32ND USENIX SECURITY SYMPOSIUM

Black-box Adversarial Example Attack towards FCG Based Android Malware Detection under Incomplete Feature Information

Heng Li[†], Zhang Cheng[‡,†], Bang Wu[†], Liheng Yuan[†], Cuiying Gao[†], Wei Yuan[†] *, Xiapu Luo[⋆]

[†] *Huazhong University of Science and Technology*

[⋆] *The Hong Kong Polytechnic University*

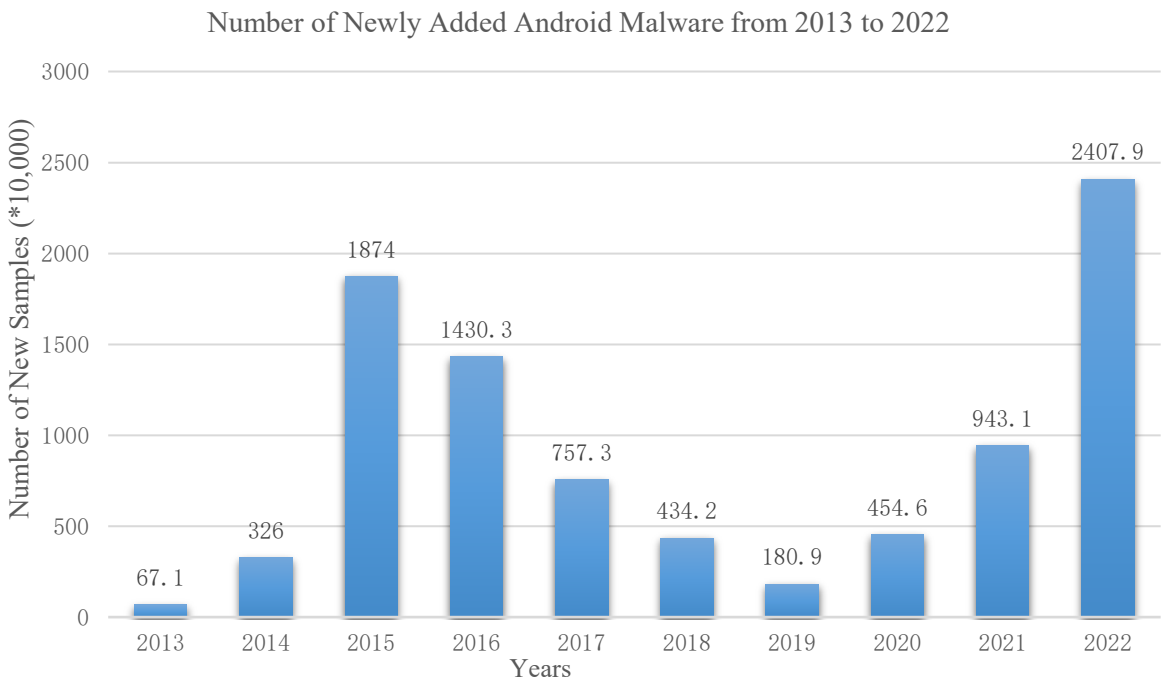
[‡] *NSFOCUS Technologies Group Co., Ltd.*

{liheng,wubangm,ylh,gaocy,yuanwei}@hust.edu.cn

chengzhang@nsfocus.com,csxluo@comp.polyu.edu.hk

Introduction

- Android Malware



1.Data Theft: Android malware can compromise sensitive user data, including personal information, login credentials, financial details, and private communications.

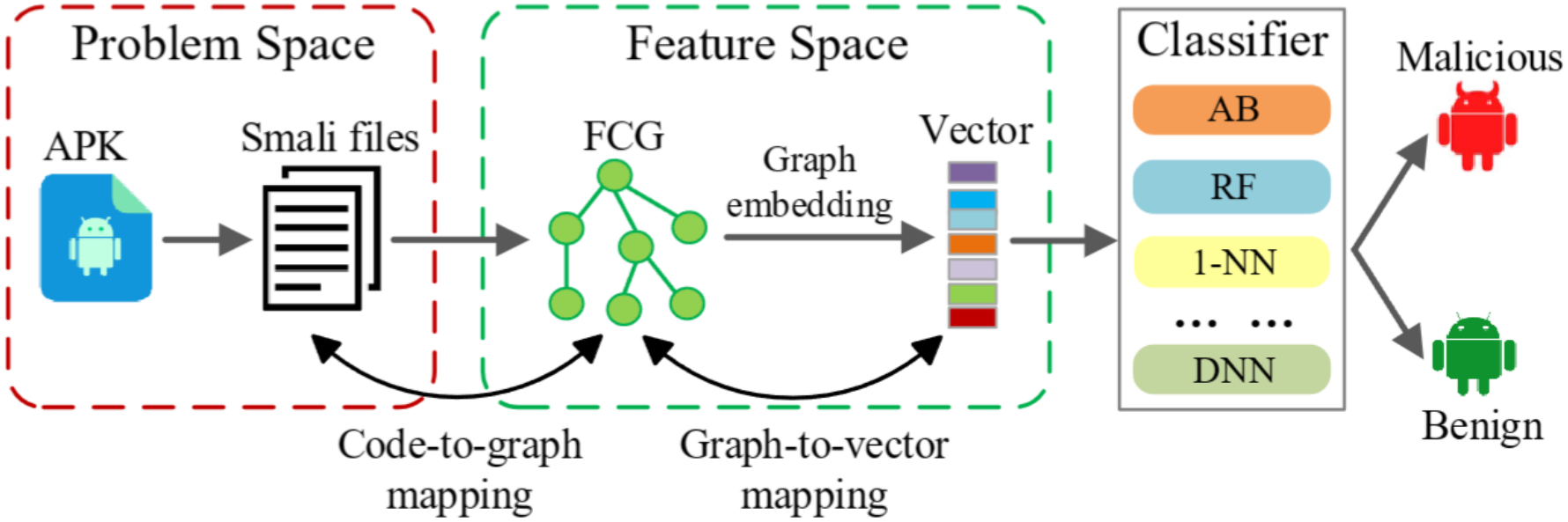
2.Financial Loss: Malicious software can initiate unauthorized financial transactions, leading to monetary losses for the victims.

3.Privacy Invasion: Android malware may enable unauthorized access to the device's camera, microphone, and other sensors, violating the user's privacy.

. . .

Introduction

- FCG based Android Malware Detection

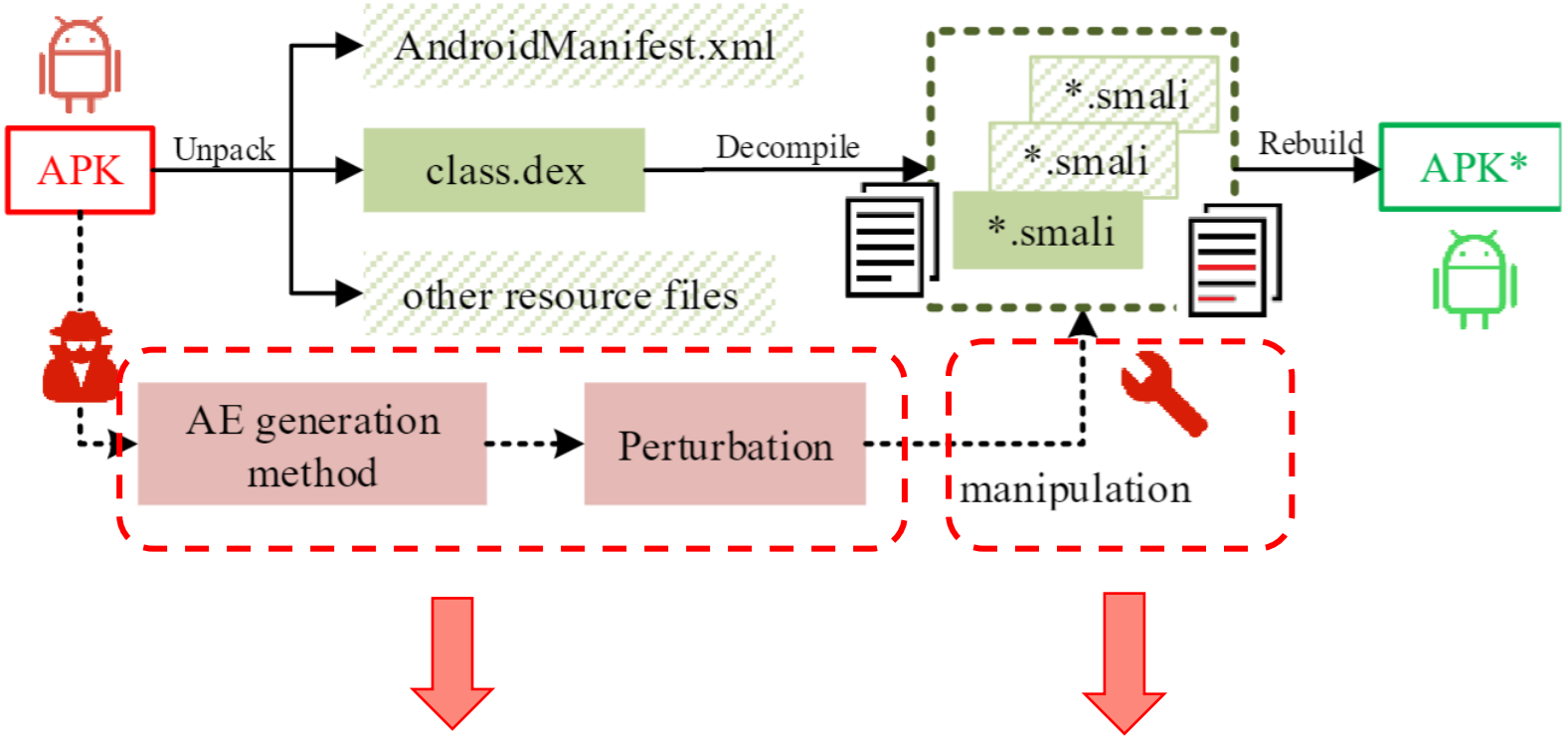


Problem Space: The real input-space objects space (e.g., APK File).

Feature Space: The feature data extracted from the problem space. (e.g., FCG feature)

Introduction

- Adversarial Example Attack towards FCG Based AMD



Attacker Knowledge

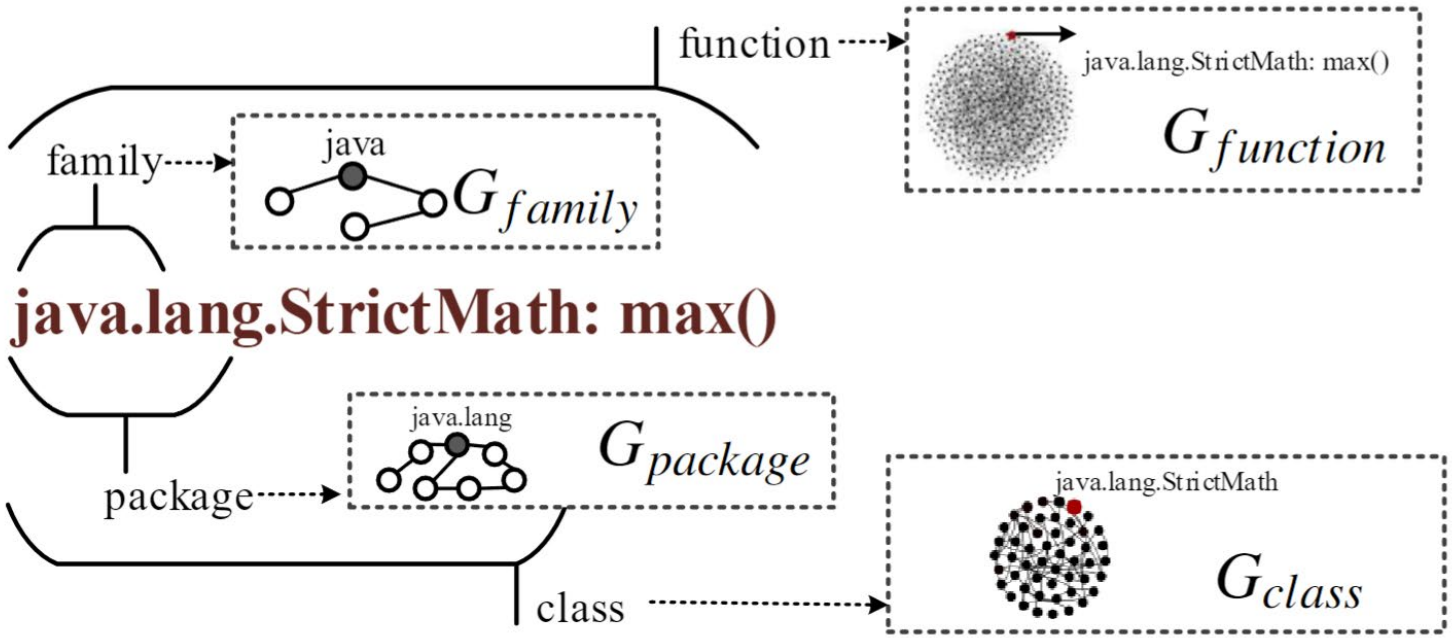
Feature granularity	✗
Classification method	✗
Model parameters and structure	✗
Output probabilities	✗

FCG baesd AMD system	✓
Output results (binary)	✓

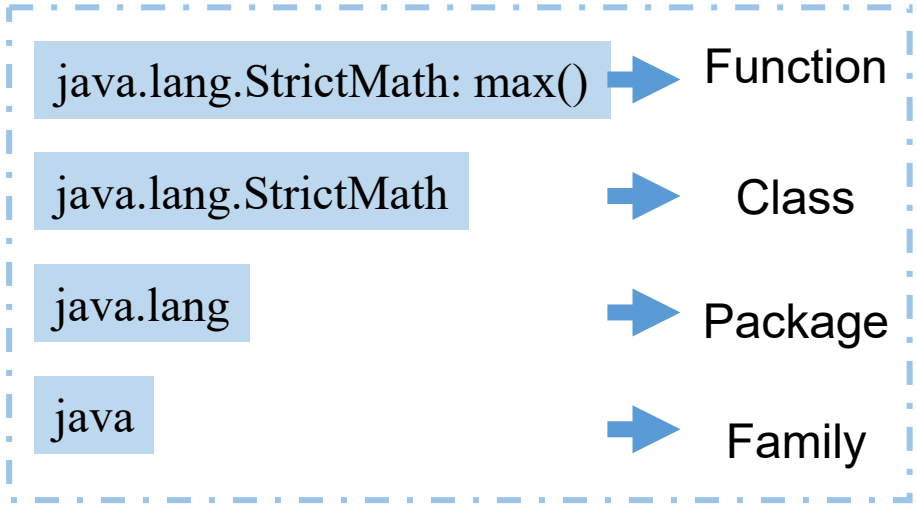
Introduction

- FCG based Android Malware Detection

Different granularities of the FCG



An Example



METHOD

➢ Malware Manipulation

Requirements

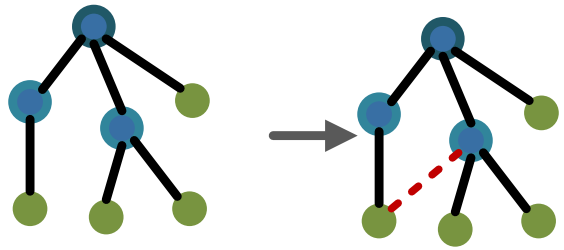
R1:Functional Consistency.

R2:All-granularities influence.

R3:Resilience to static analysis.

R4:Non-stationary perturbation.

Manipulation process



Adding a function call between a caller and a callee

Malware Manipulation

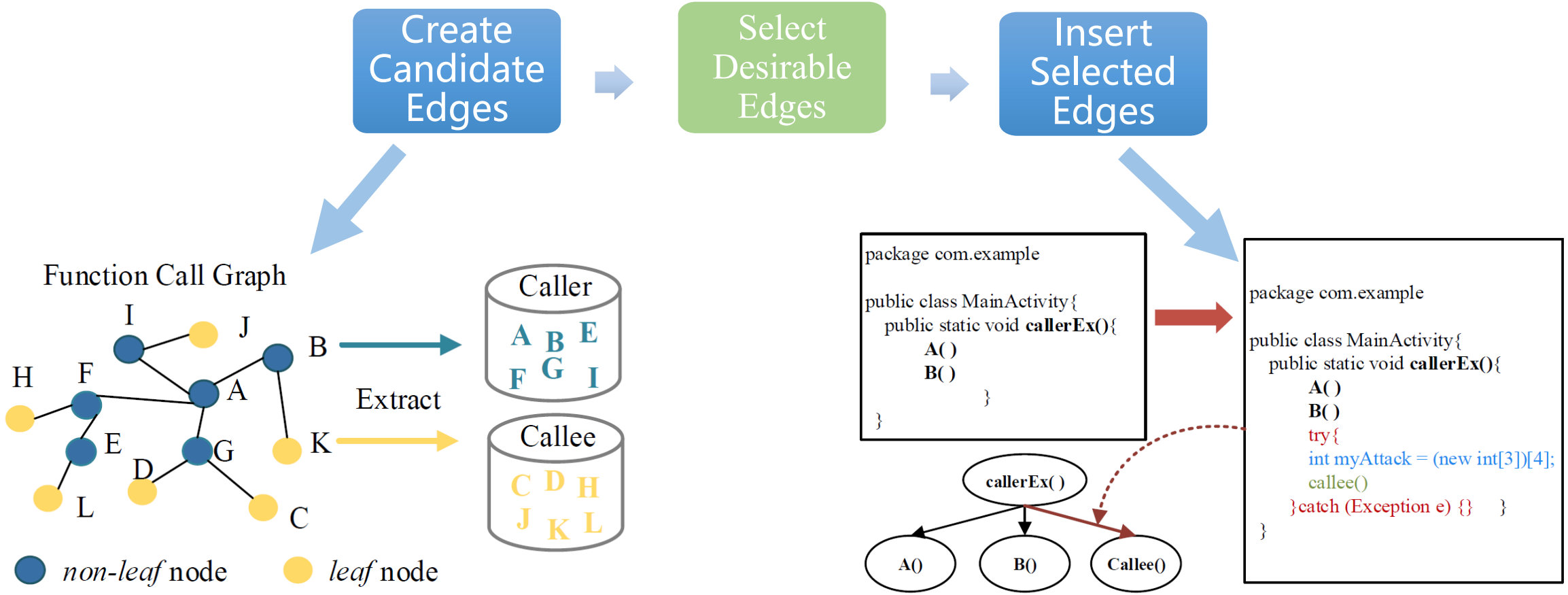
Adversarial Perturbation Generation

Create Candidate Edges → Select Desirable Edges → Insert Selected Edges

Malware Manipulation

METHOD

- Malware Manipulation

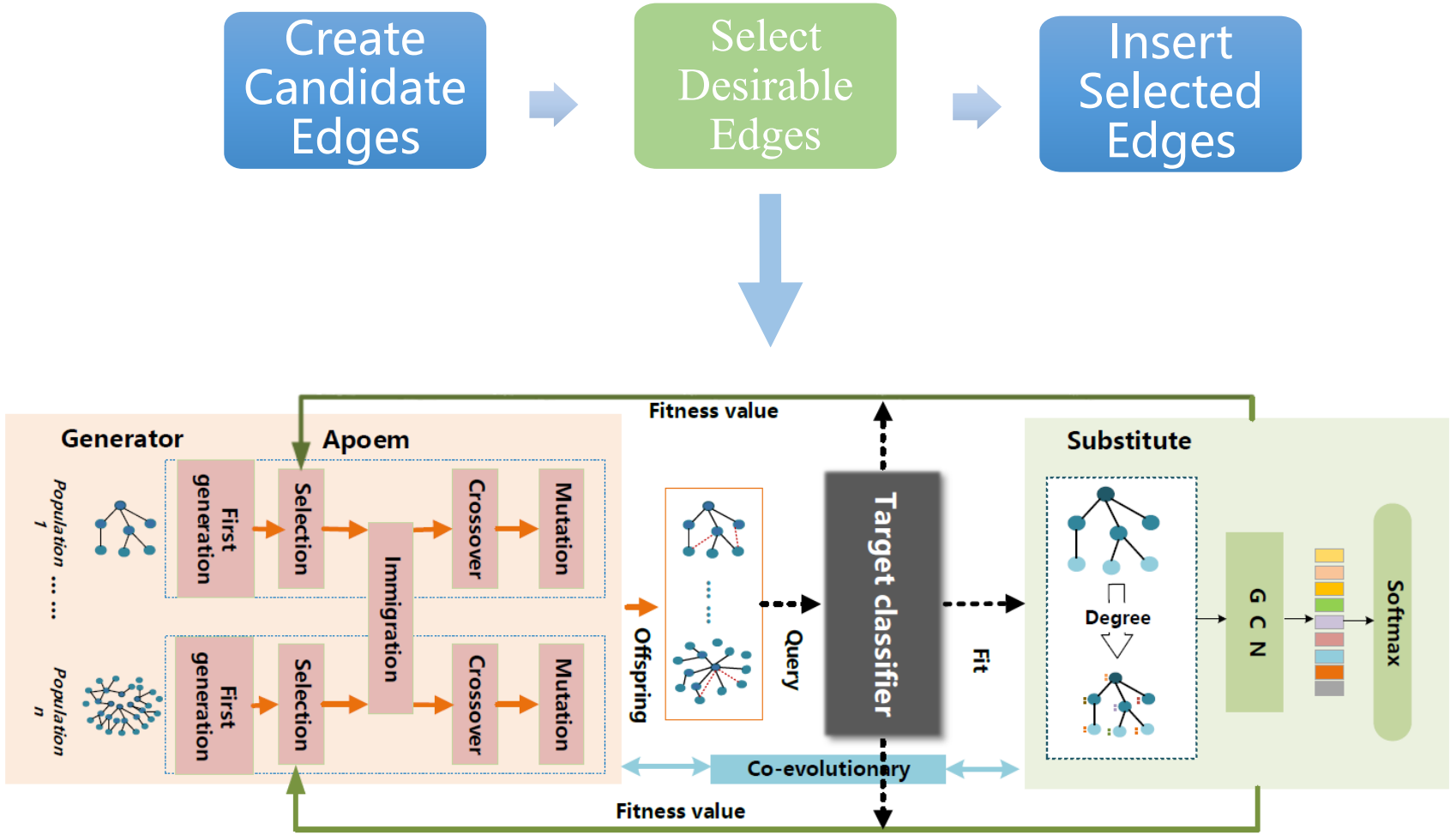


Selecting callers and callees from an FCG

An example of try-catch trap

METHOD

- Adversarial Perturbation Generation



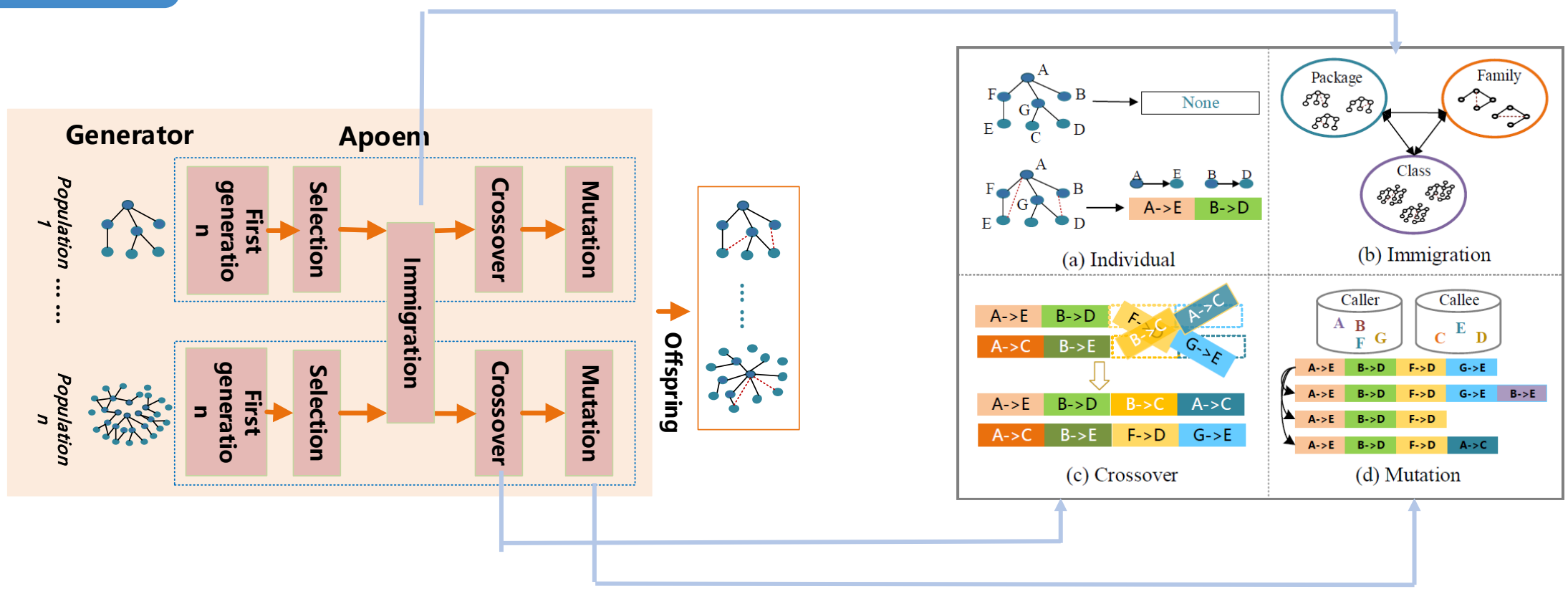
METHOD

➢ Adversarial Perturbation Generation

Generator

Adversarial multi-population co-evolution algorithm

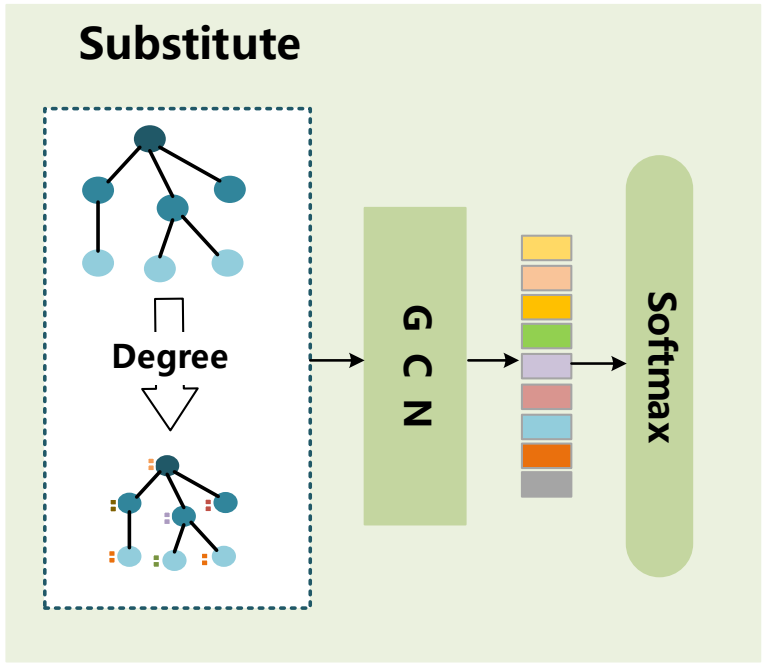
Multiple populations cooperatively evolve



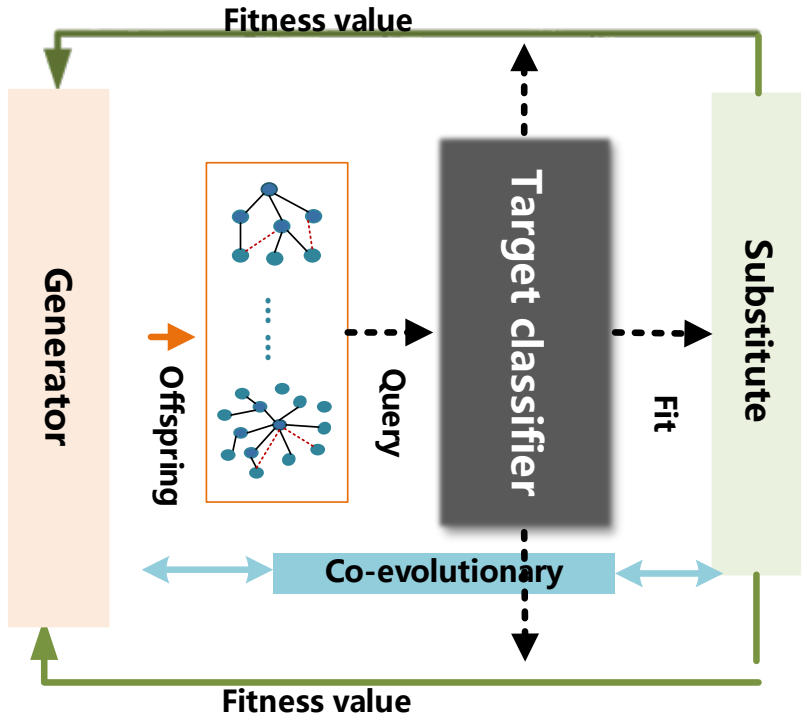
METHOD

- Adversarial Perturbation Generation

Discriminator/ Substitute



The training between the D and G



EXPERIMENTS

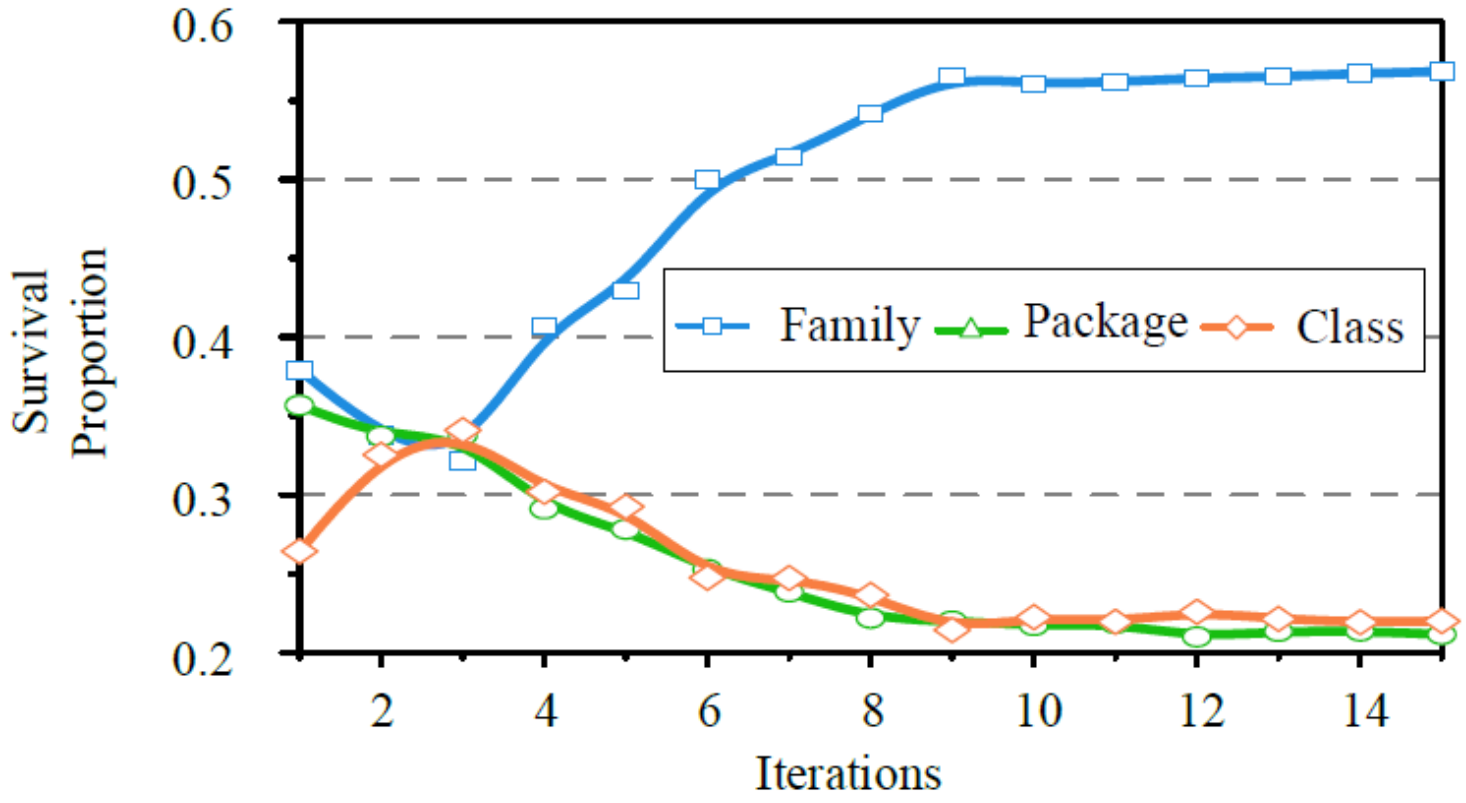
- Effectiveness

Classifier\Level		Family			Package			Class		
		ASR	APR	IR	ASR	APR	IR	ASR	APR	IR
MaMaDroid	RF	1.000	0.021	8.670	1.000	0.049	13.640	1.000	0.083	12.490
	DNN	0.990	0.149	11.130	1.000	0.134	16.730	1.000	0.153	15.907
	AB	1.000	0.066	10.270	1.000	0.072	14.300	1.000	0.118	15.460
	1-NN	1.000	0.031	7.000	1.000	0.109	11.630	1.000	0.060	10.960
	3-NN	1.000	0.037	9.390	1.000	0.142	13.380	1.000	0.072	10.770
APIGraph	RF	1.000	0.039	11.260	1.000	0.098	14.930	1.000	0.040	9.530
	DNN	1.000	0.132	14.370	1.000	0.096	18.630	1.000	0.168	12.566
	AB	1.000	0.093	14.510	0.990	0.131	18.350	1.000	0.067	12.250
	1-NN	1.000	0.058	11.190	1.000	0.089	14.040	1.000	0.012	6.910
	3-NN	1.000	0.085	11.570	1.000	0.105	13.770	1.000	0.019	7.780
GCN	DNN	1.000	0.205	11.610	1.000	0.104	17.320	-	-	-

Effectiveness of BagAmmo towards MaMaDroid, APIGraph and GCN.

EXPERIMENTS

- Evolution



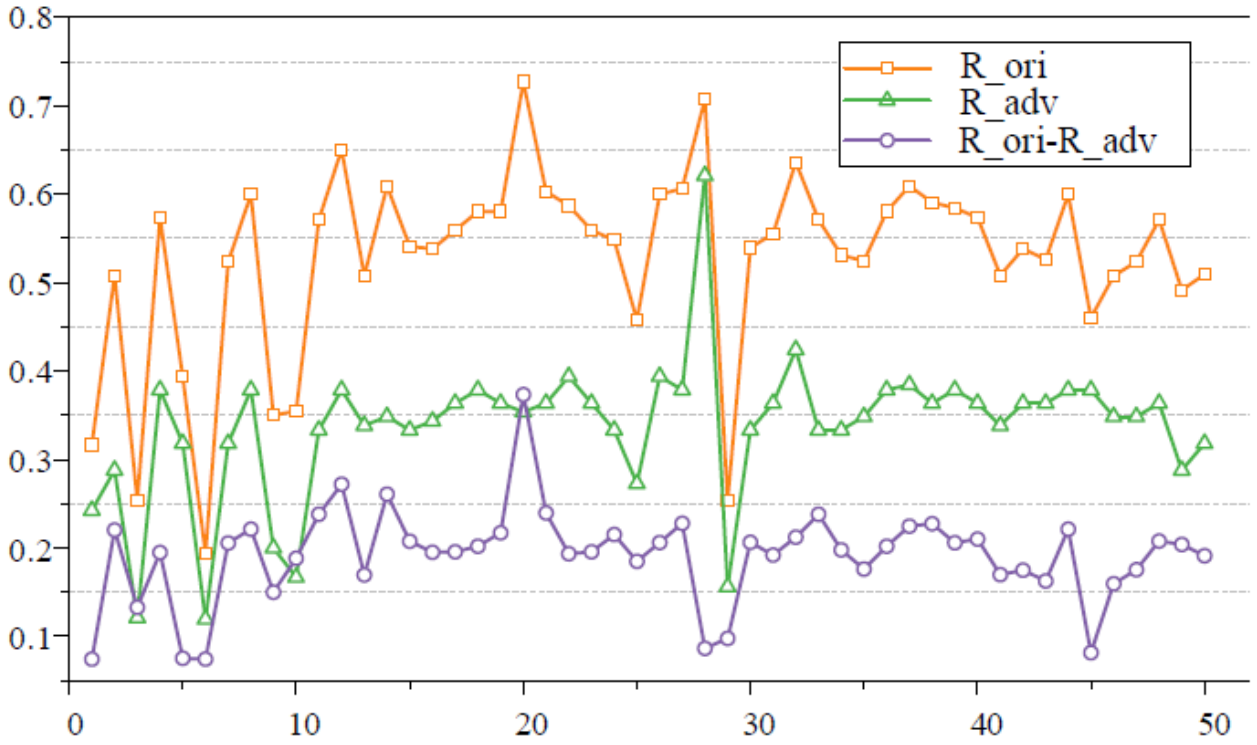
The changing trend of survival proportion

When attack a family level FCG android malware detection. As the number of queries increases, the package population and class population gradually fall to a low level.

Contrarily, the survival proportion of the population corresponding to the correct feature granularity (i.e., family level) gradually rises to a high level.

EXPERIMENTS

- Attack Performance on Virustotal



The detection success ratio on VirusTotal.

Our method can effectively reduce the probability of malware being detected by VirusTotal

32ND USENIX SECURITY SYMPOSIUM

Thank you for Listening!