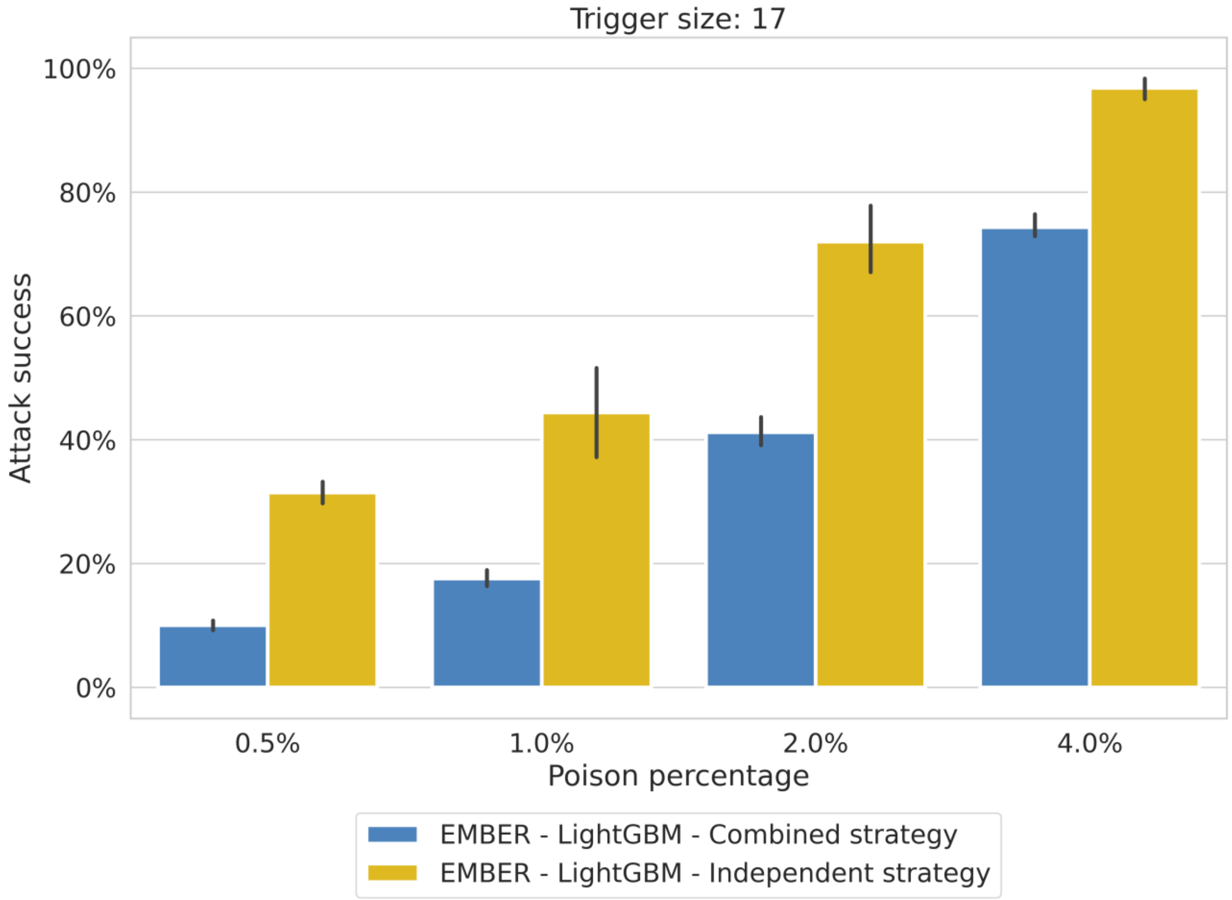
Approach:

- Find subset of modifiable features
- Penalize the selection of infeasible values

Dataset	Size	Type	Models	Approach
EMBER (Anderson et al. 2018)	800k samples 2351 features	Windows PE	LightGBM, DNN	Developed a specific backdooring utility
Drebin (Arp et al. 2014)	128k samples 545k features	Android APK	Linear SVM	Restricted modifications to manifest file
Contagio (Šrndić et al. 2014)	10k samples 135 features	PDF	Random Forest	Restricted modifications as in Šrndić et al. 2014

Experiments

- Significant damage at **1%** poison rate and **17** manipulated features
- Up to ~**80-90%** attack success at **4%** rate
- Minimal side effect on clean data accuracy
- Similar results for the feed forward Neural Network



Experiments

- Drebin:
  - Around **40%** success at **1%** poisoning rate and **30** features
- Contagio:
  - **75%** success at **1%** poisoning rate with **30** features
  - Higher variance due to dataset size

About mitigations

- We adapted different approaches from computer vision:
  - Spectral signatures (Tran et al. 2018)
  - Activation clustering (Chen et al. 2018)
  - Isolation Forests (Liu et al. 2008)
- No tested defense found all backdoors consistently
- Backdoors generated by the combined strategy are hard to identify

Conclusions

- Benign binaries can be used as carriers for poisoning attacks
- Model interpretability methods can be leveraged to guide the backdoor generation
- This approach is model-agnostic and applies to multiple data modalities
- An adversary can generate stealthy backdoors

Thank you!

https://github.com/ClonedOne/MalwareBackdoors

Some references

- Liu, Fei Tony, Kai Ming Ting, and Zhi-Hua Zhou. "Isolation forest." IEEE International Conference on Data Mining. 2008.
- Šrndić, Nedim, Laskov Pavel. "Practical evasion of a learning-based classifier: A case study." IEEE symposium on security and privacy. 2014.
- Arp, Daniel, Michael Spreitzenbarth, Malte Hubner, Hugo Gascon, Konrad Rieck. "Drebin: Effective and explainable detection of android malware in your pocket.“ Network and Distributed System Security Symposium. 2014.
- Gu, Tianyu, Brendan Dolan-Gavitt, and Siddharth Garg. "Badnets: Identifying vulnerabilities in the machine learning model supply chain." arXiv. 2017.
- Lundberg, Scott M., and Su-In Lee. "A unified approach to interpreting model predictions." Advances in neural information processing systems. 2017.
- Anderson, Hyrum S., and Phil Roth. "Ember: an open dataset for training static pe malware machine learning models." arXiv. 2018.
- Shafahi, Ali, W. Ronny Huang, Mahyar Najibi, Octavian Suciu, Christoph Studer, Tudor Dumitras, and Tom Goldstein. "Poison frogs! targeted clean-label poisoning attacks on neural networks." Advances in Neural Information Processing Systems. 2018.
- Tran, Brandon, Jerry Li, and Aleksander Madry. "Spectral signatures in backdoor attacks. Neural Information Processing Systems. 2018.
- Chen, Bryant, Wilka Carvalho, Nathalie Baracaldo, Heiko Ludwig, Benjamin Edwards, Taesung Lee, Ian Molloy, and Biplav Srivastava. "Detecting backdoor attacks on deep neural networks by activation clustering." arXiv. 2018.

Icons by Eucalyp, Pixel-perfect, Icongeek26, Kiranshastry, Freepik, and Becris. www.flaticon.com