Malware Variants Identification in Practice

Marcus Botacin[1], André Grégio[1], Paulo Lício de Geus[2]

[1]Federal University of Paraná (UFPR) – {mfbotacin, gregio}@inf.ufpr.br

[2]University of Campinas (Unicamp) – paulo@lasca.ic.unicamp.br

SBSEG 2019

Agenda

Agenda

Malware at Scale!

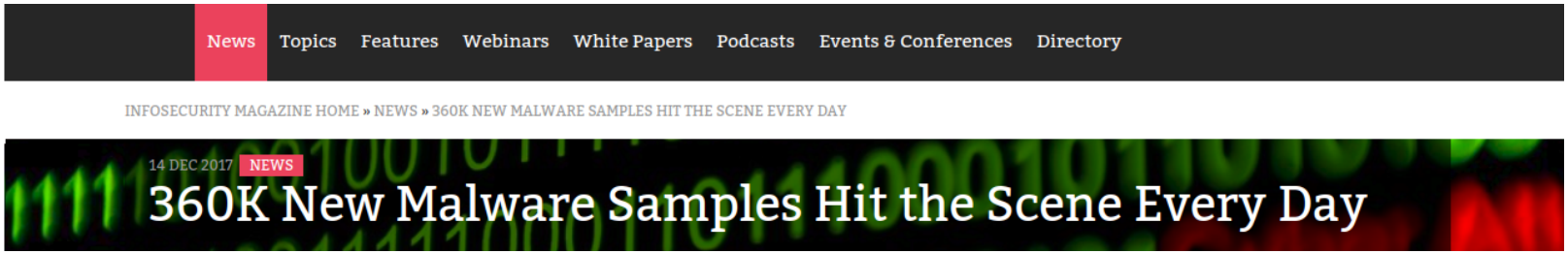

Figure: **Source:** `https://www.infosecurity-magazine.com/news/360k-new-malware-samples-every-day/`

Figure: **Source:** `https://money.cnn.com/2015/04/14/technology/security/cyber-attack-hacks-security/index.html`

Current Approaches.

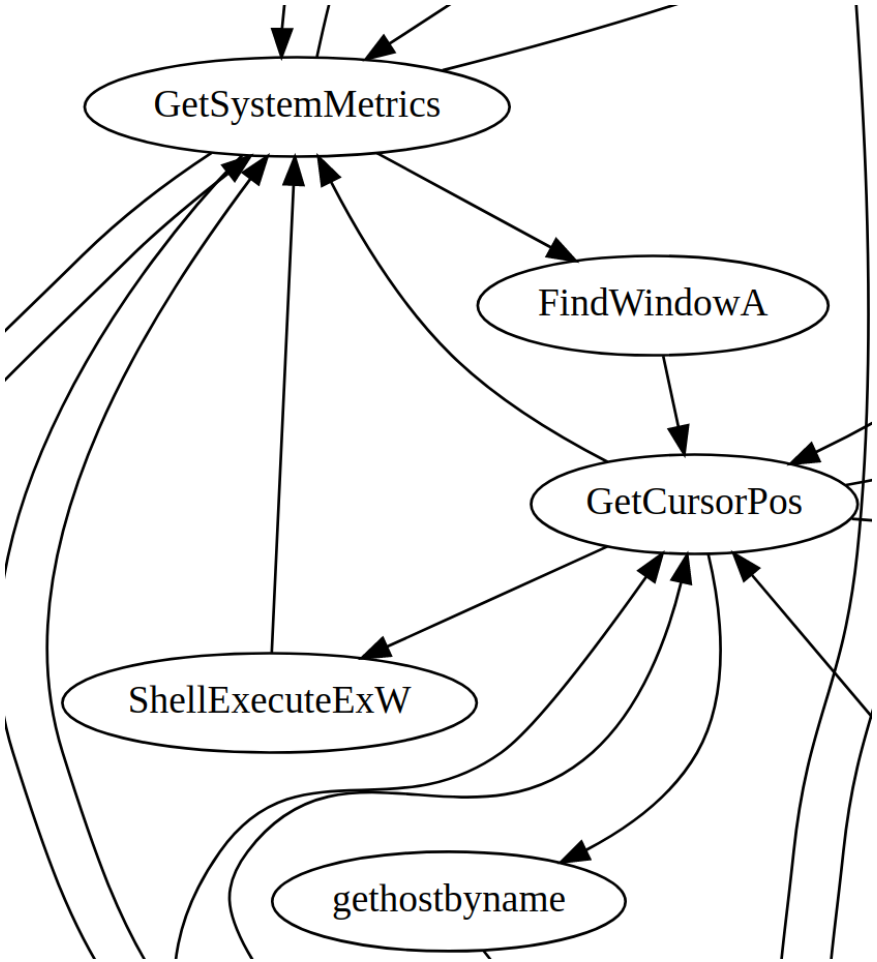

Figure: Function-based, Graph Modeling.

Agenda

Same-Behavior Function Replacement

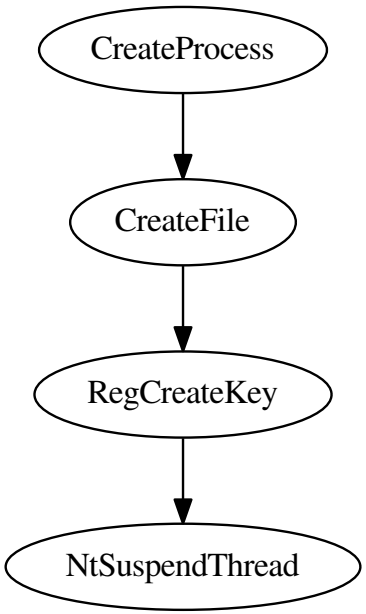

Figure: Original sample's CG.

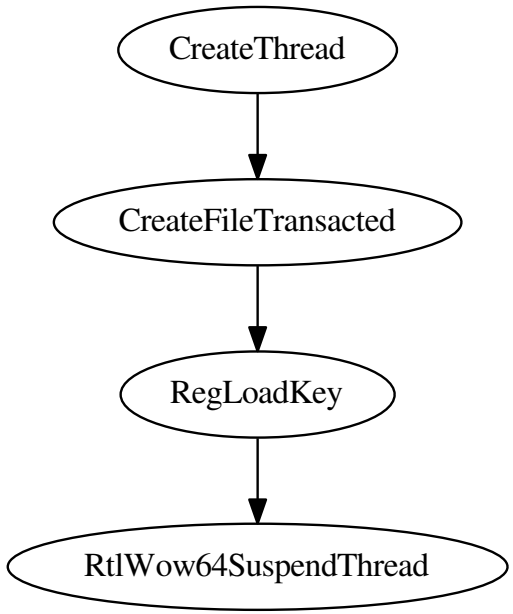

Figure: Variant sample's CG.

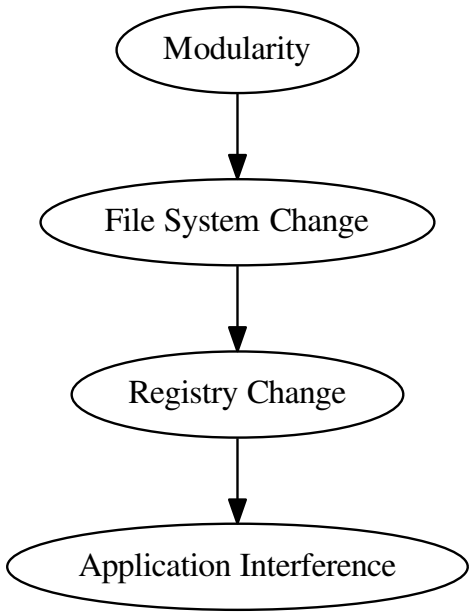

Figure: Behavioral graph from both samples.

Behavioral Classification

- Compression
- Cryptography.
- Debug.
- Delay.
- Environment.
- Escalation.
- Exfiltration.
- Fingerprint.
- File System.
- Interference.
- Internet.
- Modularity.
- Monitoring.
- Registry.
- Evidence Removal.
- Side Effects.
- System Changes.
- Target Information.
- Timing Attacks.

Behavior-based Graph.

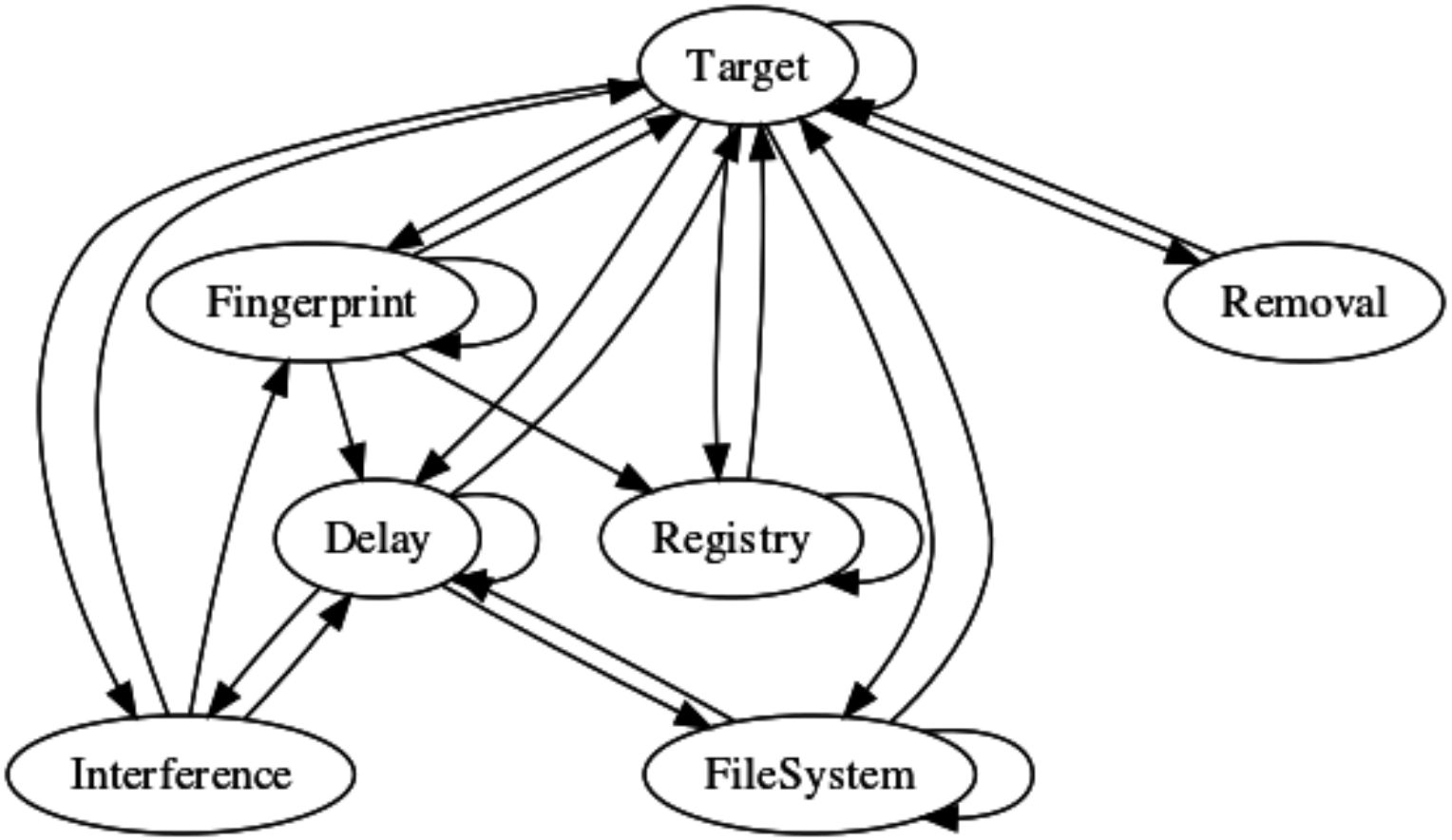

Figure: Behavior-based graph for a given sample.

Agenda

Malware Embedding

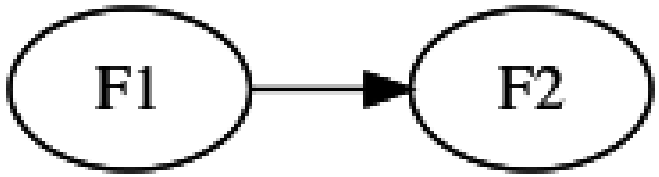

Figure: **Sample 1**. The original sample.

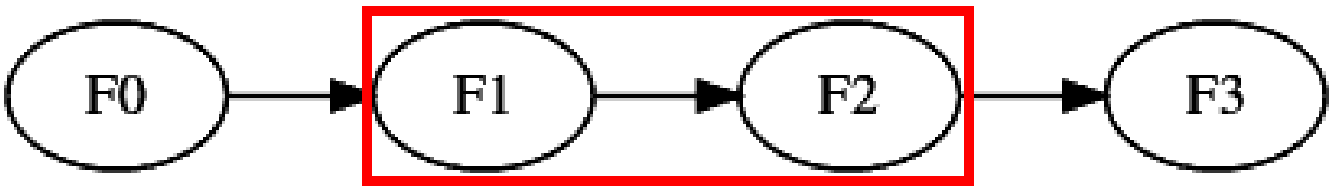

Figure: **Sample 2**. Variant sample embedding the original one.

Matching Metrics

Definition

The similarity of two malware, represented as sets, A and B, of vertices or edges of two graphs, is defined as:

$$Sim(A, B) = \frac{|A \cap B|}{|A \cup B|} \quad (1)$$

Definition

The similarity of two malware, represented as sets, A and B, of vertices or edges of two graphs, is defined as:

$$Sim(A, B) = \max\left(\frac{|A \cap B|}{|B|}, \frac{|B \cap A|}{|A|}\right) \quad (2)$$

Continence Results

Table: Continence of Sample 1 in Sample 2.

CG	A	B	C
I	**0.66**	0.52	**0.64**
J	**0.75**	0.49	**0.50**
K	0.42	**0.80**	**0.44**

Table: Continence of Sample 2 in Sample 1.

CG	A	B	C
I	0.57	**0.56**	0.43
J	0.33	**0.51**	0.44
K	**0.76**	0.65	**0.44**

Table: Maximum continence of Sample 1 and Sample 2.

CG	A	B	C
I	0.66	0.56	0.64
J	0.75	0.51	0.50
K	0.76	0.80	0.44

Agenda

The Whitelisting Effect.

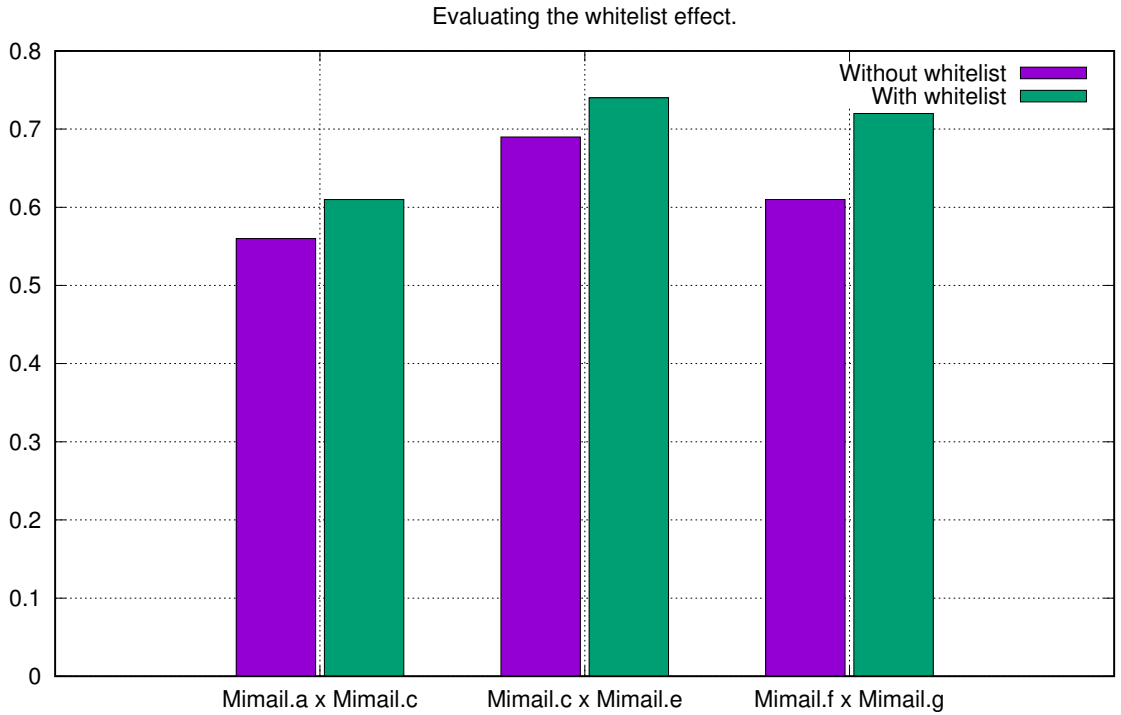

Figure: Evaluating whitelisting effect. Similarity scores are higher when using the whitelist-based approach.

Advantages of the Behavioral model.

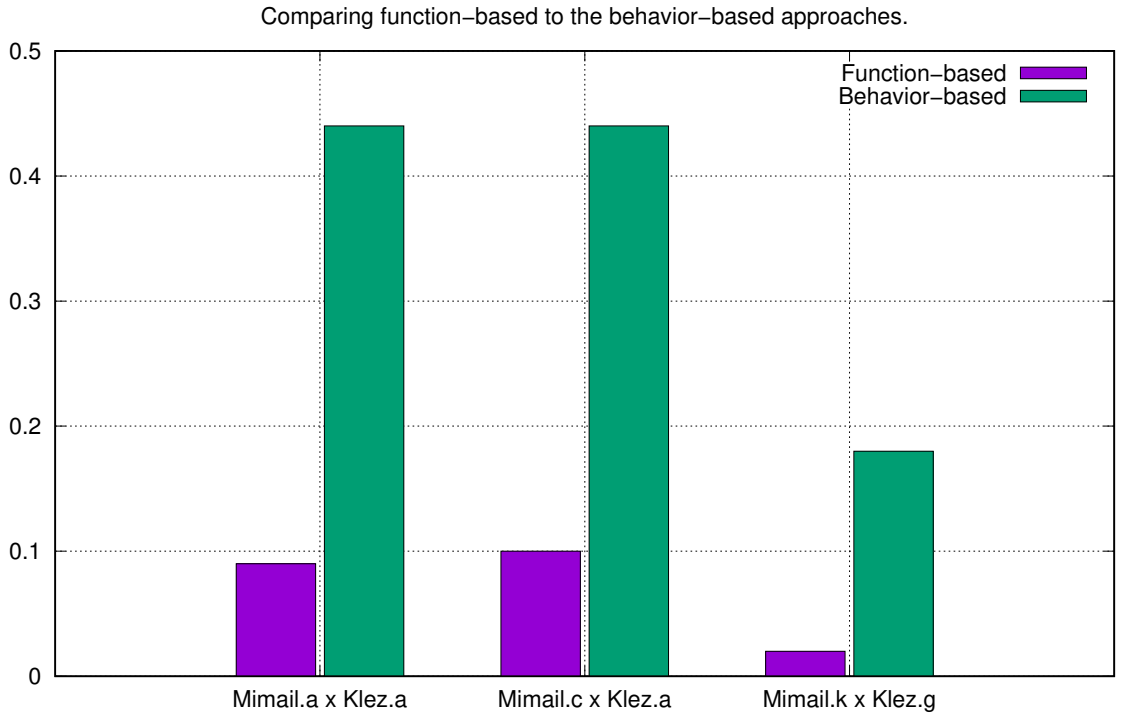

Figure: Function vs. Behavior-based approaches. Scores are higher when considering behavioral patterns.

Evaluating Metrics.

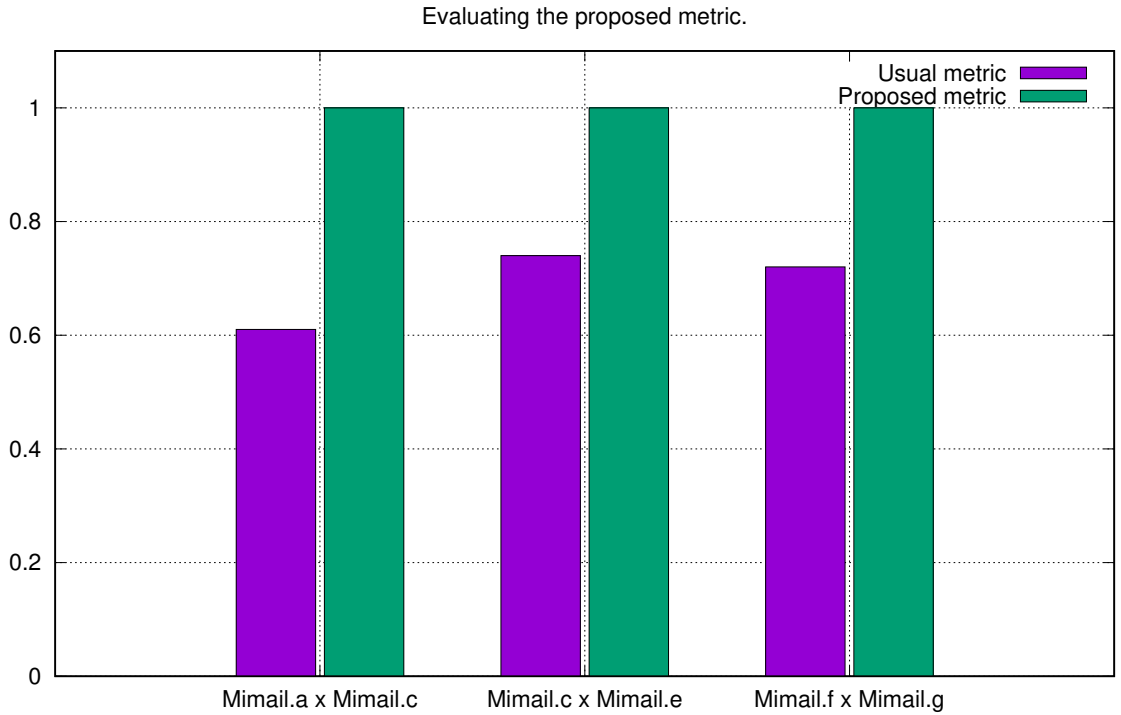

Figure: Proposed metric. Scores are higher when using it in comparison to the usual one.

Solutions Comparison.

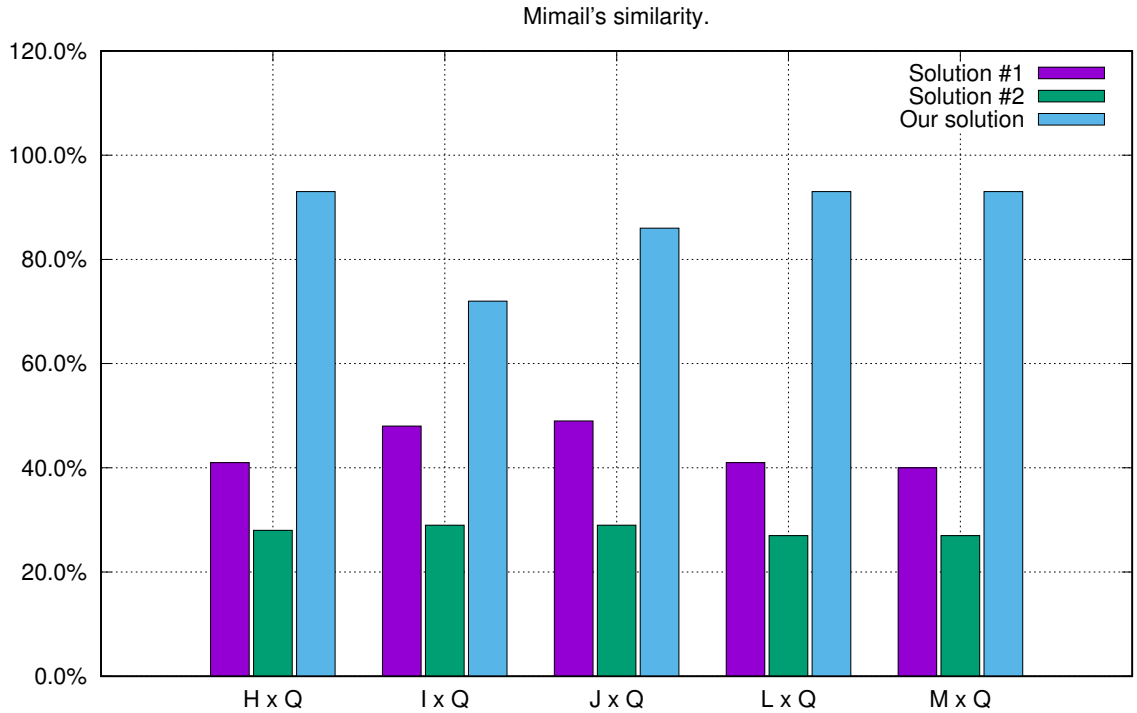

Figure: Mimail's sample similarity. Our solution's scores are higher when compared to other ones.

Domain Transformation and Similarity Measures.

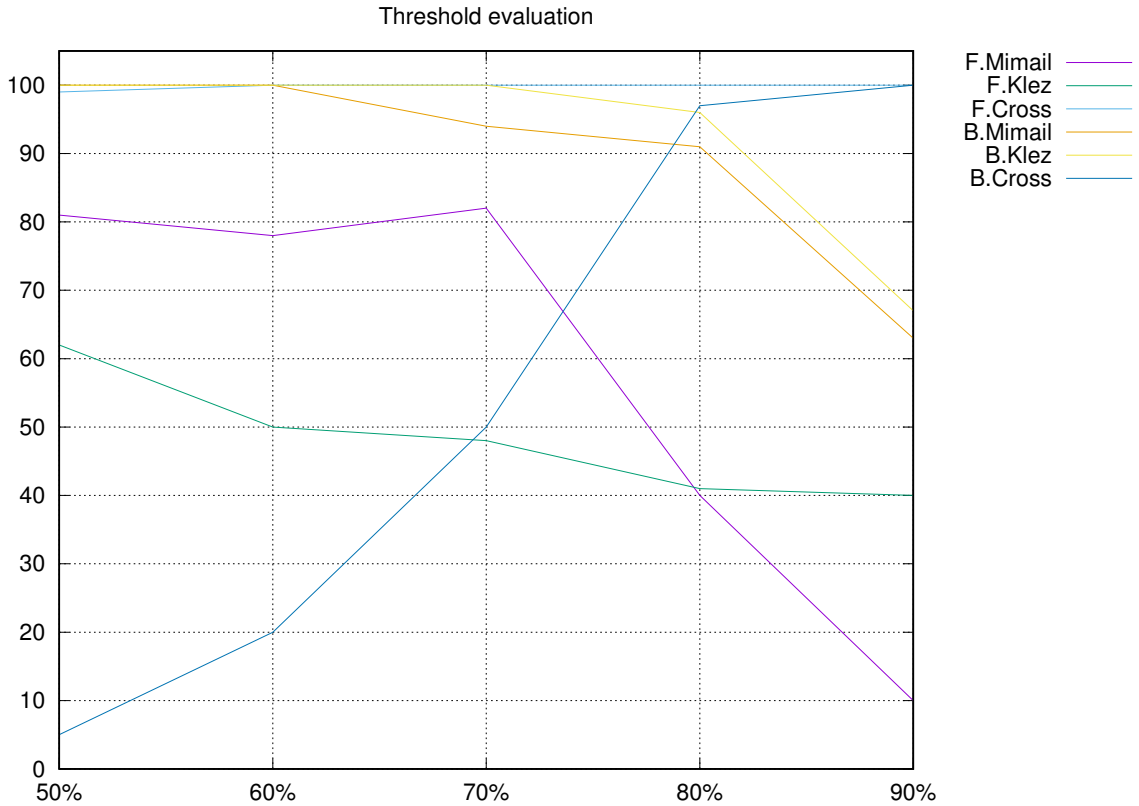

Figure: Threshold evaluation. This should be higher than 80% in order to proper label the `cross` dataset.

Real World Experiments (1/2)

Table: Identified variants among unknown, wild-collected samples.

Family	Sample	Hash	Label
1	A	c2ef1aabb15c979e932f5ea1d214cbeb	Generic_vb.OBY
	B	747b9fe5819a76529abc161bb449b8eb	Generic_vb.OBO
	C	39a04a11234d931bfa60d68ba8df9021	Generic_vb.OBL
2	A	96d13246971e4368b9ed90c6f996a884	Atros4.CENI
	B	e23588078ba6a5f5ca1c961a8336ec08	Atros4.CENI
	C	31a2b6adc781328cb1d77e5debb318ff	Atros4.CENI

Real World Experiments (2/2)

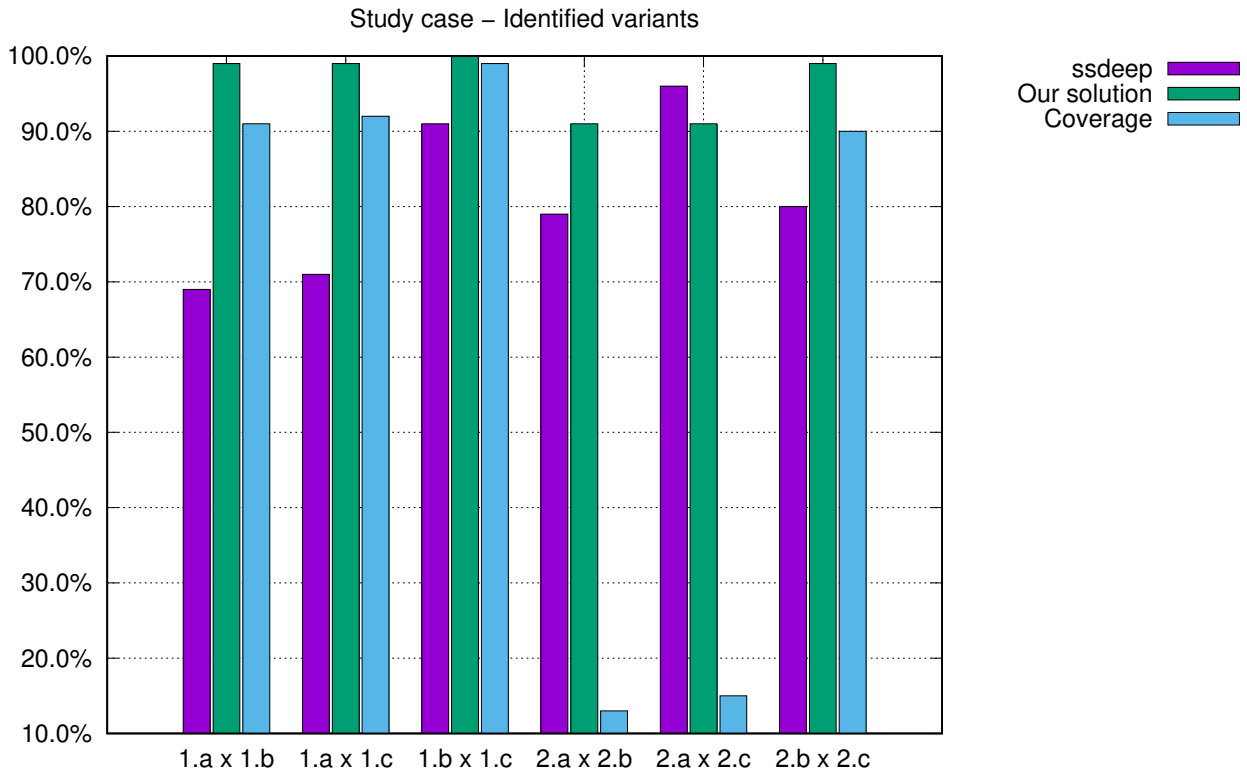

Figure: Study case: variant identification. Our approach outperforms others even on low coverage scenarios.

Agenda

Matching Complex Behaviors is Challenging!

Figure: DLL injection functions among other function calls.

Figure: Proposed DLL injection class.

Agenda

Conclusions

Challenges & Lessons

- Anti-disassembly breaks CG extraction.
- Transparent, dynamic tracing is a viable alternative.
- Same-Function Replacement breaks malware clustering.
- Behavior-based clustering is a viable alternative.
- Dead code breaks similarity metrics.
- Continence metric is a viable alternative.

Questions & Comments.

Contact

- **mfbotacin@inf.ufpr.br**

Additional Material

- `https://github.com/marcusbotacin/Malware.Variants`