

Sensor-Based Repackaged Malware Detection

Authors: Boyu Liu, Duanyue Yun, Huiyu Song, Xin Guo, Xiao Ji Mentors: Shirish Singh, Gail Kaiser

Data Science Institute



Introduction

- Repackaged malware poses a real threat to the health of the Android ecosystem.
- There is evidence that zero-permission sensors in phones are used by malware to perform various activities.
- We are interested in whether sensor usage information of an app will help with repackaged malware detection.

Columbia Engineering The Fu Foundation School of Engineering and Applied Science

Extraction





Decompilation

Data Modeling

Androzoo Local Storage APK Extract Ħ APK sensor usage A information Decompiled Sensor usage **Java Files** data Feature Engineering XML dex2jar jd-cli Android Manifest file **JAR File** Build Classifier Extract **Machine Learning** Apktool APK CSV (ML) features Other ML features **Decompilation Pipeline** Information Extraction Data Modeling

Modeling Pipeline

Columbia Engineering The Fu Foundation School of Engineering and Applied Science





Decompilation

Data Modeling

Decompilation

- Download Android Applications (APKs) from AndroZoo:
 - Dataset: 15,297 pairs 0 (2,776 original & 15,297 repackaged apps)
- Decompile APKs into:
 - Java source code Ο
 - Use dex2jar & jd-cli to convert classes.dex
 - AndroidManifest.xml Ο
 - Use Apktool to extract



The Fu Foundation School of Engineering and Applied Science

Columbia Engineering



Information Extraction

- Build search algorithm to find sensor usage information.
- Use VirusTotal to separate data .
- Drop ambiguous apps:
 - Original apps: VirusTotal report > 0
 - Repackaged apps: VirusTotal report < 5
- Keep benign apps and malwares:
 - Original apps: VirusTotal report =0
 - Repackaged apps: VirusTotal report >= 5

COLUMBIA ENGINEERING



Statistical Testing

- From the graph, the percentage of malware using sensors is higher than benign apps, but is the difference *significant*?
- Yes! the statistical test shows that the difference is significant.

p1/p2: proportion of malicious/benign apps using at least one sensor

H0: p1=p2 H1: p1>p2 The test statistic is calculated as follows:

 $z = \frac{\hat{p}_1 - \hat{p}_2}{\sqrt{\hat{p}(1-\hat{p})\left(\frac{1}{n_1} + \frac{1}{n_2}\right)}} = 13.947$ **Reject Ho!**







Data Visualization

Malware and benign apps have different sensor usage patterns!







Data Modeling

Data Visualization



Distribution of Jaccard similarity score



Columbia Engineering The Fu Foundation School of Engineering and Applied Science

Extraction





Decompilation

Data Modeling

Data Modeling

- Features: drop zero variance features
 - 10 sensor-related features Ο
 - 323 other model interpretable features Ο
- **Evaluation metric:**
 - Imbalanced dataset (80% are malware) Ο
 - Use balanced accuracy Ο
- Models experimented includes Logistic Regression • (LR), K Nearest Neighbors (KNN), Support Vector Machine (SVM), XGBoost (XGB), Deep Neural *Network* (DNN)



COLUMBIA ENGINEERING The Fu Foundation School of Engineering and Applied Science





Feature Engineering

Build Classifier

Data Modeling

		-		Decompilation	Informatio Extraction
	Rest	ılts			
	Model	Balanced Accuracy (with sensor features)	Balanced Accuracy (without sensor features)		(
	LR	0.855	0.842	CSV Sensor usage data	Er
-	KNN	0.851	0.846	CSV Other ML features	→
	SVM	0.883	0.879		
	XGB	0.870	0.863		\
	DNN	0.870	0.872		

Decompilation Information Extraction Modeling



Findings

- Including sensor-related features leads to slight improvement across all models except for DNN.
- XGBoost has the highest AUC among all models.

COLUMBIA ENGINEERING The Fu Foundation School of Engineering and Applied Science

COLUMBIA ENGINEERING The Fu Foundation School of Engineering and Applied Science



Findings

- For our data modeling, we used a lower threshold of o and upper threshold of 5. Therefore, an app with a VirusTotal score of o is labelled as benign and an app with score >=5 is labelled as malware.
- The thresholds used to determine whether an app is malware or not might have an impact on performance.





Thank you!

Gather Location: 37