KTH-AAR CTF

- May 6, 14.00 - 18.00
- Novice-friendly CTF across two Language-Based Security courses
  - KTH (Stockholm)
  - Aarhus
- Teams up to 6 people

Quantitative security

Language-Based Security 2025

aslan@cs.au.dk

Security metrics

- Noninterference is a good theoretical basis, BUT not sufficient in practice
  - (cf. declassifies in programming assignment)
- Quantitative approach:
  - “how much” information is leaked through program execution.
  - If we have such an approach then noninterference is just a special case where “how much == nothing”.

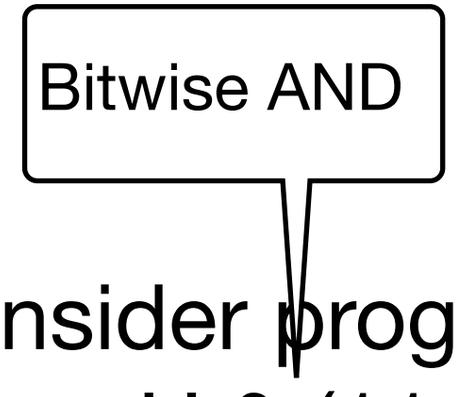
Idea behind quantitative approach

- Given a program. There are
  - some secret input H
  - some public output L
- Adversary observes L, tries to deduce something about H.
- Can we quantify the information content using *numbers*:
  - how much information there is in H?
  - how much of that information about H is leaked to L?
  - how much information about H remains unleaked?
- Intuition
  - “initial uncertainty = information leaked + remaining uncertainty”
- Whatever these numbers are they need to be meaningful:
  - Sanity checks
    - for NI, the leakage should be 0
    - for non-zero case the numbers should have value proportional to their actual security significance
-

The formal setting

- Simple imperative deterministic programming language
  - a priori known distribution of inputs
- We'll look at two definitions:
  - one based on Shannon entropy: see that it satisfies only one of the sanity checks but not the other
  - a definition based on min-entropy
- We will focus only on the *semantic definition* of security, not how to enforce it, but this is important:
  - If the semantic definition is broken, a sound/precise analysis w.r.t. that definition is useless.

Example



- Consider program
  L := H & $(11111)_2$
- This program copies the last 5 bits of H into L
- Suppose H is a 32-bit integer
  - This program leaks 5 out of 32 bits of H

Shannon entropy

- Suppose that X is a random variable
  - example: a coin, a dice, etc
  - let **X** be the set of possible values of X
    - for coin: **X** = {head, tails}
    - for a dice: **X** = {1, … ,6}
  - Write p(x) for probability that r.v. X has value x
- Definition: *entropy* of the r.v. X is a measure of *uncertainty* in X

$$\mathbf{H}(X) = -\sum_{x \in \mathbf{X}} p(x) \cdot \log p(x)$$

(Convention: $0 \cdot \log 0 = 0$)

Example: a coin

- Suppose we have a fair coin
  - p (heads) = 1/2
  - p (tails) = 1/2
- Q: what's the entropy of the fair coin?
  - A: - ((0.5 * $\log_2$ 0.5) + (0.5 * $\log_2$ 0.5))
- Obs: this is independent of the values in **X.**

Conditional entropy

- Suppose we have two r.vs. X and Y:
- Conditional entropy: how much uncertainty there is about X given Y
  - (expected value of the entropies of the conditional distribution)

Conditional random variable

$$\mathbf{H}(X|Y) = \sum_{y \in \mathbf{Y}} p(y) \cdot \mathbf{H}\,(X|Y = y)$$

probability of X = x, given Y = y

$$\text{where} \quad \mathbf{H}(X|Y = y) = -\sum_{x \in \mathbf{X}} p(x|y) \cdot \log\, p(x|y)$$

If X is determined by Y, then $\mathbf{H}\,(X|Y) = 0$

Mutual information

- Amount of information shared between r.vs X and Y
  - **I** (X; Y) = **H** (X) - **H** (X|Y)
  - Turns out to be symmetric: **I** (X;Y) = **I** (Y; X)
- Slogan:
  - “initial uncertainty = information leaked + remaining uncertainty”
- OR:
  - “information leaked = initial uncertainty - remaining uncertainty”
- If H and L are r.vs. define leakage to be **H** (H) - **H** (H | L), or simply **I** (H; L)

Deterministic programs

- L is determined by H, in this case **H** (L|H) = 0
- **I** (H; L)
  = **I** (L; H)
  = **H** (L) - **H** (L | H)
  = **H** (L)
- Suppose H is a uniformly distributed 32-bit int.

Program	$H(\mathtt{H})$	$I(\mathtt{H};\mathtt{L})$	$H(\mathtt{H}\vert\mathtt{L})$
`L := 0`	32	0	32
`L := H`	32	32	0
`L := H & 037`	32	5	27

What about sanity checks? OK for NI (leakage 0), but what about non-zero leakage?

Examples

Suppose H is uniformly distributed 8k bit integer

$0 \leq H \leq 2^{8k}$

$\mathbf{H}(H) = 8k$

Example1

```
if H mod 8 = 0 then
   L := H
else
   L := 1
```

Example2

$L := H \ \& \ 0^{7k-1}1^{k+1}$

Proposal: vulnerability

Definition 1. *Given a random variable* X *with space of possible values* $\mathcal{X}$*, the* vulnerability of X*, denoted* $V(\mathrm{X})$*, is given by*

$$V(\mathrm{X}) = \max_{x \in \mathcal{X}} P[\mathrm{X} = x].$$

Intuition: worst-case probability that adversary could guess the value of X correctly in one try.

m guesses: at most m V(x)

Definition 2. *The* min-entropy of X*, denoted* $H_\infty(\mathrm{X})$*, is given by*

$$H_\infty(\mathrm{X}) = \log \frac{1}{V(\mathrm{X})}.$$

log, because we want a metric in bits ;)

Conditional vulnerability

Definition 3. *Given (jointly distributed) random variables* $\mathtt{X}$ *and* $\mathtt{Y}$*, the* conditional vulnerability $V(\mathtt{X}|\mathtt{Y})$ *is*

$$V(\mathtt{X}|\mathtt{Y}) = \sum_{y \in \mathcal{Y}} P[\mathtt{Y} = y] V(\mathtt{X}|\mathtt{Y} = y)$$

where

$$V(\mathtt{X}|\mathtt{Y} = y) = \max_{x \in \mathcal{X}} P[\mathtt{X} = x|\mathtt{Y} = y].$$

Definition 4. *The* conditional min-entropy $H_\infty(\mathtt{X}|\mathtt{Y})$ *is*

$$H_\infty(\mathtt{X}|\mathtt{Y}) = \log \frac{1}{V(\mathtt{X}|\mathtt{Y})}.$$

Back to the slogan:

- initial uncertainty = $H_\infty(\mathtt{H})$
- remaining uncertainty = $H_\infty(\mathtt{H}|\mathtt{L})$
- information leaked = $H_\infty(\mathtt{H}) - H_\infty(\mathtt{H}|\mathtt{L})$

Theorems

Theorem 1. *If* c *is deterministic and* $\mathtt{H}$ *is uniformly distributed, then the information leaked is* $\log |\mathcal{L}|$.

Theorem 2. *If* c *is deterministic and* $\mathtt{H}$ *is uniformly distributed, then the information leaked,* $\log |\mathcal{L}|$*, is equal to the channel capacity of* c*.*

channel capacity is the max value of H (L)

Dangers of scientific bandwagons

> “While we feel that information theory is indeed a valuable tool in providing fundamental insights into the nature of communication problems and will continue to grow in importance, it is certainly no panacea for the communications engineer or, a fortiori, for anyone else. Seldom do more than a few of nature’s secrets give way at one time.”

C. E. Shannon, “The bandwagon,” IRE Transactions on Information Theory, vol. 2, no. 1, p. 3, **1956**.

Conclusion

- Security definitions are tricky business ;-)
- Quantitative security: measuring leakage of a program (system) in bits
  - Min-entropy is the consensus definition these days
  - Applications: timing leaks, where the timing behavior of the program is the observable output depending on secrets
  - Research trends: synthesis of adaptive attacks for timing leaks from source code analysis