On deobfuscation in practice

Vasily Bukasov

Dmitry Schelkunov

Obfuscation applications

- Software protection against computer piracy

- Malware protection against automatic detection and to impede analysis of a malicious code

Obfuscators and protectors

- Manual obfuscation requires a lot of resources

- It’s much easier to use obfuscators and protectors which promise a strong obfuscation

Common code protection techniques

- Code encryption (out of scope of our report)
- Code virtualization
- Code morphing

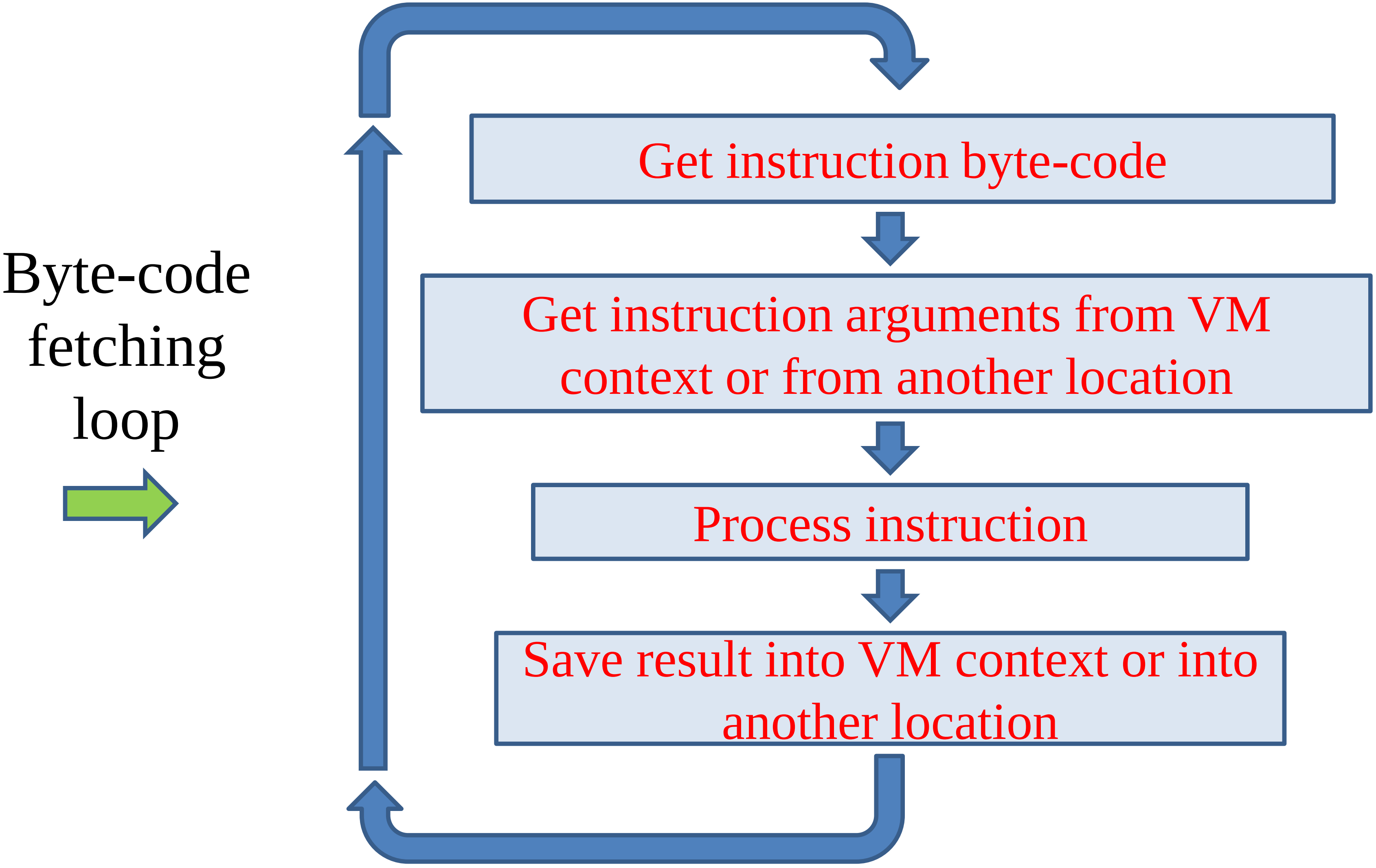
Code virtualization

- Converts a source assembler code to the specially generated byte-code
- Inserts byte-code and byte-code interpreter into the source PE file

Code virtualization

Byte-code mostly represents original assembler instructions so its execution has the same effect as from the original instructions

Code virtualization



Byte-code fetching loop

Get instruction byte-code

Get instruction arguments from VM context or from another location

Process instruction

Save result into VM context or into another location

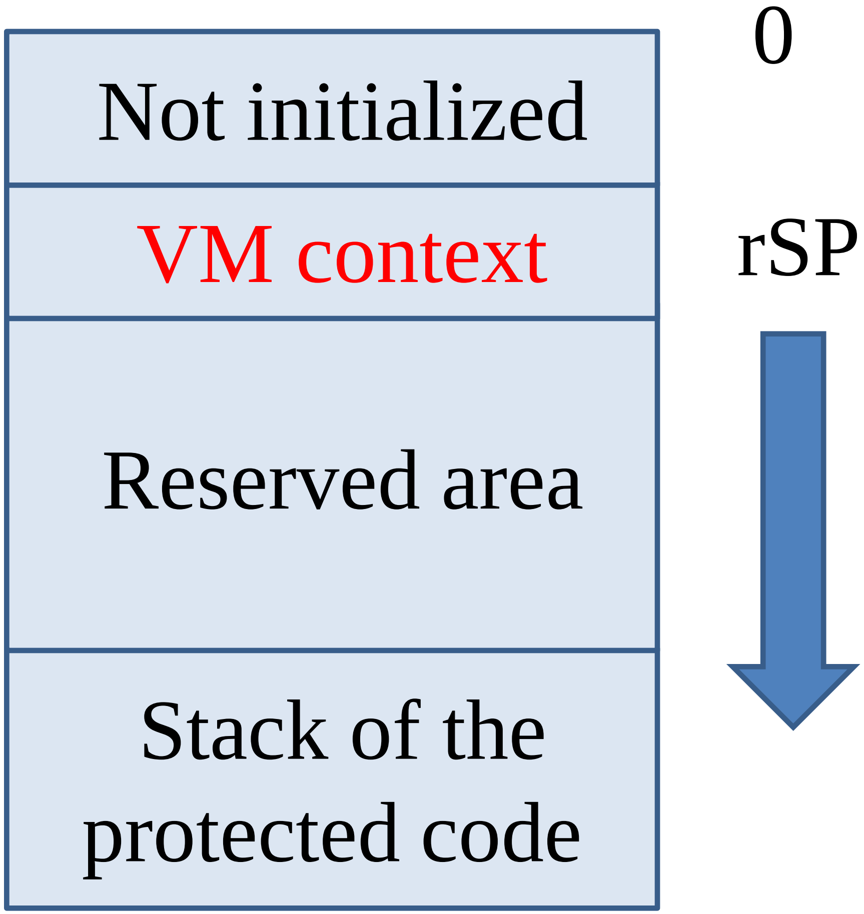
VM context

- Contains variables associated with processor registers
- Contains VM state
- Its location can be easily found in most cases

VM context location

- Dynamically allocated memory (VirtualAlloc, HeapAlloc)
- Global memory (access via spinlock)
- Stack

VM stack context layout



0

Not initialized

VM context

rSP

Reserved area

Stack of the protected code

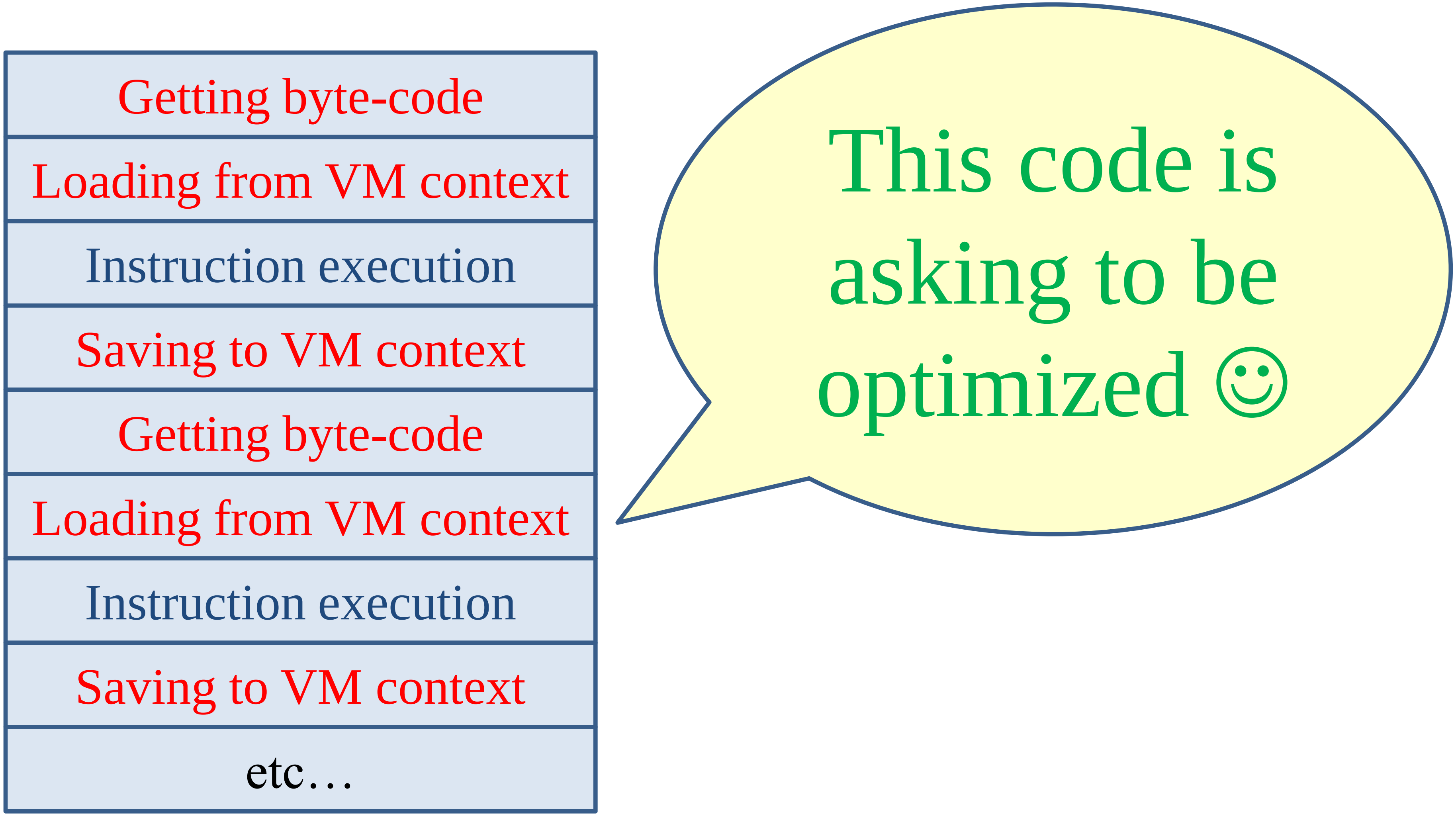
«Virtualized» addition

```
void unoptimal_addition( int a, int b, int *p )
{
    int u, v, t, *r;

    u = a;
    v = b;
    r = p;

    t = u + v;
    *r = t;
}
```

Virtualized code execution



Getting byte-code

Loading from VM context

Instruction execution

Saving to VM context

Getting byte-code

Loading from VM context

Instruction execution

Saving to VM context

etc…

This code is asking to be optimized ☺

Code devirtualization

- We can locate VM context
- We can get CFG in most cases
- We can use common code optimization algorithms to deobfuscate a virtualized code

Code morphing

- Used to increase resistance to the static analysis
- Used for the CFG obfuscation
- Used to increase VM body analyzing complexity

Code morphing and CFG obfuscation

It’s a difficult task to decompile a machine code

Therefore protectors don’t even try to do it ☺

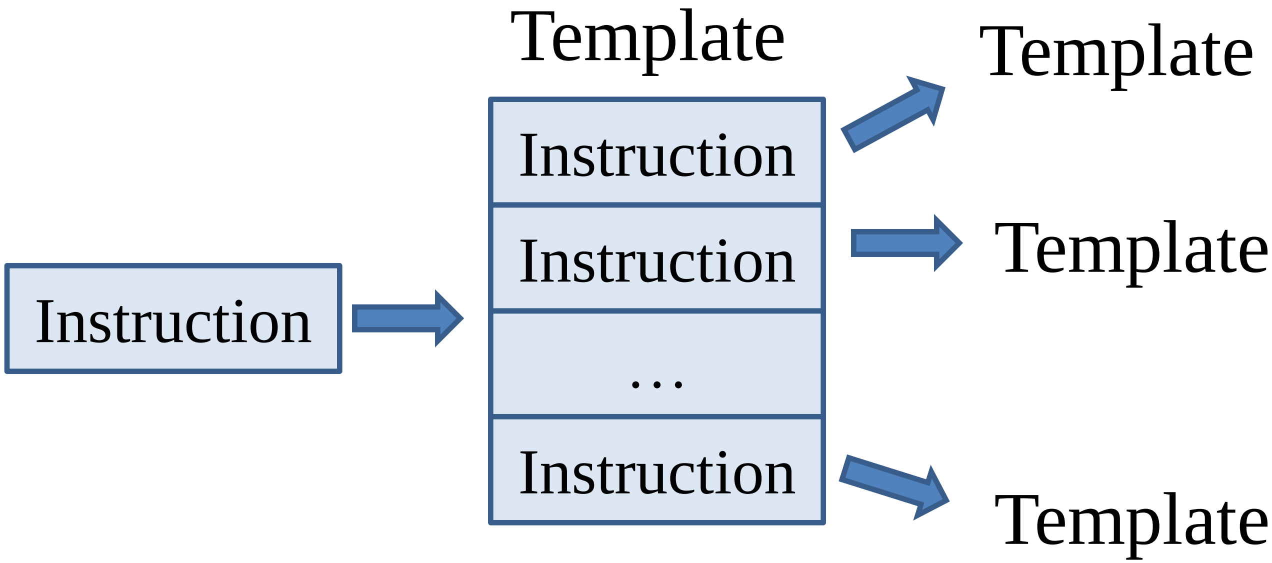
Code morphing and CFG obfuscation

Data dependencies analysis is weak in protectors

Therefore they are limited in choice of obfuscation techniques

Code morphing common techniques

Recursive templates



Code morphing common techniques

- Dead code insertion
- Garbage code insertion
- Opaque predicates
- Jump address calculation
- Code cloning

Morphed code deobfuscation

- Decompilation into IR
- IR instruction emulation
- Collecting variables values
- Emulation-based deobfuscation techniques

Ariadne engine

- An engine for RE
- Can be used as IDA plugin
- Enables PE format analyzing, disassembling and modifying
- Supports GP, FPU, MMX, SSE, SSE2, SSE3, SSSE3, SSE4.1, SSE4.2, SSE4a, VMX, SMX

Ariadne engine

- Supports assembler instructions translation into Ariadne Intermediate Representation (AIR)
- Supports IR instructions emulation
- Contains emulator-based code tracing mechanisms

Ariadne engine

- Contains built-in trace deobfuscation (AIR Wave Deobfuscation Technology)

AIR Wave Deobfuscation Technology

- Static deobfuscation
  - based on the classical compiler theory approaches
  - doesn’t use emulation

AIR Wave Deobfuscation Technology

- Dynamic deobfuscation
  - uses Ariadne IR emulator
  - calculates values of variables
  - determines in a lot of cases where a pointer points to
  - used for dereferenced pointers deobfuscation

AIR Wave Deobfuscation Technology

- Deobfuscation techniques
  - dead code elimination
  - variables propagation
  - constant folding
  - math simplifications

AIR Wave Deobfuscation Technology

- Deobfuscation techniques
  - loop unrolling
  - common subexpression elimination
  - pointer analysis and alias classification

Our results

- Many obfuscators/protectors provide a weak obfuscation
- Ariadne engine can be effectively used for deobfuscation

AIR Wave Deobfuscation Technology

Tested on …

See it for yourself ☺

And our thanks go…

- To Rolf Rolles for his works about virtualization obfuscation unpacking
- To Leta Group for Ariadne sponsorship

Ariadne engine

http://ariadne.group-ib.ru

GROUP IB