Introduction

Mário S. Alvim
(msalvim@dcc.ufmg.br)

Quantitative Information Flow

DCC-UFMG
(2021/1)

Introduction

- Protecting sensitive information from improper disclosure is fundamental.

  But this goal clearly hasn't been achieved well at all...

- Compromises arise from many causes, including:
  - social (e.g., "phishing" attacks), or
  - technical (flaws in the design and implementation of systems).

- When a compromise is discovered, it's important to fix it.

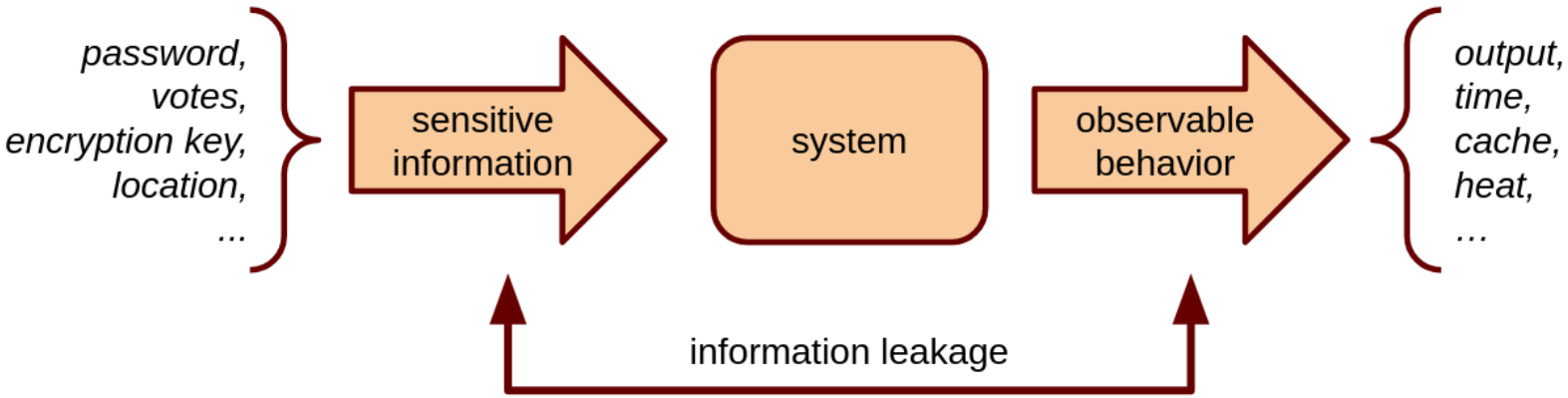
  But just patching flaws as they are discovered is fundamentally limited...
  - At best it corrects all known flaws...
  - ... but we don't know whether there might be other flaws yet to be discovered!

- Secure and trustworthy systems require a more disciplined approach:

We must turn our focus from particular attacks to a true science of security.

Introduction

- In this course we'll study one aspect of computer security: **information flow**.
- Consider a system that:
  - is given some **sensitive information** as **input**; and
  - as it processes it, somehow produces some **publicly observable output**.



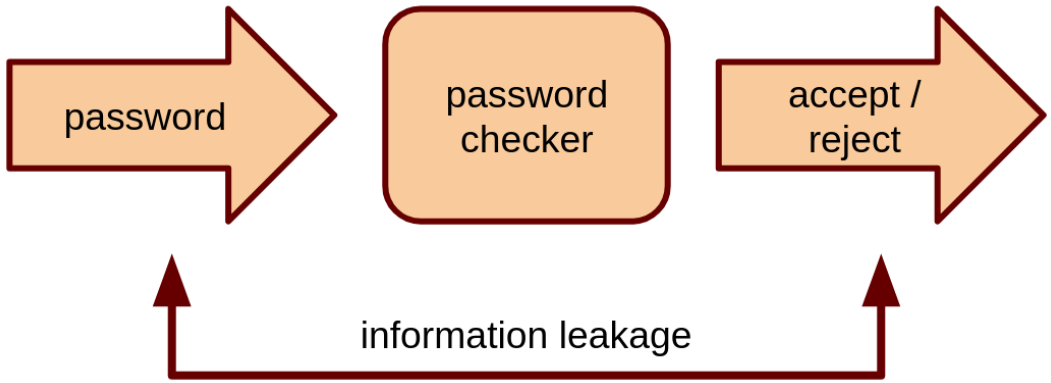
- We might expect that "secure" systems present no leakage...

  But this isn't always possible!

Introduction

- Example 1 **(Password checker)** Consider a password checker that has been given a secret password, which of course it should not leak.

  When a user tries to log in with some guessed password, the checker must reveal whether the guess is correct or not.



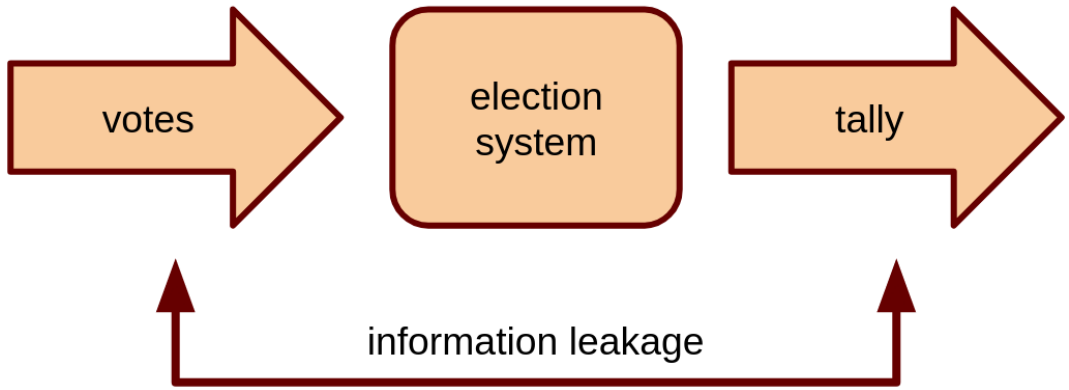
Note that:

- Accepting a correct guess reveals it's the right password.
- Rejecting an incorrect guess leaks the fact that the secret password is different from the guess. ◁

Introduction

- Example 2 **(Election-tallying program)** Consider an election-tallying program that takes as inputs the ballots of a group of voters.

  Typically we would demand that the ballots be kept secret, yet the election system needs to output the result of the election.



  That clearly leaks some information about the secret ballots.

  E.g., in the extreme case of an election that turns out to be unanimous, the tally reveals how everyone voted!

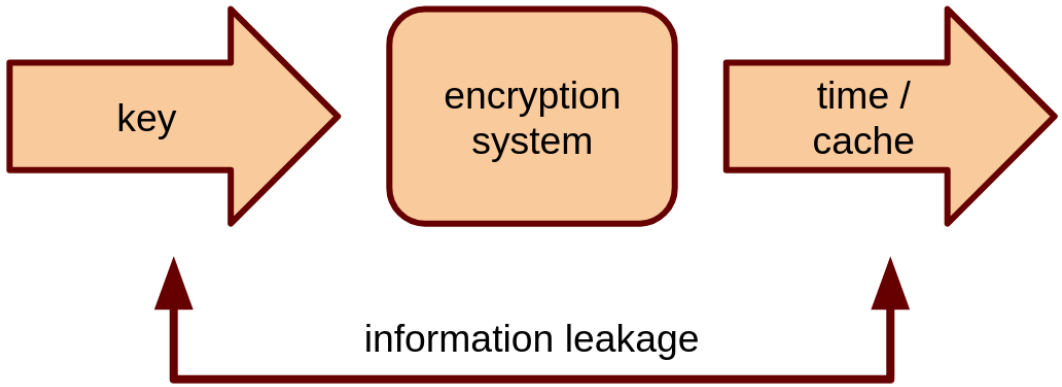
  What then is the best choice for publishing the election results so that the aims of both integrity and privacy are served? ◁

Introduction

- Example 3 (**Side channels**)

  In typical implementations the time required to do an *RSA* decryption varies, depending on the secret key: that can lead to a **timing attack**.

  Similarly, the cache state that results from an *AES* decryption varies, depending on the secret key.



  In both cases, there's leakage from the secret key to system timing or caching behavior, which can lead the attacker to recover the secret key.

  How to determine the effectiveness of some defense employed? ◁

Introduction

- These examples highlight important points:

1. We'd prefer simple mathematical models that exhibit the essential features of the systems while abstracting from irrelevant details.

   Challenge: What's "essential" and what's "irrelevant"?

```
PasswordChecker (guess:  n-bits)
  out := "ACCEPT"
  for i:=1 to n do
    if guess(i) != pwd(i) then
      out := "REJECT"
      return out
  return out
```

2. We shouldn't necessarily be concerned with whether a system leaks sensitive information, but with how much it leaks.

   If the leak is "small", then we may decide it's safe to use the password checker, or the election system.

Introduction

- In this course we cover a theory of **Quantitative Information Flow (QIF)**.

  We aim to explain precisely:

  1. what information leakage is;
  2. how it can be assessed quantitatively, and
  3. how systems can be constructed that satisfy rigorous information-flow guarantees.

- In the rest of this lecture we provide an informal introduction to QIF, which will be formalized properly along this course.

A first discussion of information leakage

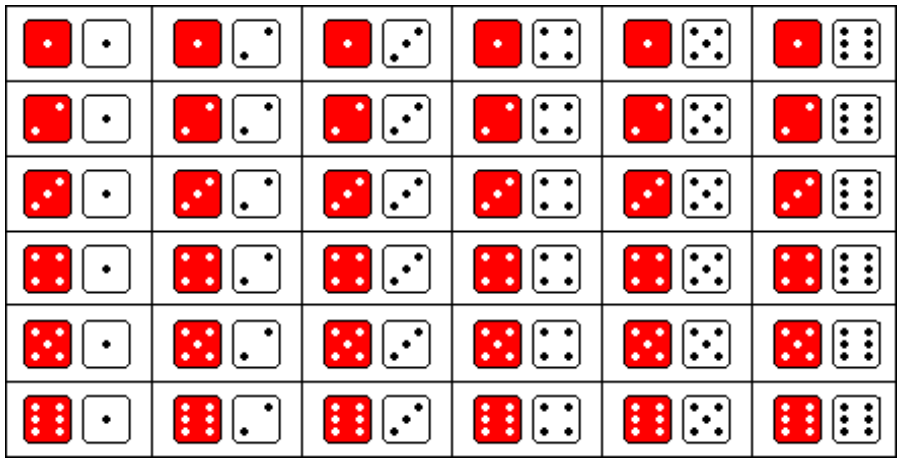
A first discussion of information leakage: Secrets

- Example 4 Suppose that a secret called X is generated by rolling a pair of distinguishable dice, one red and one white.

  (We can think of X as a 2-digit PIN, using digits from 1 to 6.)

The set $\mathcal{X}$ of possible values of X has 36 elements.

We can write each value as a pair (r, w), indicating the result of the red/white dice, respectively.



Saying that

"X is a secret *wrt. an adversary"*

means that the adversary knows only a **probability distribution** π that specifies the probability π_x of each possible value x of X.

A first discussion of information leakage: Secrets

- Example 4 (Continued)

  If we assume that the two dice are fair and independent, then π is uniform:

  $$\pi_x = \pi_{x'}, \qquad \text{for all } x, x' \text{ in } \mathcal{X}.$$

  Notation: We will usually write ϑ for a generic uniform distribution.

  In this example we have:

  $$\vartheta_{(1,1)} = \vartheta_{(1,2)} = \vartheta_{(1,3)} = \cdots = \vartheta_{(6,6)} = 1/36 \quad .$$

  ◁

- We refer to that distribution (be it uniform or not) as a **prior distribution**, as it reflects the adversary's knowledge about X before observing the output of a system.

A first discussion of information leakage: Bayes vulnerability

- Assuming that the adversary's knowledge of X is limited to π, there are many reasonable ways to quantify the "threat" to X.
- For now we focus on a basic measure that we call **Bayes vulnerability**.

  It represents an adversary's maximum probability of guessing the value of X correctly in one try.
- Denoting the Bayes vulnerability of π by $V_1(\pi)$, we then have

$$V_1(\pi) \ := \ \max_{x \in \mathcal{X}} \pi_x \quad .$$

- Example 5 In the dice example, the prior π is the uniform distribution ϑ, so

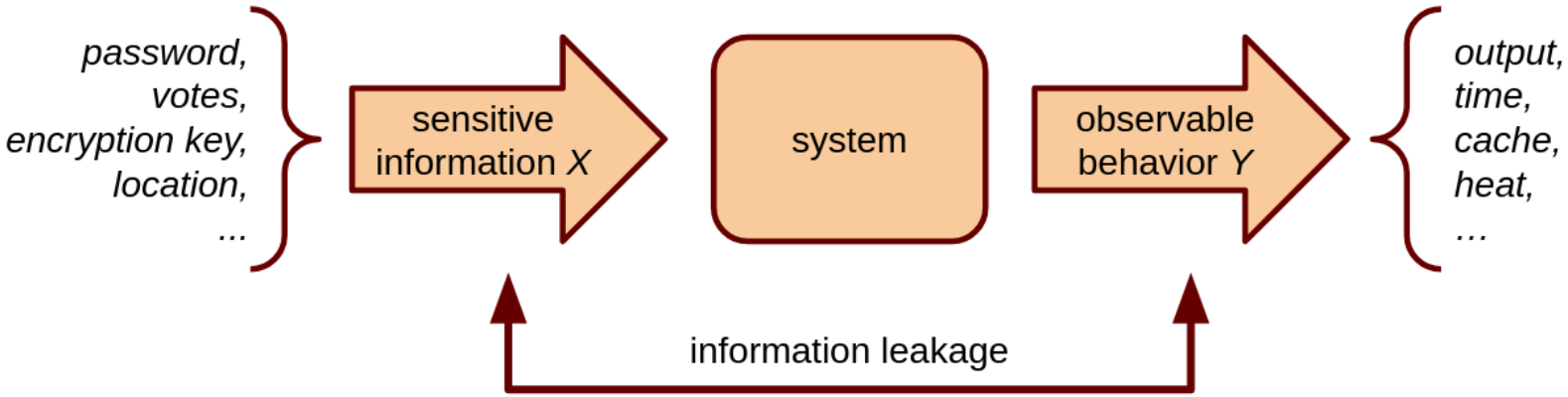
$$V_1(\pi) = V_1(\vartheta) = 1/36.$$

A first discussion of information leakage: Deterministic channels

- Let's think about the information leakage caused by a **system** C that processes a secret X.

  In this course we focus on **channels**, which are systems that:

  - take a secret X as input;
  - produce as sole observable behavior an output Y.



A first discussion of information leakage: Deterministic channels

- Let's start with **deterministic** channels, where each input value x leads to a unique output value

$$y = C(x).$$

- An adversary seeing an output value y learns that:
  - the value of X must be one of the values in $\mathcal{X}$ that are mapped by C to y; and
  - all other values for X are eliminated.

  That conclusion depends on the worst-case assumption, adopted in this course, that the adversary knows how channel C works.

  That's reflected in **Kerckhoffs' Principle**,

  "No security through obscurity",

  and in **Shannon's maxim**,

  "The enemy knows the system".

A first discussion of information leakage: Deterministic channels

- Example 6 Returning to our dice example, we suppose that C takes as input the value (r, w) of X and outputs the sum of the two dice, so that

$$\mathsf{C}(r, w) = r+w.$$

(We can think of C as a malware that leaks the sum of the digits of a PIN.)

- Output set is $\mathcal{Y} = \{2, 3, 4, \ldots, 12\}$.
- Outputs induce a partition of $\mathcal{X}$.
- Each block is one of 11 possible states of knowledge, or "worlds", that an adversary seeing the output of C can end up in.

Y	Possible values of X
2	$\{(1,1)\}$
3	$\{(1,2),(2,1)\}$
4	$\{(1,3),(2,2),(3,1)\}$
5	$\{(1,4),(2,3),(3,2),(4,1)\}$
6	$\{(1,5),(2,4),(3,3),(4,2),(5,1)\}$
7	$\{(1,6),(2,5),(3,4),(4,3),(5,2),(6,1)\}$
8	$\{(2,6),(3,5),(4,4),(5,3),(6,2)\}$
9	$\{(3,6),(4,5),(5,4),(6,3)\}$
10	$\{(4,6),(5,5),(6,4)\}$
11	$\{(5,6),(6,5)\}$
12	$\{(6,6)\}$

A first discussion of information leakage: Deterministic channels

- In deterministic channels, many partitions are possible.
  - At one extreme, the partition might consist of a single block with all of $\mathcal{X}$.

    That happens when the channel gives the same output on all inputs.

    In this case there's no leakage at all.
  - At the other extreme, the blocks in the partition might all be singletons.

    That happens when the channel is a one-to-one function.

    In this case X is leaked completely.
- But in general the partition consists of a number of blocks of varying sizes.

A first discussion of information leakage: Posterior distributions and hyper-distributions

- To better understand the effect of the partition, we must consider probabilities, noting the following.

  1. Each block in the partition gives rise to a posterior distribution on $\mathcal{X}$.

     In our dice example we are assuming a uniform prior distribution ϑ, which means that each of the 11 posterior distributions is also uniform.

  2. Each posterior distribution itself has a probability of occurring, coming from the probability of the output value from which that posterior was deduced.

     That is the probability that the adversary will find herself in that world.

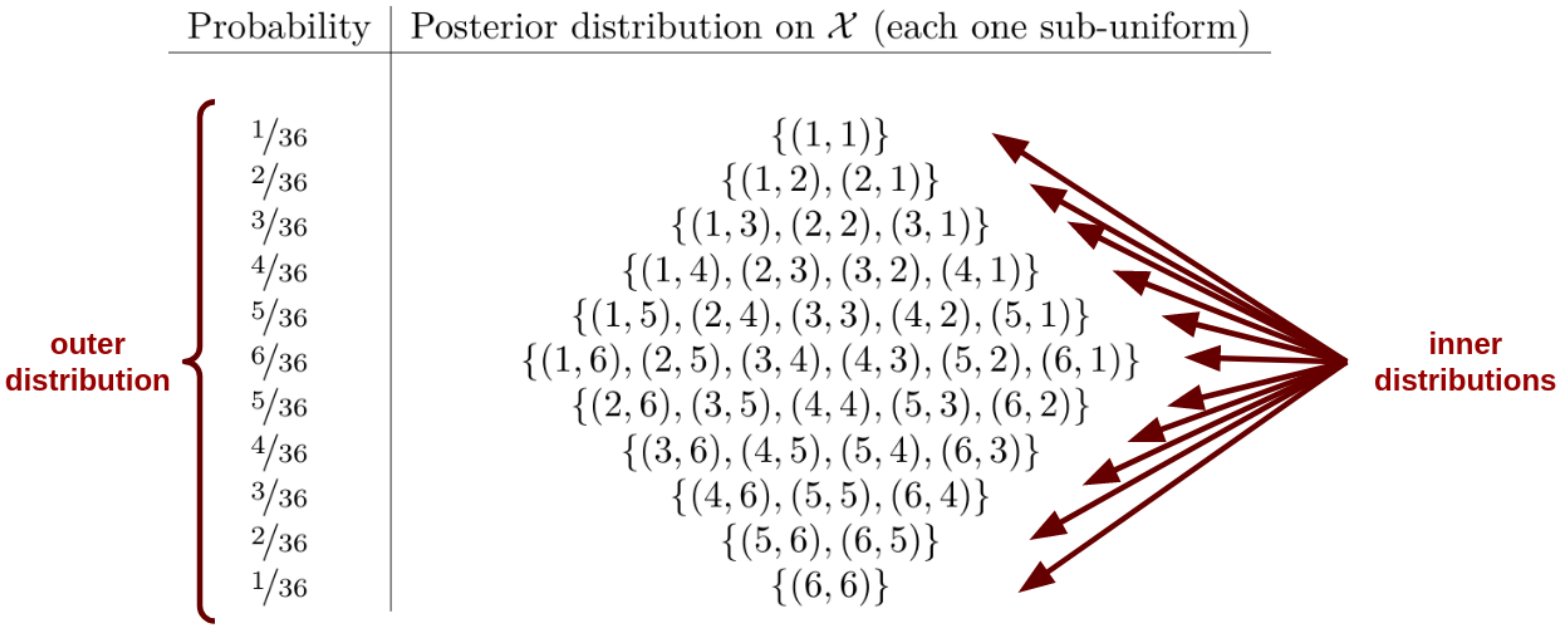
- The result is that channel C maps prior distribution ϑ to a distribution on posterior distributions, called a **hyper-distribution** and denoted by

$$[\vartheta \triangleright \mathsf{C}].$$

A first discussion of information leakage: Posterior distributions and hyper-distributions

- **Example 7** A representation of the hyper-distribution for dice example is:

$[\vartheta \triangleright C]$:



Note that, from the point of view of the adversary:

- The best posteriors (from $Y{=}2$ and $Y{=}12$) are the least probable ($1/36$).
- The worst posterior (from $Y{=}7$) is the most probable ($6/36$). ◁

A first discussion of information leakage: Posterior Bayes vulnerability

- We'll now refer to $V_1(\pi)$ as the **prior Bayes vulnerability**.
- As for **posterior Bayes vulnerability**, there are two approaches:
  - The **dynamic** approach considers the posterior vulnerability after a particular output value y has been observed.
  - The **static** approach considers the channel C as a whole, looking at the entire hyper-distribution $[\pi \triangleright C]$ all at once.
- In this course we'll mostly focus on the static perspective of leakage:
  1. We first calculate the "threat" associated with each of the posterior distributions separately by taking its Bayes vulnerability.
  2. We then combine those numbers into a single value.

     But there's more than way of doing this combination...

A first discussion of information leakage: Posterior Bayes vulnerability

- The first option is the "pessimistic" way, which considers the **maximum** Bayes vulnerability over all the posterior distributions.

- Example 8 Consider again the dice example.

$[\vartheta_{\triangleright} C]$:

Probability	Posterior distribution on $\mathcal{X}$ (each one sub-uniform)
$1/36$	$\{(1,1)\}$
$2/36$	$\{(1,2),(2,1)\}$
$3/36$	$\{(1,3),(2,2),(3,1)\}$
$4/36$	$\{(1,4),(2,3),(3,2),(4,1)\}$
$5/36$	$\{(1,5),(2,4),(3,3),(4,2),(5,1)\}$
$6/36$	$\{(1,6),(2,5),(3,4),(4,3),(5,2),(6,1)\}$
$5/36$	$\{(2,6),(3,5),(4,4),(5,3),(6,2)\}$
$4/36$	$\{(3,6),(4,5),(5,4),(6,3)\}$
$3/36$	$\{(4,6),(5,5),(6,4)\}$
$2/36$	$\{(5,6),(6,5)\}$
$1/36$	$\{(6,6)\}$

The maximum posterior Bayes vulnerability is 1, corresponding to the case when the sum is 2 or 12. ◁

A first discussion of information leakage: Posterior Bayes vulnerability

- In this course we'll mostly adopt a second option: the "moderate" way.

  We'll consider the **expectation** or **average** of posterior vulnerabilities in $[\pi \triangleright C]$, weighted by the outer distribution, denoted by

  $$V_1[\pi \triangleright C].$$

  That definition has the following operational interpretation:

  > *"The probability that a smart adversary will correctly guess the value of secret X, randomly chosen according to π, given the output of C."*

A first discussion of information leakage: Posterior Bayes vulnerability

- **Example 9** Let's consider our dice channel C again.

$[\vartheta \triangleright C]$:

Probability	Posterior distribution on $\mathcal{X}$ (each one sub-uniform)
$1/36$	$\{(1,1)\}$
$2/36$	$\{(1,2),(2,1)\}$
$3/36$	$\{(1,3),(2,2),(3,1)\}$
$4/36$	$\{(1,4),(2,3),(3,2),(4,1)\}$
$5/36$	$\{(1,5),(2,4),(3,3),(4,2),(5,1)\}$
$6/36$	$\{(1,6),(2,5),(3,4),(4,3),(5,2),(6,1)\}$
$5/36$	$\{(2,6),(3,5),(4,4),(5,3),(6,2)\}$
$4/36$	$\{(3,6),(4,5),(5,4),(6,3)\}$
$3/36$	$\{(4,6),(5,5),(6,4)\}$
$2/36$	$\{(5,6),(6,5)\}$
$1/36$	$\{(6,6)\}$

Under a uniform prior π, we can compute the posterior Bayes vulnerability:

$$\begin{aligned} V_1[\pi \triangleright C] &= 1/36 \cdot 1 + 2/36 \cdot 1/2 + 3/36 \cdot 1/3 + 4/36 \cdot 1/4 + 5/36 \cdot 1/5 + 6/36 \cdot 1/6 + \\ &= 11/36 \quad . \end{aligned}$$

A first discussion of information leakage: Posterior Bayes vulnerability

- There's a general property of deterministic channels under uniform priors.

Theorem (1.1)

Let ϑ be a uniform prior distribution on an N-element set $\mathcal{X}$ and let C be a deterministic channel with M possible output values (which implies that $1 \leq M \leq N$). Then $V_1[\vartheta \triangleright C] = M/N$.

Proof. Let the possible output values be $y_1, y_2, \ldots, y_M$, and let the corresponding blocks have sizes $s_1, s_2, \ldots, s_M$.

The Bayes vulnerability given output y_m is $1/s_m$, since the posterior distribution is uniform on the s_m block elements; and the probability of output y_m is s_m/N. Hence we have

$$V_1[\vartheta \triangleright C] \quad = \quad \sum_{m=1}^{M} s_m/N \cdot 1/s_m \quad = \quad \sum_{m=1}^{M} 1/N \quad = \quad M/N \quad .$$

A first discussion of information leakage: Posterior Bayes vulnerability

- The theorem says that the sizes of the blocks don't matter...

  All that matters is the number of possible output values!

- This illustrates how the scientific study of quantitative information flow can yield results that might be found surprising:
  - If we have two deterministic channels C and D taking input X, then to compare the posterior Bayes vulnerabilities under a uniform prior ϑ, it suffices to count the number of possible outputs of C and of D.
  - The "shapes" of the two partitions do not matter!

- Finally, notice that the theorem implies that C never causes Bayes vulnerability to decrease, since we have

$$V_1(\vartheta) \quad = \quad {}^1\!/_N \quad \leq \quad {}^M\!/_N \quad = \quad V_1[\vartheta \triangleright \mathsf{C}].$$

A first discussion of information leakage: Quantifying leakage

- We have quantified vulnerability before and after the operation of channel C.

  We are now ready to quantify the leakage of X caused by C.

- It's natural to do that by comparing:
  - prior Bayes vulnerability $V_1(\pi)$, and
  - posterior Bayes vulnerability $V_1[\pi \triangleright \mathsf{C}]$.
- Comparison can be done by a ratio or a difference, yielding:
  - **Multiplicative Bayes leakage**:

$$\mathcal{L}_1^{\times}(\pi, \mathsf{C}) \quad := \quad \frac{V_1[\pi \triangleright \mathsf{C}]}{V_1(\pi)}.$$

  - **Additive Bayes leakage**:

$$\mathcal{L}_1^{+}(\pi, \mathsf{C}) \quad := \quad V_1[\pi \triangleright \mathsf{C}] - V_1(\pi).$$

A first discussion of information leakage: Quantifying leakage

- Example 10 In our dice channel, we have seen that the prior and posterior Bayes vulnerabilities are:

$$V_1(\vartheta) = 1/36 \qquad \text{and} \qquad V_1[\vartheta \triangleright \mathsf{C}] = 11/36.$$

Hence:

  - The multiplicative Bayes leakage is

$$\mathcal{L}_1^{\times}(\vartheta, \mathsf{C}) = \frac{(11/36)}{(1/36)} = 11$$

  - The additive Bayes leakage is

$$\mathcal{L}_1^{+}(\vartheta, \mathsf{C}) = 11/36 - 1/36 = 5/18$$

◁

A first discussion of information leakage: Quantifying leakage

- Those particular results are again a reflection of a general property, which follows immediately from Theorem 1.1.

Corollary (1.2)

Let ϑ be a uniform prior distribution on an N-element set $\mathcal{X}$ and let C be a deterministic channel with M possible output values.

Then:

- $\mathcal{L}_1^{\times}(\vartheta, \mathsf{C}) = M$, *and*
- $\mathcal{L}_1^{+}(\vartheta, \mathsf{C}) = {}^{(M-1)}/_{N}$.

- Those results for the Bayes leakage of a deterministic channel under a uniform prior are especially useful, as they say that to calculate leakage we just need to count the number of possible output values.

A first discussion of information leakage: Quantifying leakage

- Example 11 Consider a (deterministic) channel D that (like our dice channel C) takes as input the value (r, w) of X, but instead returns the product of the two dice:

$$\mathrm{D}(r, w) = r \cdot w.$$

If we wish to compute the Bayes leakage of D, then by Corollary 1.2 it suffices to count the number of possible output values, which are 18:

$$1, 2, 3, 4, 5, 6, 8, 9, 10, 12, 15, 16, 18, 20, 24, 25, 30, 36.$$

- The corresponding multiplicative leakage is

$$\mathcal{L}_1^{\times}(\vartheta, \mathrm{D}) = 18$$

- The corresponding additive leakage is:

$$\mathcal{L}_1^{+}(\vartheta, \mathrm{D}) = {}^{17}/_{36}$$

Looking ahead

Looking ahead

- Our discussion of information leakage so far introduced some important concepts of QIF, but raises questions:
- Question 1: Although Bayes vulnerability clearly measures a basic security concern, what if an adversary's goals are not to guess the secret correctly in one try?

  E.g., maybe the adversary:

  - can succeed by guessing the secret within three tries; or
  - can succeed by guessing the secret only approximately; or
  - is penalized for making an incorrect guess...

  How can we address the great multiplicity of possible operational scenarios?

  That question will be addressed when we present the family of g-**vulnerability** measures $V_g(\pi)$ (Chapters 2-3).

Looking ahead

- Question 2: Our dice examples C and D are deterministic and we studied them under the assumption that the prior distribution was uniform.
  - Can we analyze channels whose behavior is probabilistic?
  - Can we deal with a non-uniform prior distribution π (as would result from biased dice)?

In Chapter 4 we'll see that the view of channels as mappings from prior distributions to hyper-distributions remains valid in that more general setting.

In Chapter 5 we'll see that we can obtain **posterior g-vulnerability** and g-**leakage** for an arbitrary gain function g.

Looking ahead

- Question 3: But this richness also raises questions...

  Recall that we found that the multiplicative Bayes leakage of dice channel C is 11, while that of dice channel D is 18.

  - Does that mean that C is more secure than D in general, so replacing D with C can only decrease leakage?
  - What if the prior distribution π were not uniform?
  - What if we measured leakage using some g-vulnerability other than Bayes vulnerability?

  Moreover, we can ask:

  - Would the leakage ordering between C and D always stay the same?

  It turns out that the answer is no!

  We discuss the robustness of leakage analyses in Chapters 6-10.

Looking ahead

- Question 4: A final question is how QIF can be applied in real scenarios?

  In Chapter 18 we see will the application of QIF to the analysis of the Crowds anonimity protocol, which is the basis for current protocols like Tor.