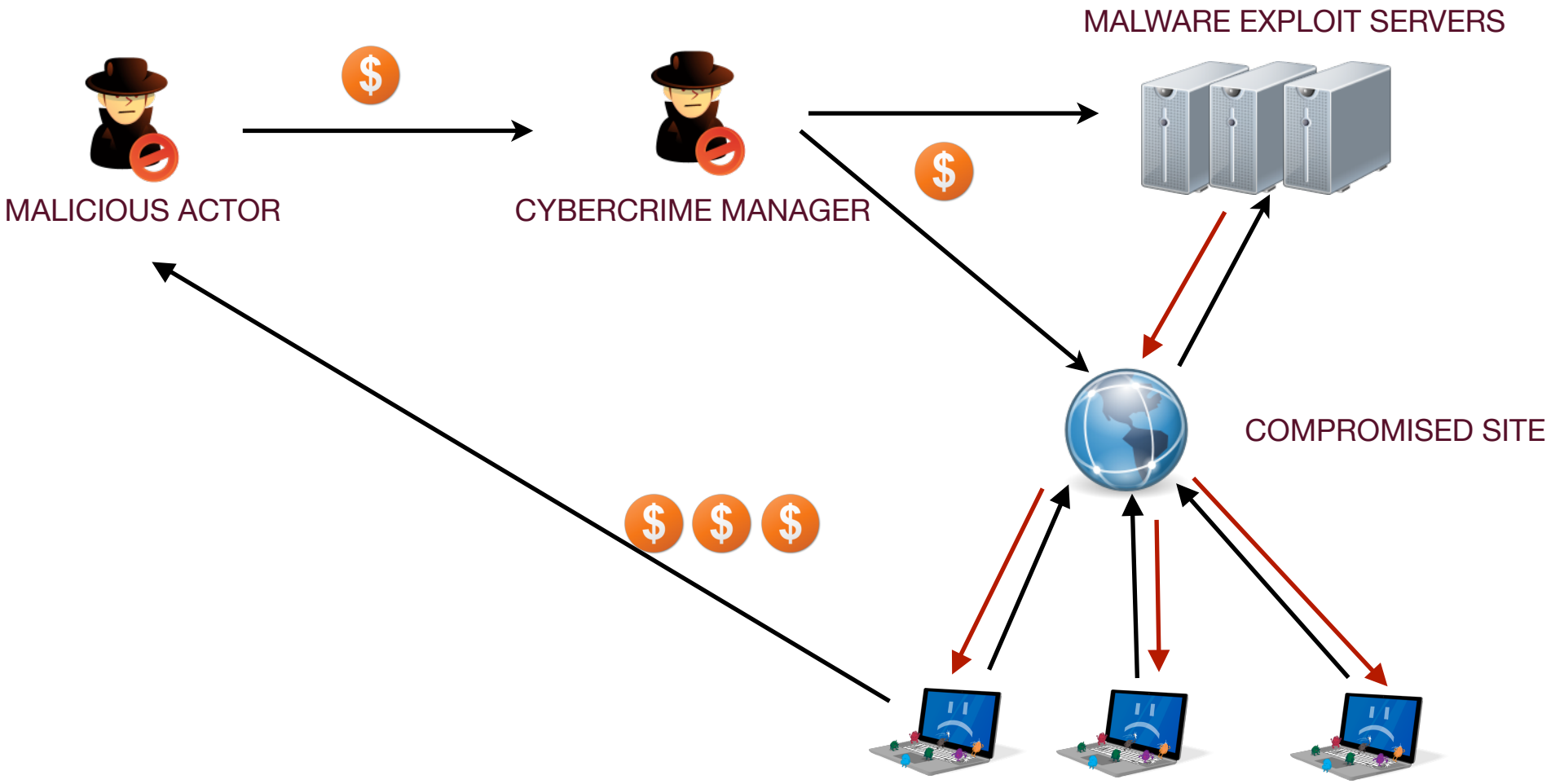
I progetti di ricerca a Verona: Progettare le difese vs studiare gli attacchi

Dipartimento di Informatica
Università di Verona

MALWARE IN MODERN CYBERCRIME



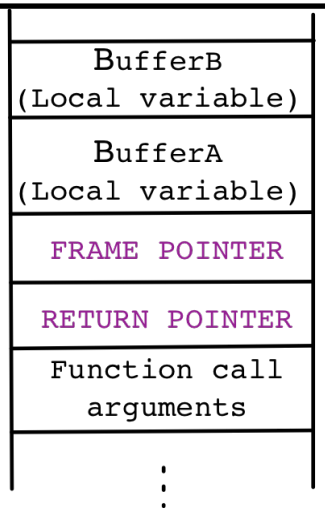
VULNERABILITIES

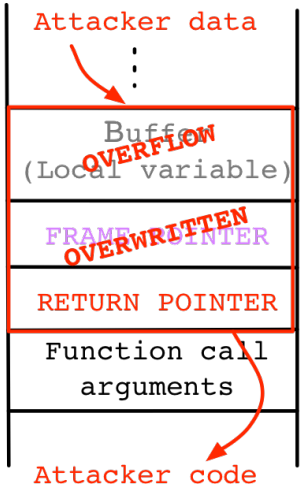
SYSTEM VULNERABILITIES

HUMAN ISSUE

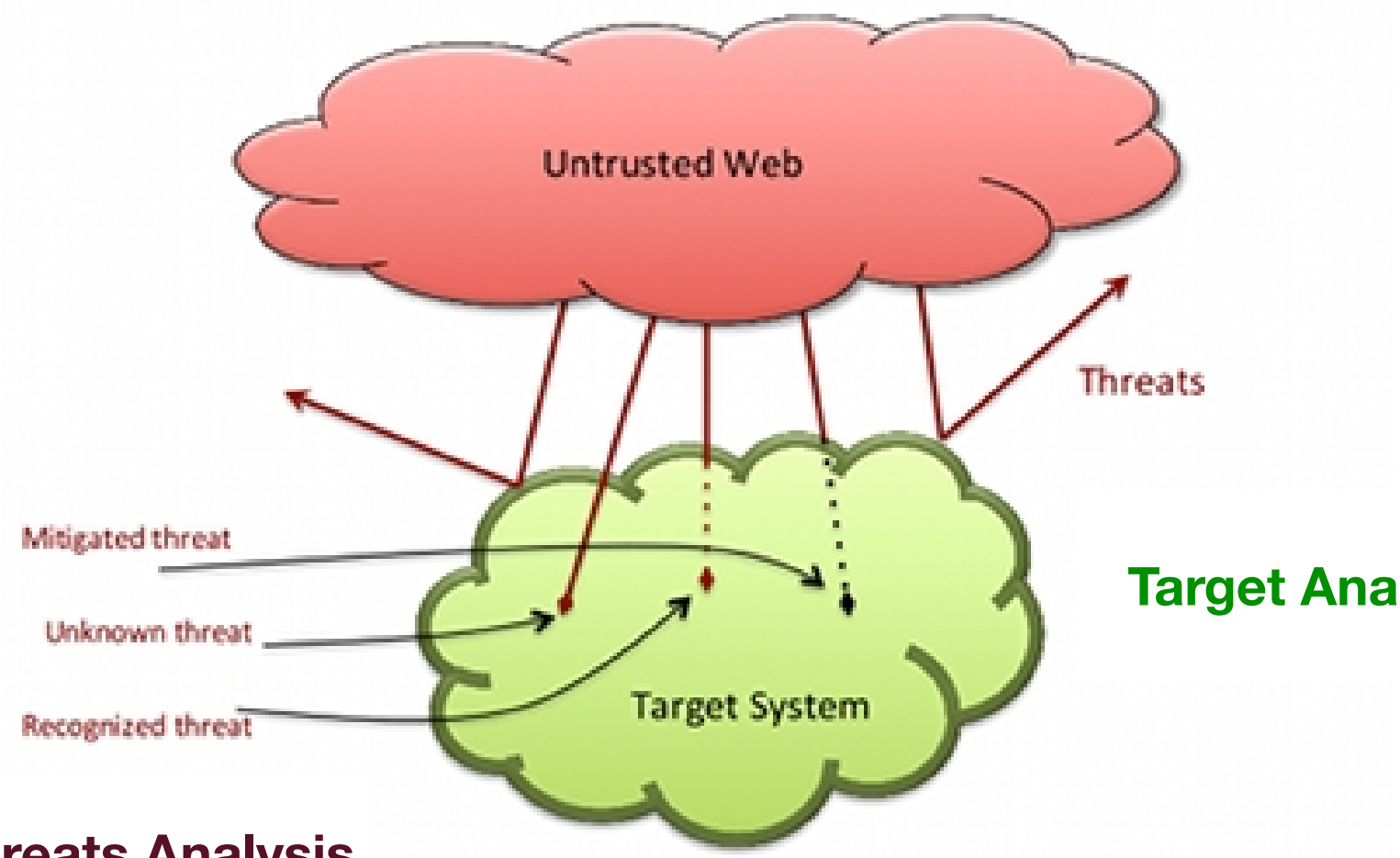
SLOW

```
void sampleFunction() {
        char bufferA[50];
        char bufferB[16];
        printf("where do you live?");
        gets(bufferA);
        strcpy(bufferB,bufferA);
        return;
}
main() {
        printf("Hello World!");
        sampleFunction();
        printf("All done!");
}
```





SECURITY SCENARIOS

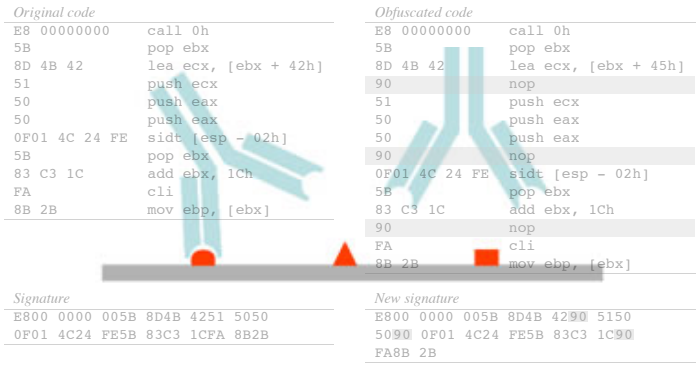


Target Analysis

Threats Analysis

FACE: Formal Avenue for Chasing malwarE

METAMORPHISM
(**static** analysis)



Original code

```
E8 00000000     call 0h
5B              pop ebx
8D 4B 42        lea ecx, [ebx + 42h]
51              push ecx
50              push eax
50              push eax
0F01 4C 24 FE   sidt [esp - 02h]
5B              pop ebx
83 C3 1C        add ebx, 1Ch
FA              cli
8B 2B           mov ebp, [ebx]
```

Obfuscated code

```
E8 00000000     call 0h
5B              pop ebx
8D 4B 42        lea ecx, [ebx + 45h]
90              nop
51              push ecx
50              push eax
50              push eax
90              nop
0F01 4C 24 FE   sidt [esp - 02h]
5B              pop ebx
83 C3 1C        add ebx, 1Ch
90              nop
FA              cli
8B 2B           mov ebp, [ebx]
```

Signature

```
E800 0000 005B 8D4B 4251 5050
0F01 4C24 FE5B 83C3 1CFA 8B2B
```

New signature

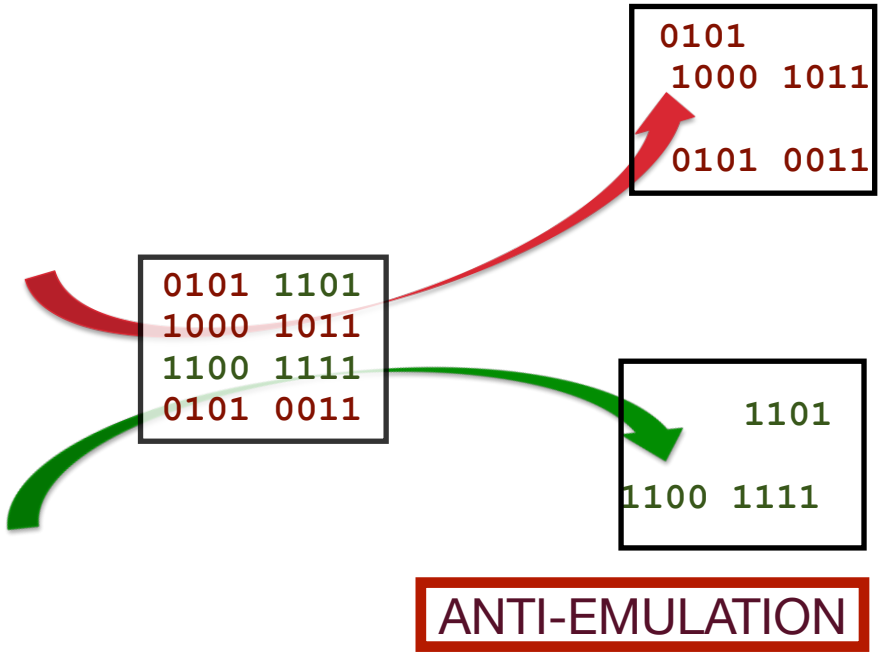
```
E800 0000 005B 8D4B 4290 5150
5090 0F01 4C24 FE5B 83C3 1C90
FA8B 2B
```

[SYMANTEC 2013]
2011 variants per malware rate **5:1**
2012 variants per malware rate **38:1**

CODE SHAPE MUTATION

MALWARE-ENVIRONMENT INTERPLAY
(**dynamic** analysis)

DORMANT BEHAVIOUR



ANTI-EMULATION

METAMORPHISM

Self-modifying malware contains the metamorphic engine

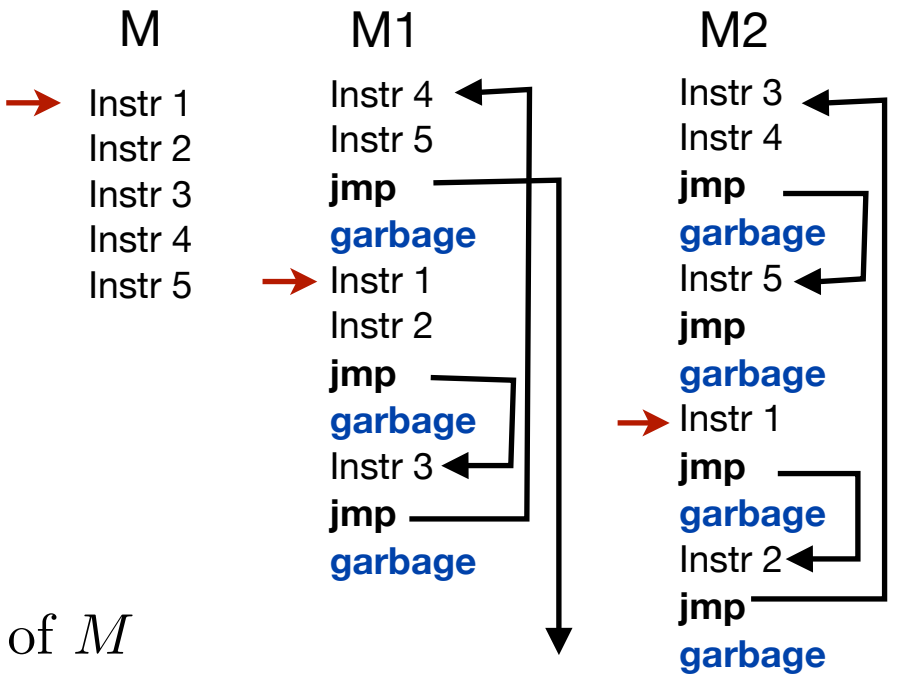
EXTRACT METAMORPHIC ENGINE BY
STATICALLY ANALYSING THE METAMORPHIC CODE
HARD

Phase semantics [2010 VR] precisely models code changes, but leads to an undecidable detection scheme

CODE SHAPE
MUTATION

Lose precision to gain decidability

$$\forall P : \mathit{flow}(P) = \mathit{flow}(M) \Rightarrow P \text{ is variant of } M$$



METAPHOR SIGNATURE

Merge of the flow graphs (of system calls) after 100 mutations

andrototal

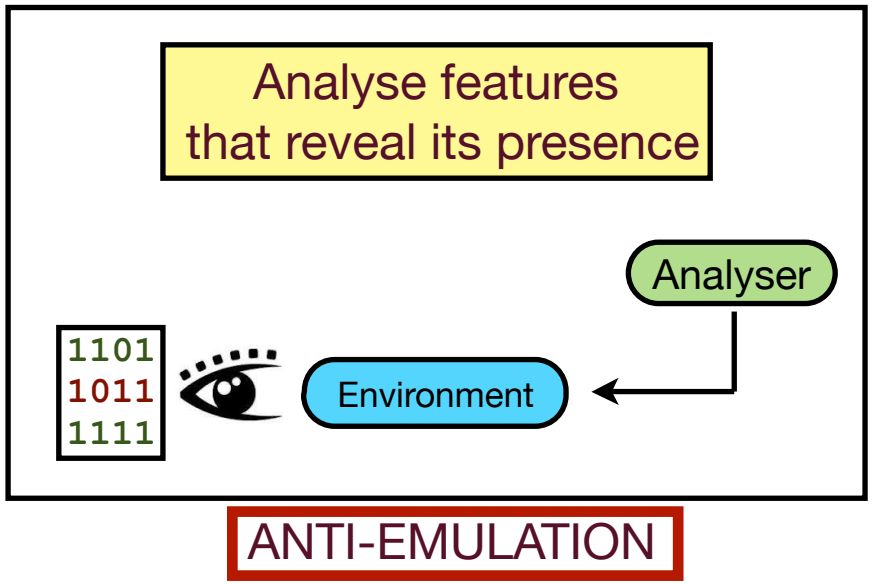
Validation of the model on mobile malware

MALWARE-ENVIRONMENT INTERPLAY

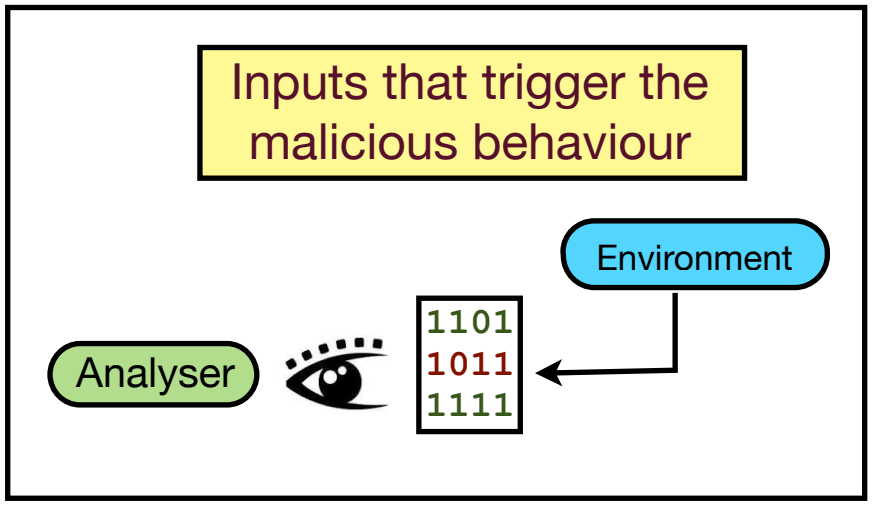
Study **program interaction**: how the execution of a program interferes with the execution of another program

Abstract Non-Interference (ANI) theory [2004 VR] on data

Lift ANI theory on programs
HARD



ANTI-EMULATION



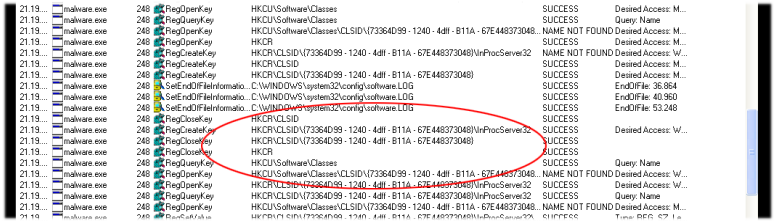
DORMANT BEHAVIOUR

Validation of the model on mobile malware

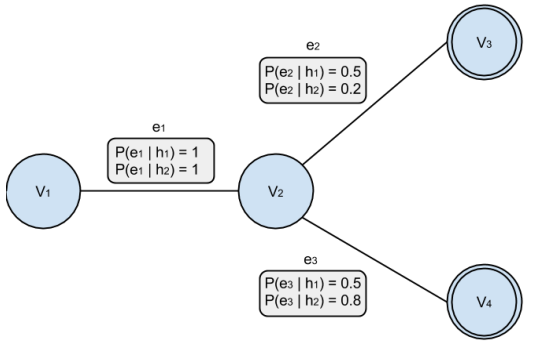
andrototal

IA and MALWARE ANALYSIS

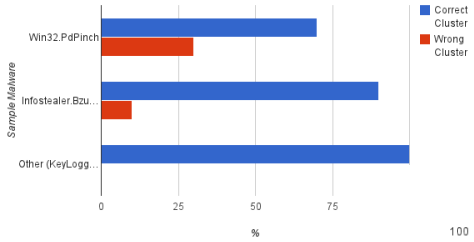
- Analisi attiva del malware attraverso un gioco
- Valutazione empirica della metodologia

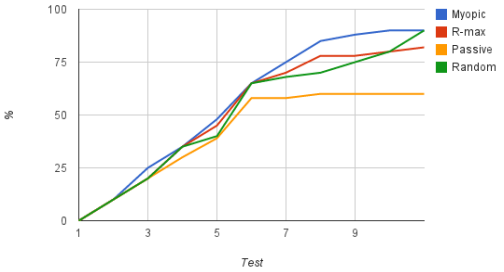


Log di esecuzione

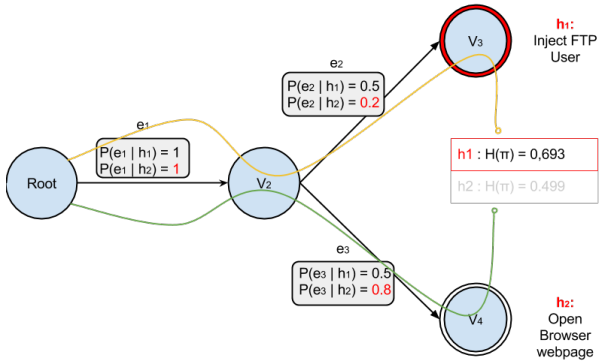


Creazione Modello





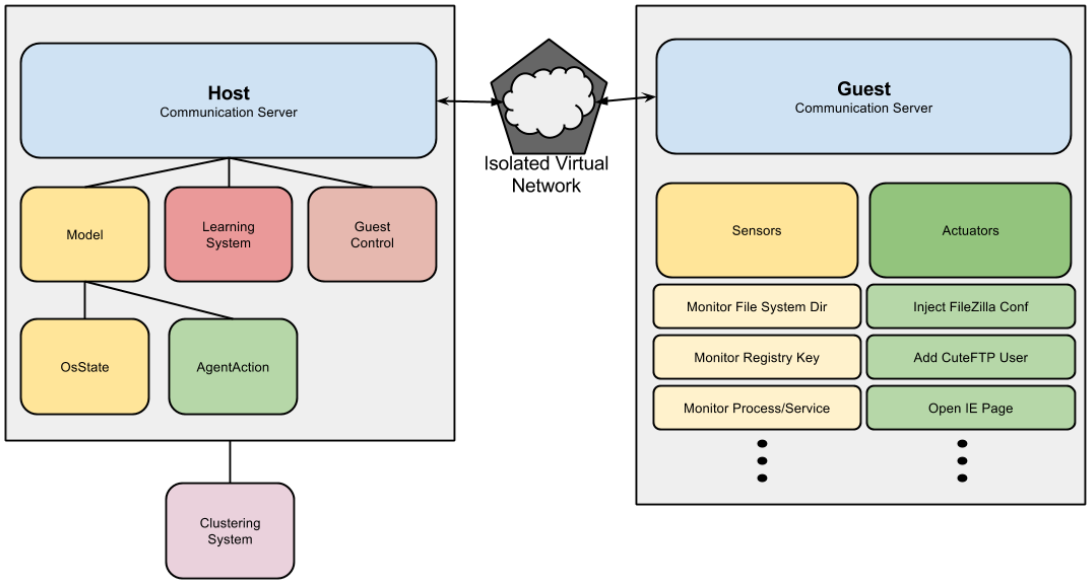
Valutazione Sperimentale



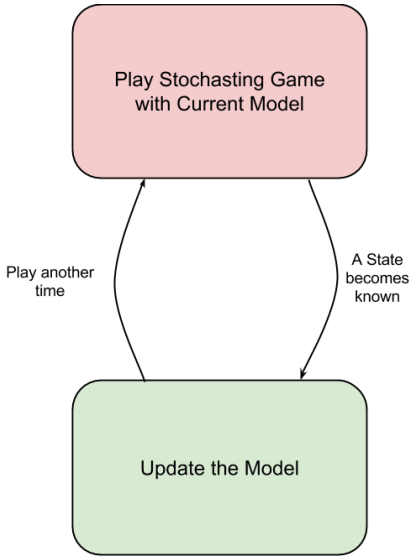
Scelta *honey*

IA and MALWARE ANALYSIS

- Strumenti sviluppati:
  - ambiente per analisi empirica
  - algoritmi di analisi
- Problemi aperti:
  - Generazione automatica del modello
  - Raffinamento delle tecniche di analisi



Ambiente sperimentale

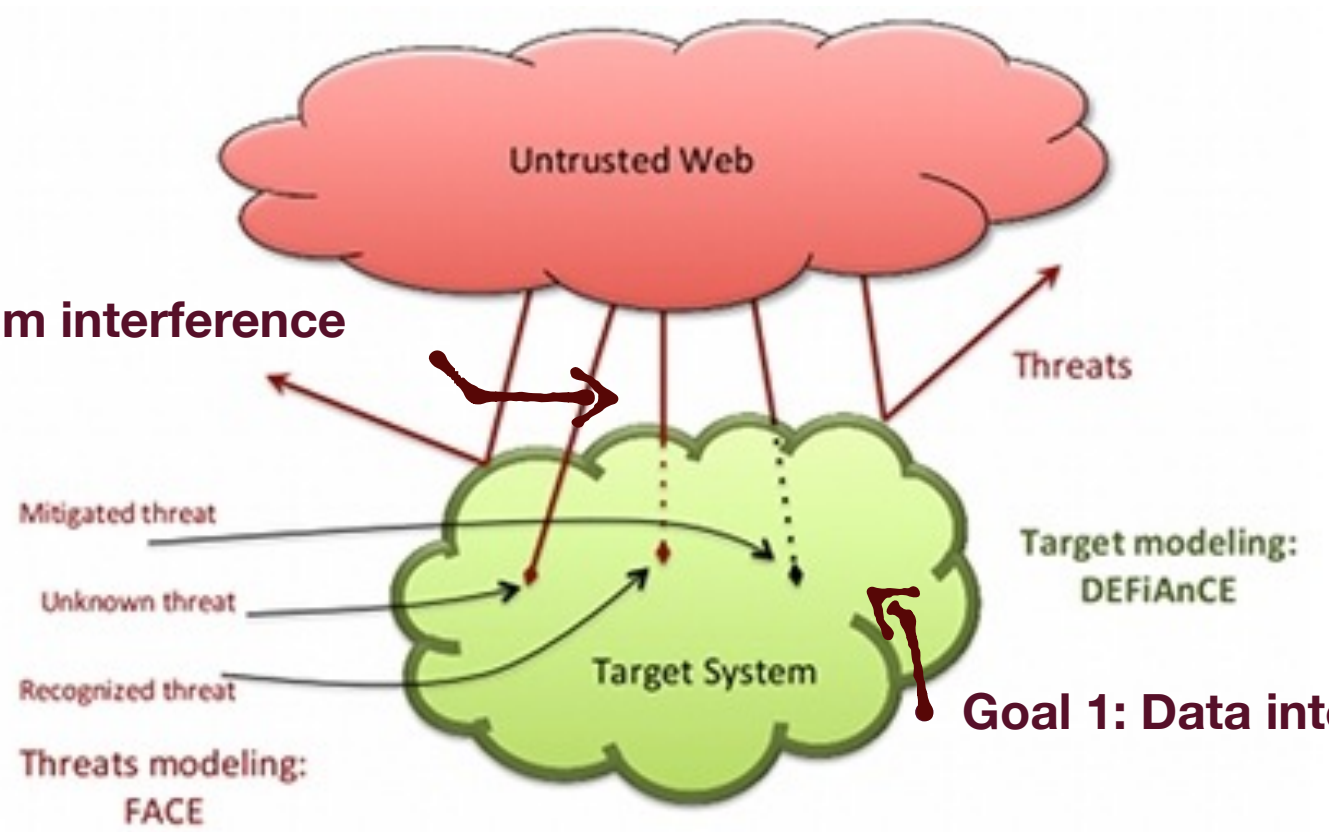


Metodologia analisi

DefiAnCE

A proactive Defence against Cyber Crime

- **Goal 2: Program interference**



Goal 1: Data interference

- **Goal 3: Data/program interference (Computer forensics)**

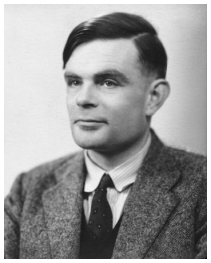
AbScript

Static analyzer for dynamic languages (eg. PHP)
based on abstract interpretation

Need to model and analyze the evolution of the code

Static analyzer which handles dynamic code mutations

TECHNICAL RISKS



Alan Turing

Kurt Godel

ANALYSIS
PRECISE vs COMPLEXITY vs COVERAGE

HIGH
COMPLEXITY
HARD

DYNAMIC
PARTIAL

RECOVERY

SPY

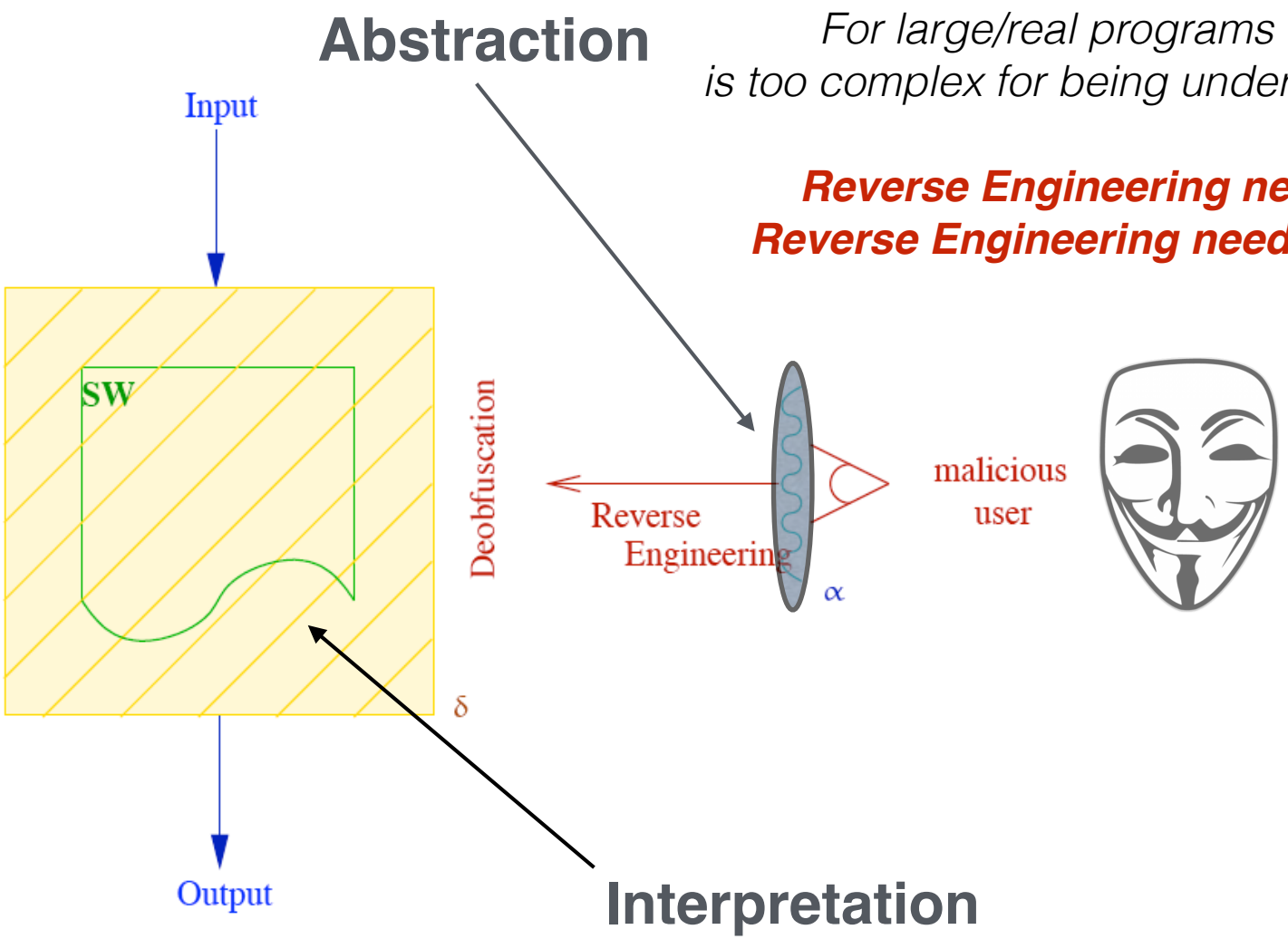
STATIC
EXHAUSTIVE

LOW
COMPLEXITY

PRECISE

LOOSE

DESIGN and MEASURE CODE PROTECTION

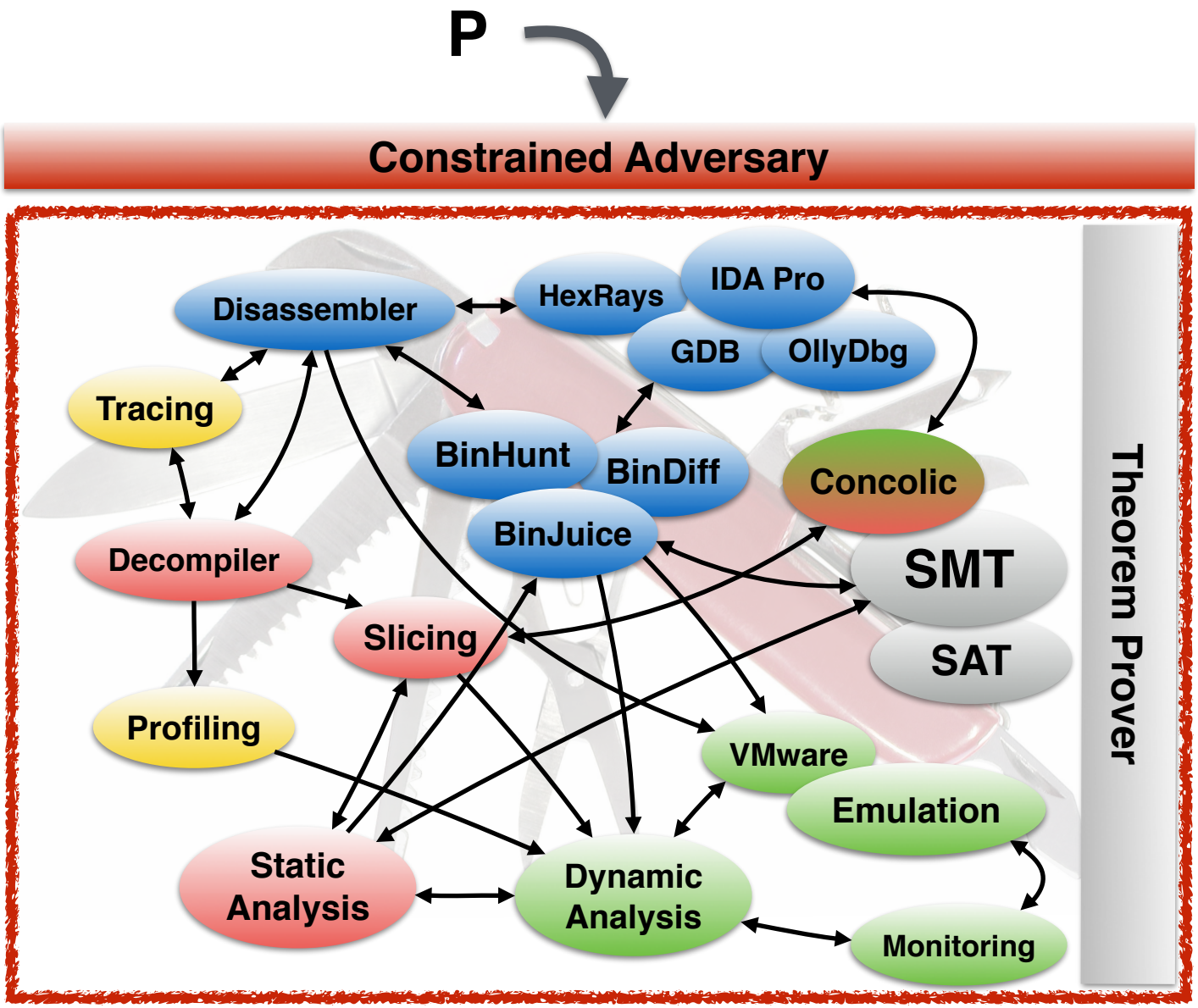


For large/real programs control/data flow is too complex for being understandable by humans:

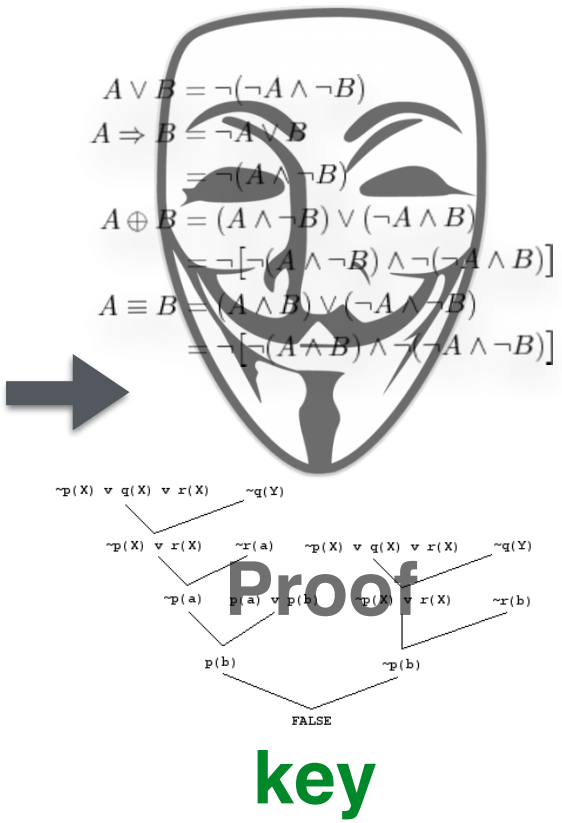
Reverse Engineering needs abstraction!
Reverse Engineering needs automated tools!

Abstract Interpretation is a general theory for approximating the semantics of dynamic systems (Cousot & Cousot 1977)

DESIGN and MEASURE CODE PROTECTION



- Each tool is an Abstract Interpretation

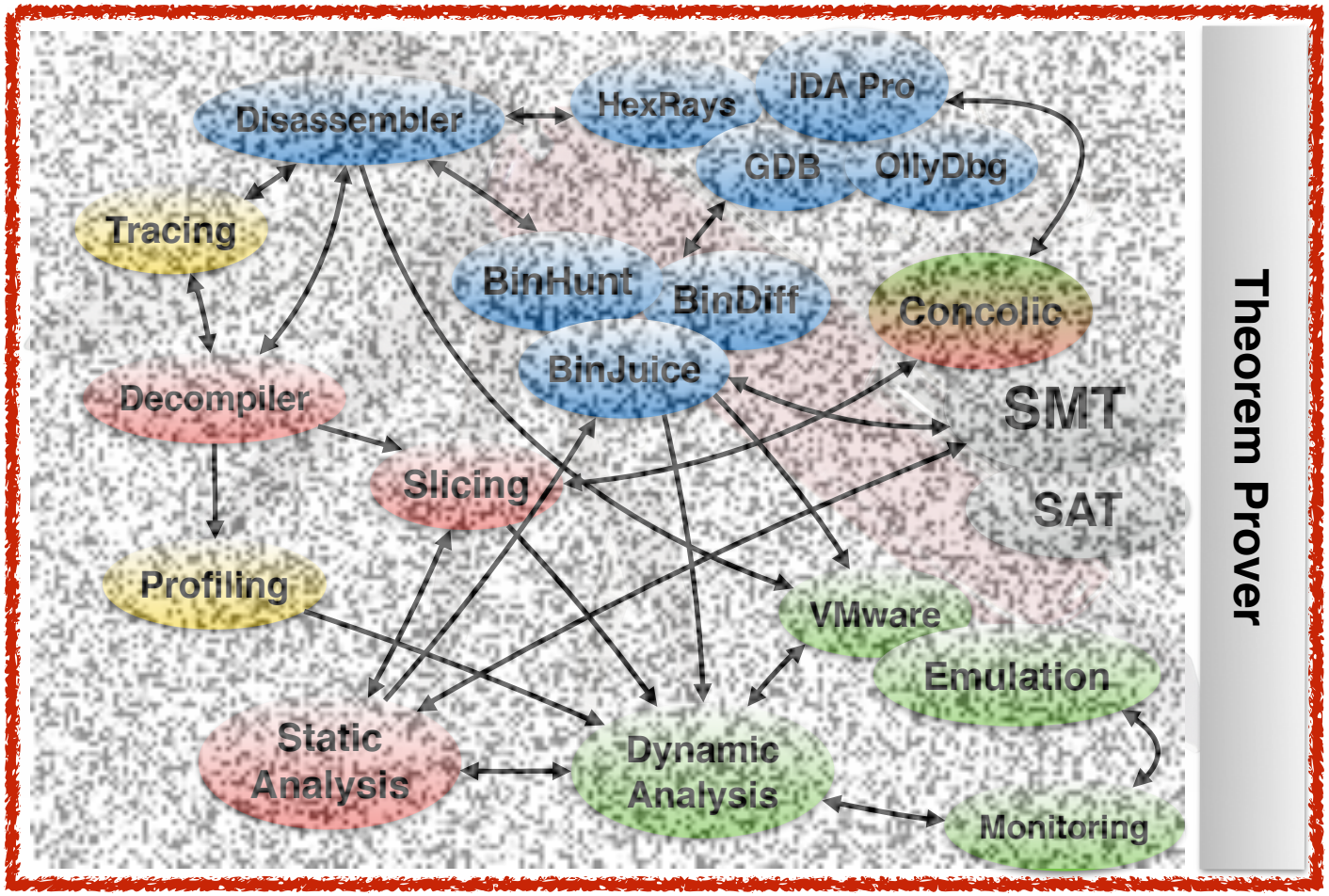


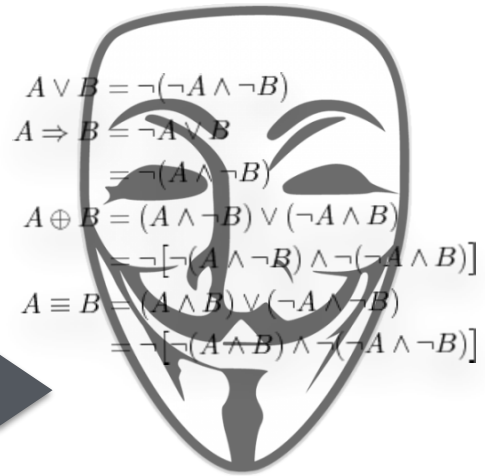
We can quantify the security achieved by looking at proof complexity

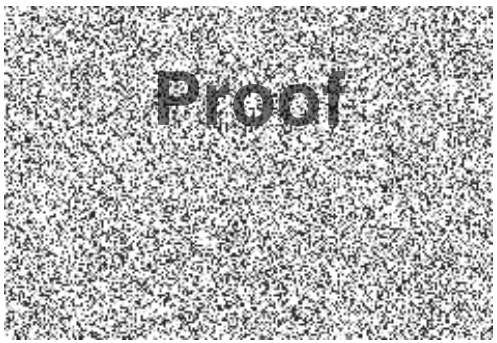
DESIGN and MEASURE CODE PROTECTION

- Transform code to make *all* tools blind

Constrained Adversary

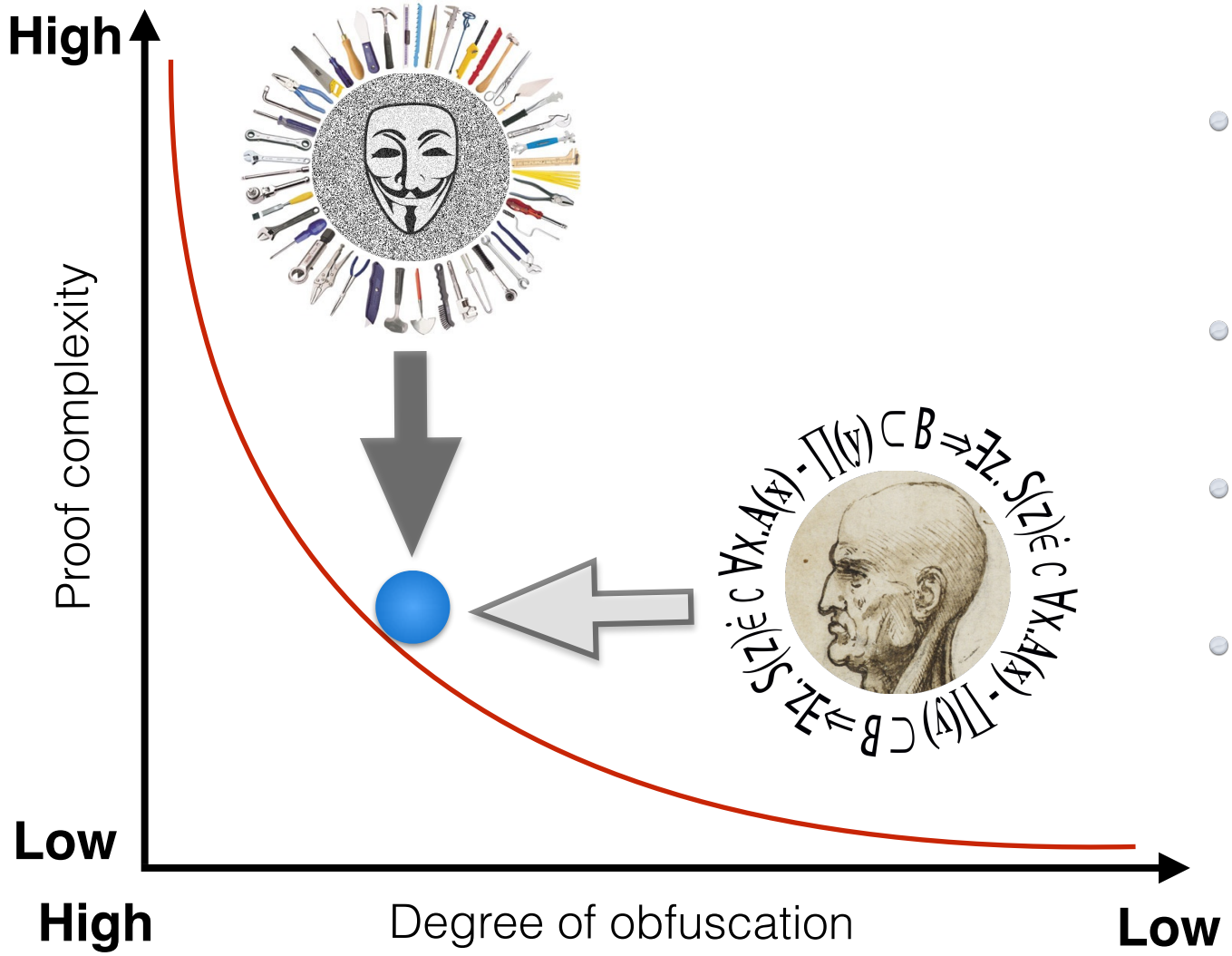






Removing noise means refining abstractions/complicating proofs (Giacobazzi et al. 2000/2012)

DESIGN and MEASURE CODE PROTECTION



- Force the attacker to use automated tools (programs of large size and highly interconnected)
- Design code transformations making tools blind
- Determine lower bounds for proof complexity in obfuscated code
- Measure the degree of noise/slowdown induced in obfuscation

By constraining the adversary within a theorem prover we can quantify the security achieved from obfuscation

PROJECTS

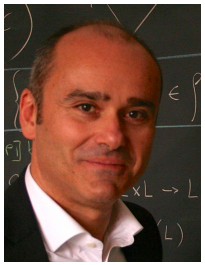
- FIRB-2013 (Coordinatore: Mila Dalla Preda)
  FACE: Formal Avenue for Chasing malwarE (marzo 2014 - febbraio 2017)

- SIR 2014 (Coordinatore: Isabella Mastroeni)
  DefiAnCE: proactive DEFence against Cyber crimE (in fase di valutazione da parte del Ministero)

- Joint Project (Coordinatore: Isabella Mastroeni)
  AbScript: Abstract Interpretation based Analysis of Scripting Languages
  (settembre 2014 - agosto 2016)

- Joint Project (Coordinatore: Roberto Giacobazzi)
  Interpretation-based design and measurement of code-protecting transformations
  (settembre 2014 - agosto 2016)

PEOPLE



Roberto Giacobazzi

Abstract Interpretation, Malware analysis, non-interference, static analyzer PHP

Fausto Spoto

Abstract Interpretation, static analyzer PHP

Isabella Mastroeni

Abstract Interpretation, Malware analysis, non-interference, static analyzer PHP, game theory and AI

Alessandro Farinelli

game theory and AI

Mila Dalla Preda

Abstract Interpretation, Malware analysis, non-interference, static analyzer PHP