Logical complexity as a resource for security by obscurity

Dusko Pavlovic

Royal Holloway and Twente

October 2011

Outline

Introduction

Background: Security and obscurity

Idea: Attack models

Approach: Directions

Summary

Outline

Introduction

What is a resource?

Complexities as resources

Background: Security and obscurity

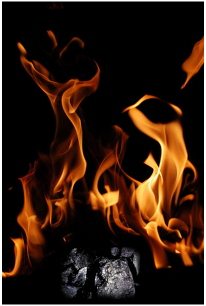
Idea: Attack models

Approach: Directions

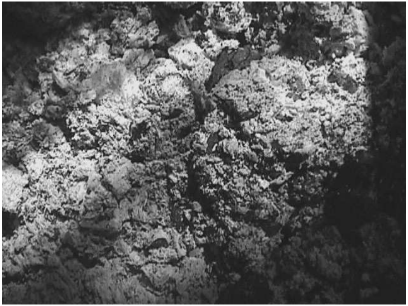
Summary

Resource

Utility

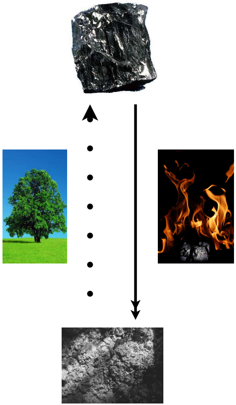


Residue

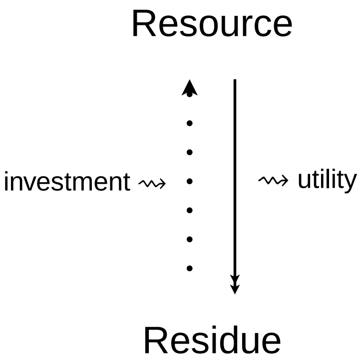


Exploitation is easy

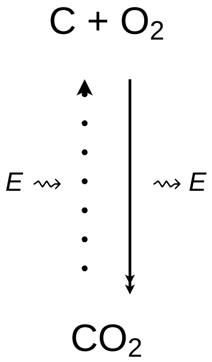
Regeneration is hard



Resources yield one-way functions

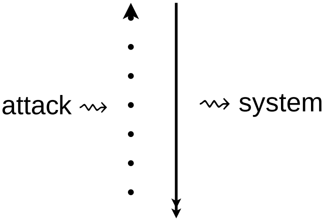


Resources yield one-way functions



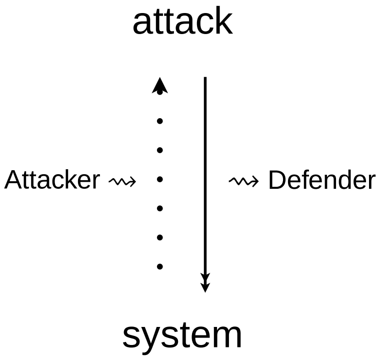
Computational resources for security

$$11,213 \times 756,839$$



$$8,486,435,707$$

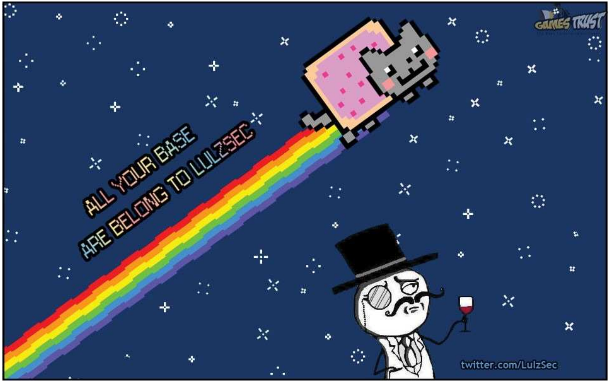
Wanted: *"Logical resources for security"*



Question

Do logical resources for security exist?

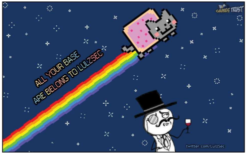
Notation



ATTACK

Idea

Suppose that you are given a system C such that



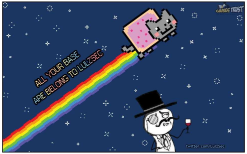
C

$$\Downarrow$$

$$P = NP$$

Idea

Suppose that you are given a system C such that



C

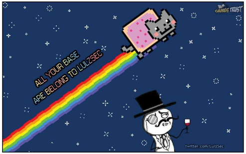
$$\Downarrow$$

$$P = NP$$

Would you consider it secure?

Idea

Suppose that you are given a system $\mathcal{L}$ such that



$\mathcal{L}$

$$\Downarrow$$

$$P \neq NP$$

Would you consider it secure?

Idea

Theorem

System $\mathcal{L}$ is secure enough to protect an account with \$1,000,000

Proof.

Proving $P \neq NP$ yields \$1,000,000 from Clay Institute. □

Alarm

If $P \neq NP$, then this is security by obscurity:

- security of the system $\mathcal{L}$ is based on
- obscurity of the proofs of $P \neq NP$

Outline

Introduction

Background: Security and obscurity

Idea: Attack models

Approach: Directions

Summary

What is security by obscurity?

Kerckhoffs' Principle

"The system must not be required to be secret, and it must be able to fall into the hands of the enemy without inconvenience."

Jean Guillaume Auguste Victor François Hubert Kerckhoffs

What is security by obscurity?

Shannon's Maxim

"The enemy knows the system."

Claude Shannon

Secure key *vs* obscure system

Lock can only be opened using the correct key

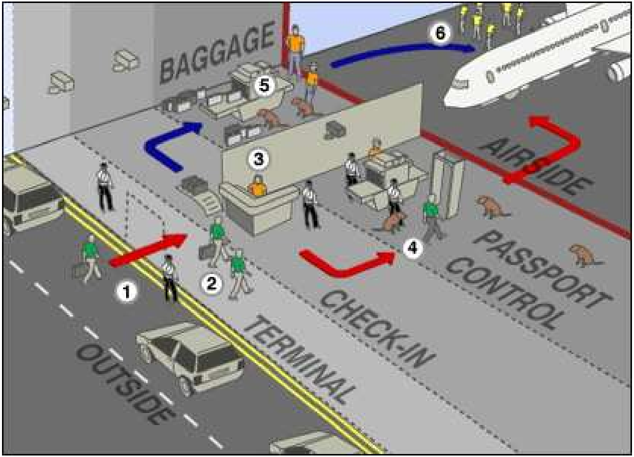
Secure key *vs* obscure system

. . . and **not** by breaking the system

Outside cryptography

there are systems with no key

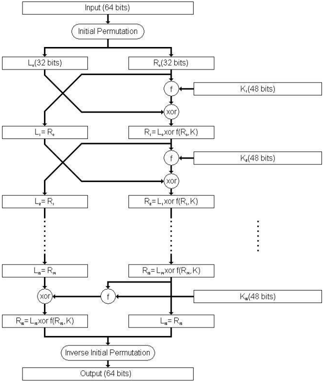
Outside cryptography



there is not much more to hide except the system

In cryptography

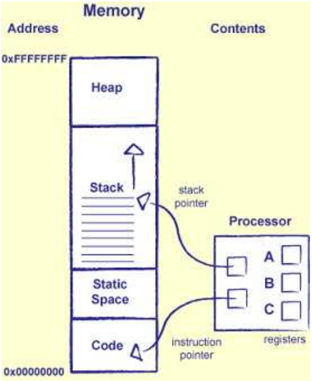
- keys = data
- system = program



In computation

(Gödel, Von Neumann, Kleene)

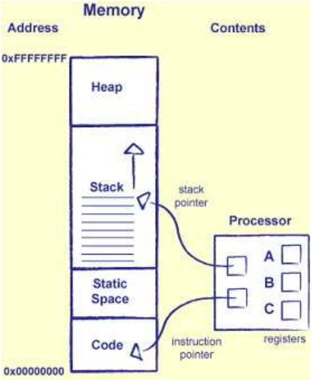
- keys = data = program
- system = program = data



In computation

(Gödel, Von Neumann, Kleene)

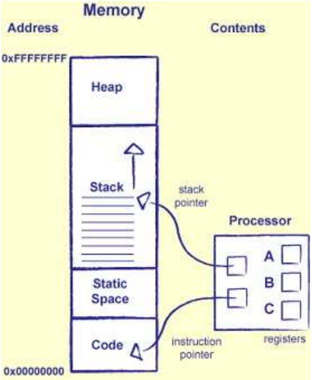
- keys = data = program
  - data $\rightsquigarrow$ encrypted
- system = program = data
  - programs $\rightsquigarrow$ obfuscated



In computation
(Gödel, Von Neumann, Kleene)

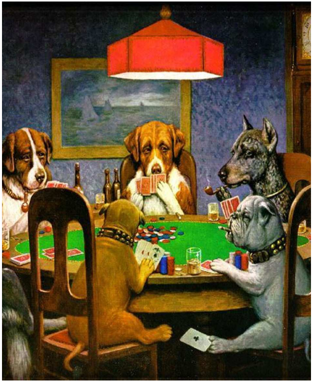
- keys = data = program
  - data $\rightsquigarrow$ encrypted
- system = program = data
  - programs $\rightsquigarrow$ obfuscated

Theorem [Barak et al]
Obfuscators do not exist.



In poker

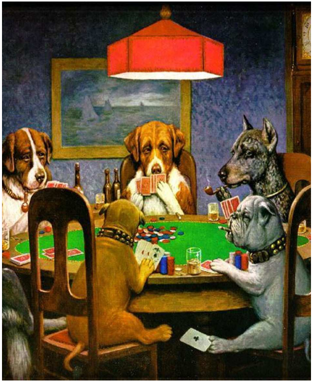
- keys = hands of cards
- system = tactics



In games

(Von Neumann-Morgenstern, Harsanyi, Aumann. . .)

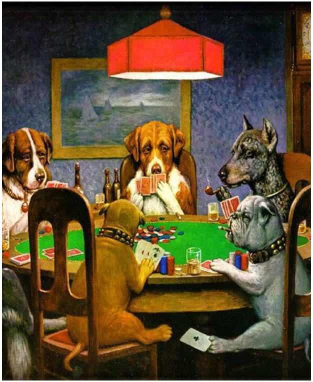
- keys = players' states
- system = players' types



In games

(Von Neumann-Morgenstern, Harsanyi, Aumann. . .)

- keys = players' states
  - (im)perfect information
- system = players' types
  - (in)complete information

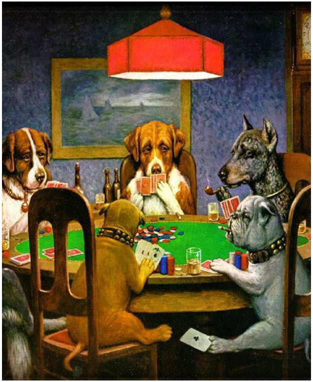


In games

(Von Neumann-Morgenstern, Harsanyi, Aumann. . .)

- keys = players' states
  - (im)perfect information
- system = players' types
  - (in)complete information

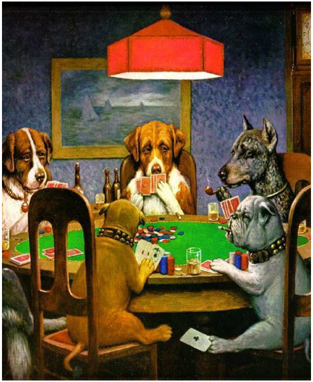
Kerckhoffs' Principle
Security is a game of imperfect information.



In security games
(Kerckhoffs, Shannon)

- keys ⇝ cryptanalysis
  - hard
- system ⇝ decompilation
  - easy

Kerckhoffs' Principle
Security is a game of imperfect information.



Claim

Security is a game of incomplete information

Claim

There is security by obscurity even in cryptography

- **not** through obfuscated code
- **but** through logically complex algorithms

Outline

Introduction

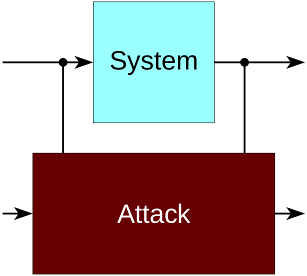
Background: Security and obscurity

Idea: Attack models

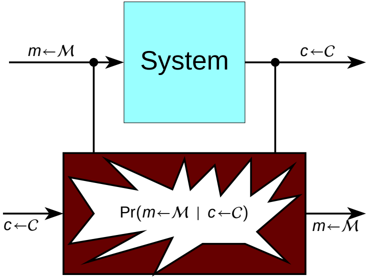
Approach: Directions

Summary

Security as a game

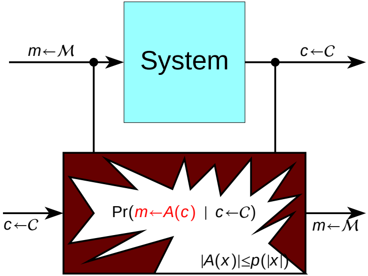


Shannon's attacker: computationally unbounded
(omnipotent computer)



If a source conveys some information,
the attack will extract that information.

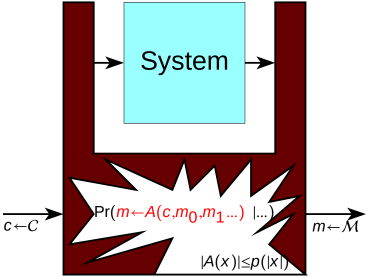
Diffie-Hellman's attacker: computationally bounded
(real computer)



Public key determines the corresponding private key, but the attacker cannot compute one from the other.

Adaptive attacker: queries the system

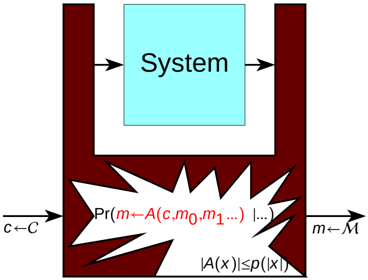
(still a real computer)



If there is a vulnerability,
an attack algorithm will make use of it.

Adaptive attacker: queries the system

(still a real computer)

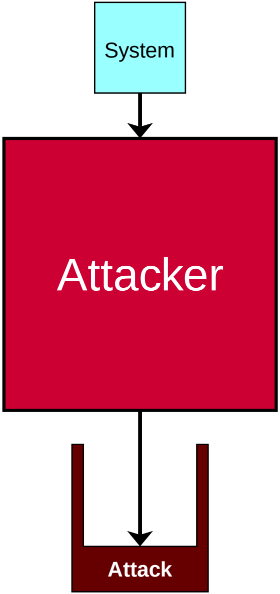


If there is a vulnerability,
an attack algorithm will make use of it.

But where do the attack algorithms come from?

Kerckhoffs' attacker: logically unbounded

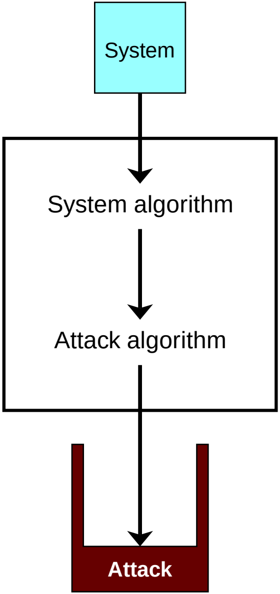
(omnipotent programmer)



If there is an attack,
the attacker will find it.

Kerckhoffs' attacker: logically unbounded

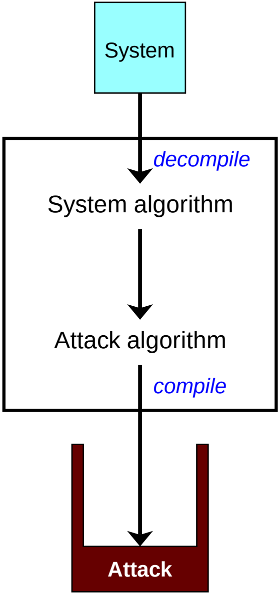
(omnipotent programmer)



If an attack exists,
the attacker will find it

Kerckhoffs' attacker: logically unbounded

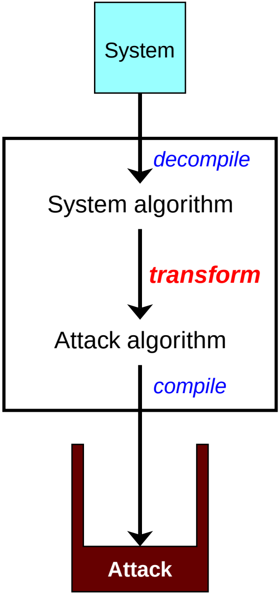
(omnipotent programmer)



If an attack exists,
the attacker will find it.

Kerckhoffs' attacker: logically unbounded

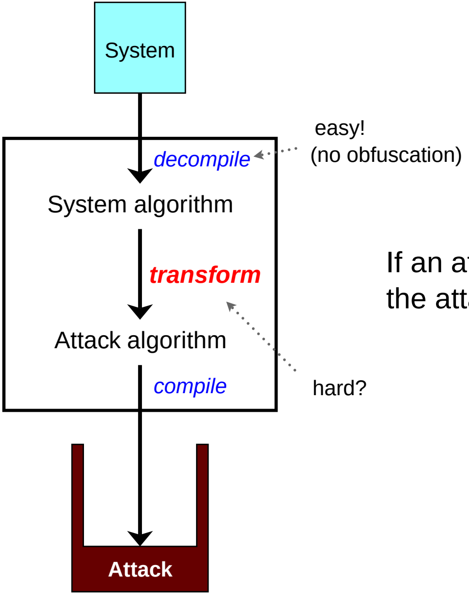
(omnipotent programmer)



If an attack exists,
the attacker will find it

Kerckhoffs' attacker: logically unbounded

(omnipotent programmer)



If an attack exists,
the attacker will find it

Real attacker: logically bounded

(someone's student)

power	*unbounded*	*bounded*
computational	Shannon	Diffie-Hellman
rationality	Cournot	Simon
logical	Kerckhoffs	?????

Idea

$$\frac{\text{computational complexity}}{\text{secrecy}} = \frac{\text{logical complexity}}{\text{obscurity}}$$

Two directions

- hinder adaptation of attack to system
- improve adaptation of system to attack

Two directions

- hinder adaptation of attack to system
  - use **algorithmic information theory** in security
- improve adaptation of system to attack
  - use **epistemic game theory** in security

Outline

Introduction

Background: Security and obscurity

Idea: Attack models

Approach: Directions

x-direction: Algorithmic information theory

y-direction: Epistemic game theory

Summary

Question

What is logical complexity?

- Which proofs / algorithms are hard to construct?

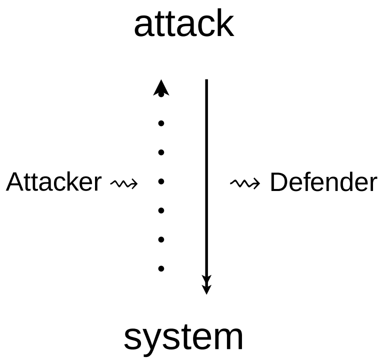
Question

What is logical complexity?

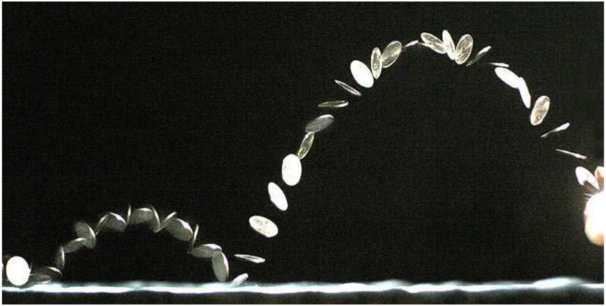
- Which proofs / algorithms are hard to construct?
- Which attack algorithms are hard to derive from which system algorithms?

Question

Is there "one-way programming"?

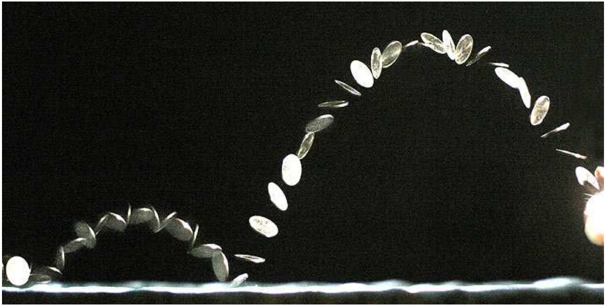


Predictability and probability



$$\underbrace{0101010101010101010\cdots01}_{\text{100 times}}$$

Predictability and probability



$$\Pr(01010101010101010101010\cdots01) = 2^{-100}$$
$$\Pr(01001100011100001111 0\cdots11) = 2^{-100}$$
$$\Pr(11010001001101010010 1\cdots00) = 2^{-100}$$

Predictable events are improbable

"We arrange in our thought all possible events in various classes; and we regard as extraordinary those classes which include a very small number. In the game of heads and tails, if heads comes up a hundred times in a row then this appears to us extraordinary, because the almost infinite number of combinations that can arise in a hundred throws are divided in regular sequences, or those in which we observe a rule that is easy to grasp, and in irregular sequences, that are incomparably more numerous."

Pierre-Simon Laplace

Probability is not about predictability

"In everyday language we call random those phenomena where we cannot find a regularity allowing us to predict precisely their results. Generally speaking, there is no ground to believe that random phenomena should possess any definite probability. Therefore, we should distinguish between randomness proper (as absence of any regularity) and stochastic randomness (which is the subject of probability theory). There emerges the problem of finding reasons for the applicability of the mathematical theory of probability to the real world."

Andrei N. Kolmogorov

Probability is not about events

- Probability only describes ensembles of events

Probability is not about events

- Probability only describes ensembles of events
- Information theory only speaks of global properties.

Probability is not about events

- Probability only describes ensembles of events
- Information theory only speaks of global properties.
- "Which local function is entropy the integral of?"

How do we predict events?

- $\underbrace{01010101010101010101010 \cdots 01}_{100}$ can be written as
  - $(01)^{50}$

How do we predict events?

- $\underbrace{0101010101010101010\cdots 01}_{100}$ can be written as
  - $(01)^{50}$
  - do i=1..50 write 01 od

How do we predict events?

- $\underbrace{0101010101010101010\cdots01}_{100}$ can be written as
  - $(01)^{50}$
  - do i=1..50 write 01 od
- $\underbrace{0100110001110000111 10\cdots11}_{100}$ can be written as
  - $\underbrace{0^1 1^1 0^2 1^2 \cdots 0^i 1^i \cdots}_{100}$
  - i=1; do until length=100 write $0^i 1^i$; i = i+1 od

How do we predict events?

- $\underbrace{01010101010101010101\cdots01}_{100}$ can be written as
  - $(01)^{50}$
  - do i=1..50 write 01 od
- $\underbrace{01001100011100001111 0\cdots11}_{100}$ can be written as
  - $\underbrace{0^1 1^1 0^2 1^2 \cdots 0^i 1^i \cdots}_{100}$
  - i=1; do until length=100 write $0^i 1^i$; i = i+1 od
- $\underbrace{11010001001101010010 1\cdots00}_{100}$ can be written as
  - print $11010001001101010010 1\cdots00$

How do we predict events?

- $\underbrace{01010101010101010101\cdots01}_{100}$ can be written as
  - $(01)^{50}$
  - do i=1..50 write 01 od
- $\underbrace{01001100011100001110\cdots11}_{100}$ can be written as
  - $\underbrace{0^1 1^1 0^2 1^2 \cdots 0^i 1^i \cdots}_{100}$
  - i=1; do until length=100 write $0^i 1^i$; i = i+1 od
- $\underbrace{11010001001101010010 1\cdots00}_{100}$ can be written as
  - print $110100010011010100101\cdots00$ ⇜ random

Predictable = programmable

Ray Solomonoff (1960): Science as programming

- $\Pr(1 \mid 010101010101010101010 \cdots 01) = 0$
- $\Pr(1 \mid 010011000111000011110 \cdots 11) = 1$
- $\Pr(1 \mid 110100010011010100101 \cdots 00) = \frac{1}{2}$

Algorithmic information

Definition (Solomonoff 1960, Komogorov 1965)

Algorithmic information contained in data a is the length of the shortest program that outputs a

$$C(a) \quad = \quad \bigwedge_{\{p\}()=a} |p|$$

Algorithmic information

Theorem (Schack 1997)

Algorithmic information is the local function that yields entropy as its global average

$$H(q) \quad \approx \quad \int_{i \in I} C(q_i)$$

Algorithmic distance

Definition

Algorithmic distance between $a, b \in \mathbb{N}$ is the length of the shortest program that inputs a and outputs b

$$C(a,b) \quad = \quad \bigwedge_{\{p\}(a)=b} |p|$$

Idea

- Algorithmic information is a measure of impredictability.

Idea

- Algorithmic information is a measure of impredictability.
- Is algorithmic information a good concept of logical complexity?

Idea

Charles Bennett: Logical depth

- of an organism: the time it takes to evolve
  - virus: computationally simple, logically deep

Idea

Charles Bennett: Logical depth

- of an organism: the time it takes to evolve
  - virus: computationally simple, logically deep
- of an algorithm: the time complexity of its derivation
  - PRIMES: computationally simple, logically deep

Idea

Charles Bennett: Logical depth

- of an organism: the time it takes to evolve
  - virus: computationally simple, logically deep
- of an algorithm: the time complexity of its derivation
  - PRIMES: computationally simple, logically deep
- logical depth measures complexity
  - of evolutionary processes
  - as computational processes

Logical complexity

Definition

Logical complexity of $a \in \mathbb{N}$ is the time complexity of the simplest program that outputs a

$$D(a) \quad = \quad \bigwedge_{\substack{\{p\}()=a \\ C(p)=|a|}} |\{p\}|$$

Logical distance

Definition

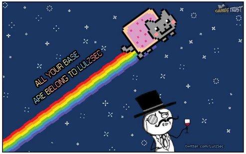
Logical distance of $a, b \in \mathbb{N}$ is the complexity of the simplest program that inputs a and outputs b

$$D(a,b) \quad = \quad \bigwedge_{\substack{\{p\}(a)=b \\ C(a,b)=|p|}} |\{p\}|(|a|)$$

Idea of logical security

S is secure if $D(S, A)$ is "large" for all attacks A.

Idea of logical security



$\mathcal{L}$

$$\Downarrow$$

$$P \neq NP$$

Idea of logical security

$$D\Big(\mathcal{L},$$  $$_{\mathcal{L}}\Big) \;\geq\; D\Big(\mathcal{L}, \ulcorner P \neq NP \urcorner\Big)$$

Task

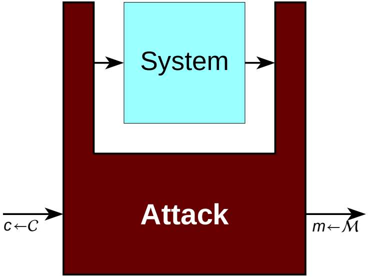
Implement this idea.

Approach

Epistemic game theory of security.

Adaptive attacker: queries the system

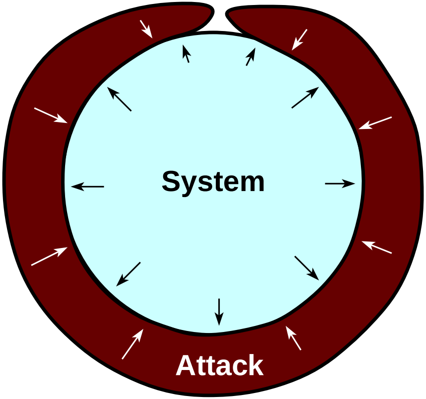
(still a real computer)



If there is a vulnerability,
an attack algorithm will make use of it.

Game of attack vectors

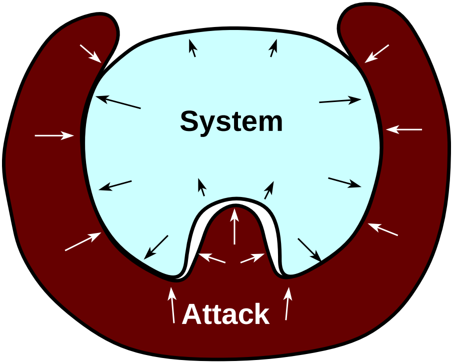
Fortification



System must defend all vectors, Attacker just needs one

Game of attack vectors

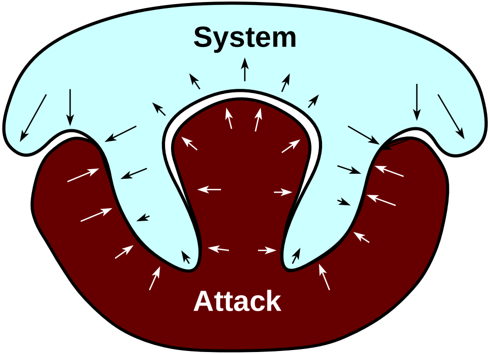
Honeypot



System passively observes Attacker

Game of attack vectores

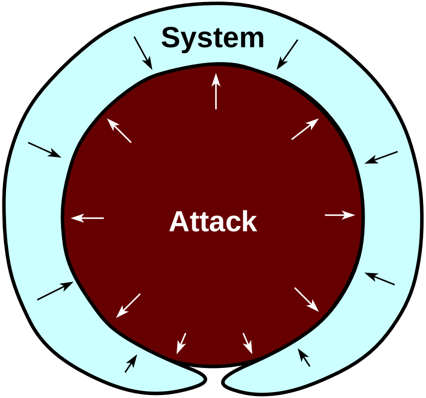
Sampling



System actively queries Attacker

Game of attack vectors

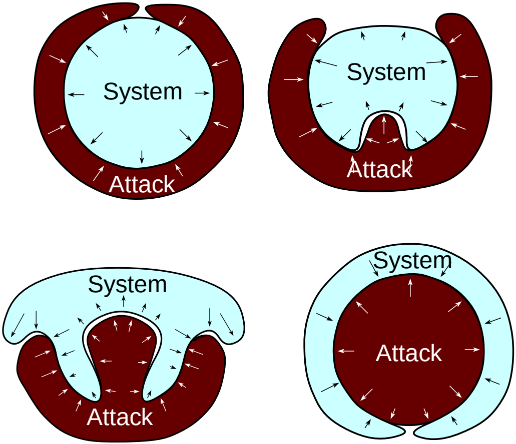
Adaptation



Attacker must defend all markers, System just needs one

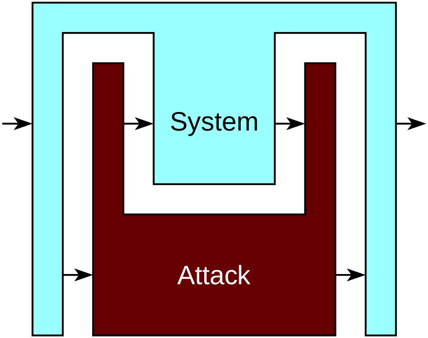
Game of attack vectors

From fortification to adaptation



Adaptive defender: queries the users

(another computer)



If the attacker queries the system
then the system should query the attacker

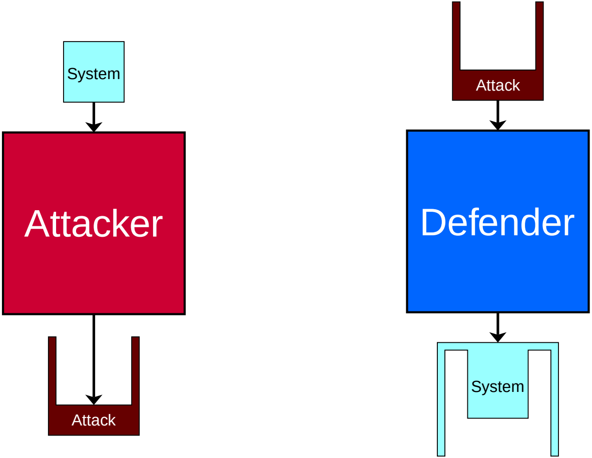
It is good to keep the invaders out. . .

. . . but it is better to bring them in

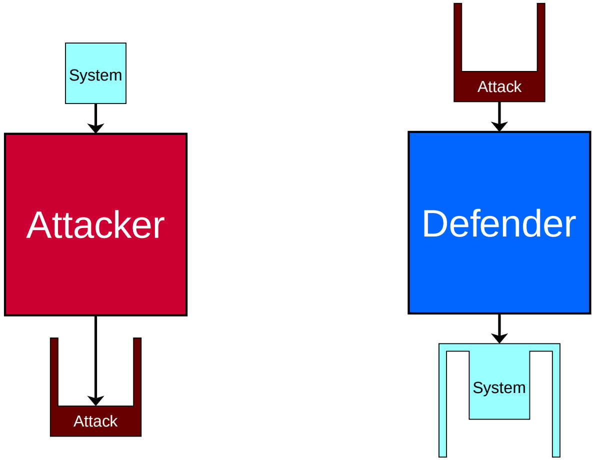
. . . but it is better to bring them in

One-way-programming: adaptive immune response

Arms race for algorithms



Arms race for algorithms



Socratic method: *Answer questions by questions*

Outline

Introduction

Background: Security and obscurity

Idea: Attack models

Approach: Directions

Summary

Summary

New directions in security by obscurity

- improve adaptation of system to attack
  - use **epistemic game theory** in security
  - turn compromise into advantage
    - from fortification to adaptation
- hinder adaptation of attack to system
  - use **algorithmic information theory** in security
  - leverage emergent behaviors
    - emergency as logical complexity

Summary

Obstacles

- complexity of strategies with incomplete information
- incompleteness of theories of logical distance