Challenging Anti-virus through Evolutionary Malware Obfuscation

Marco Gaudesi Andrea Marcelli Ernesto Sanchez Giovanni Squillero Alberto Tonda

Goal

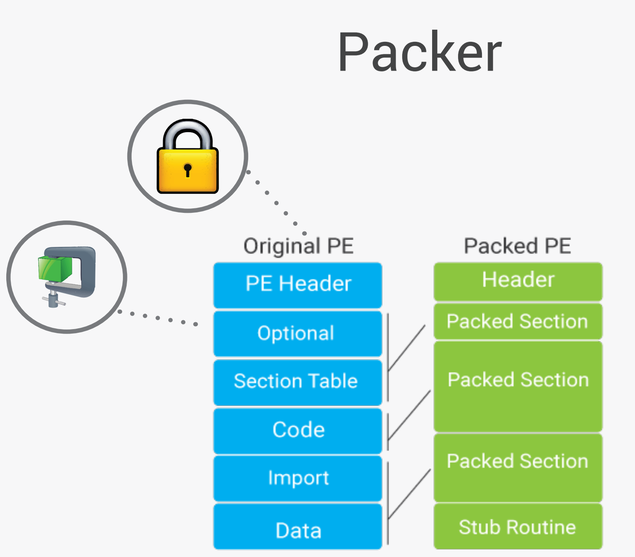
Develop a new obfuscation mechanism based on evolutionary algorithms.

It can be used by security industries to stress the analysis methodologies and to test the ability to react to malware mutations.

Malware

/ malicious software /

- hides as long as possible
- communicates
- executes the payload
- propagates



A packer compresses or encrypts the instructions and data of a program generating a new executable version. At run time, the new executable decompress the original program in memory, and then jump into it.

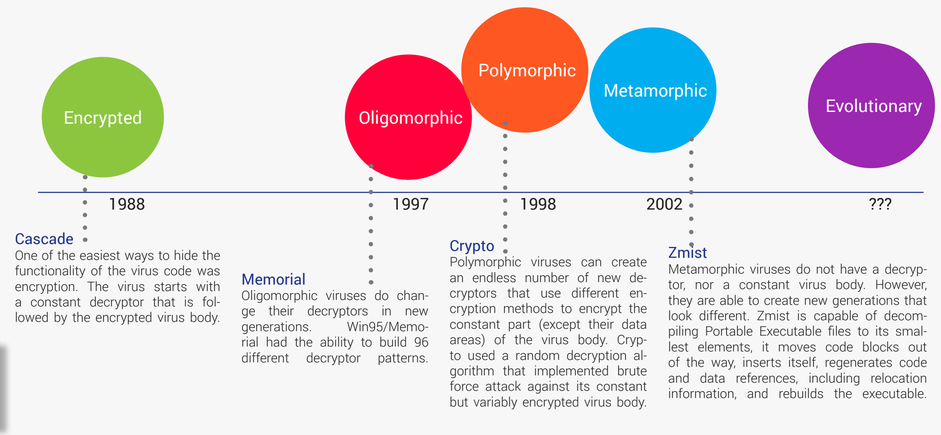
Packers have been originally designed to save disk space.
Then they have been introduced in the word of malicious software: the code must be decrypted before static analysis can be applied.
Moreover changing the encryption key produces a completely different executable.

The unpacking stub:

1) It decompresses and decrypts the original code.

2) It resolves the imports of the executable: if the import table is packed, the loader cannot resolve the imports and load the corresponding DLLs.

3) It transfers back the control to the Original Entry Point (OEP).

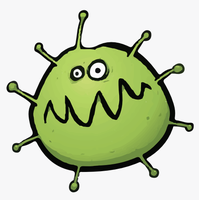


The idea of genetic selection for behaviours was first seen in 2002.
W32/Smile

Polymorphism using genetic algorithms was first seen in 2005.
W32/Zellome

**The malware uses an evolutionary algorithm to generate completely new obfuscating algorithms.
The individuals are a set of working packers and the 'fitness' is how similar the new executable is to the original one.**

Generating the code



1. **Generate an opcode sequence**
   Randomly-generated, variable-length sequence of x86 assembler instructions.

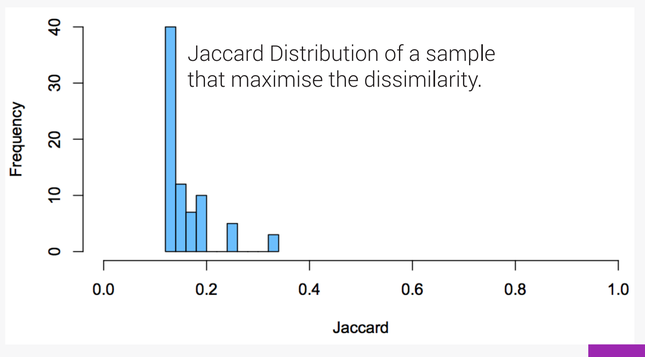
5E	pop esi
56	push esi
311E	xor [esi],ebx
01C3	add ebx,eax
85C0	test eax,eax
75F7	jnz 0x5
C3	ret

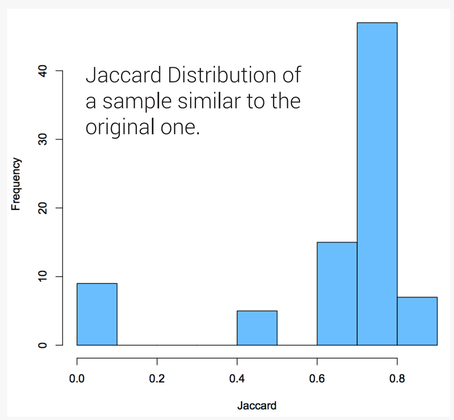
2. **Test the sequence. Is it reversible?**
   Encoding and decoding routines are applied subsequently to sequence of bytes.
3. **Fitness evalutation with the Jaccard Index**
4. **Code encryption**
   Creation of a new packer variant.
5. **Reproduction**
   The decoding routine is embedded in the new executable. At run time it will restore the original program in memory.

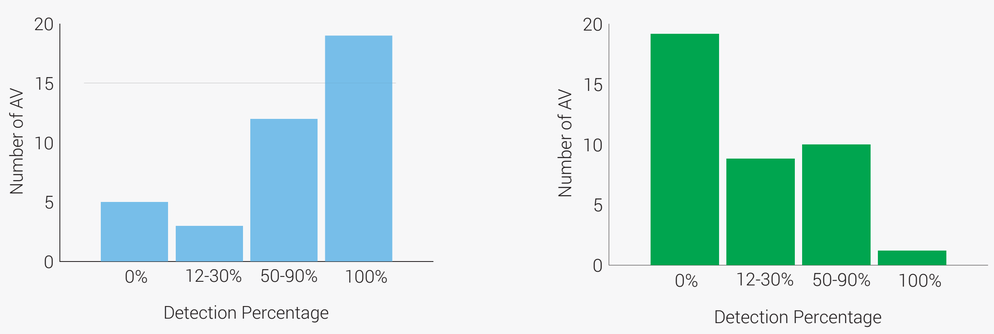
Jaccard Index

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|}$$

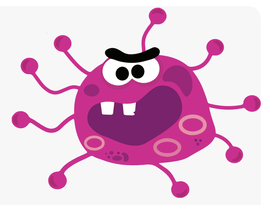
It is used to evaluate the similarity between a Malware sample and the original one.







Experimental Evaluation



8 recent malware samples for Windows 32 bit

High initial detection rate
+
Executable behavior susceptible to heuristic evaluation

virustotal — 57 AV engines

OPSWAT. Metascan® — 44 AV engines

Further evaluation with locally installed AVs

Future Work

IoT Worm

The diffusion of Internet of the Things devices, strongly network oriented, which often lack of proper security measures represents the perfect environment where a new evolutionary worm, platform indipendent, can spread.

Malware Detection Through Machine Learning

With over 1 million malware samples caught every day in honeypots all over the world, new detection approaches are necessary.
The research aims at developing a detection mechanism based on multiple classifier where each one targets a particular malware family.

Try the Evolutionary Obfuscator against advanced Anti-Virus based on Deep Neural Network.

INRA SCIENCE & IMPACT

POLITECNICO DI TORINO 1859 1906

evo* 2016