Measuring Information Leakage using Generalized Gain Functions

Mário S. Alvim, Kostas Chatzikokolakis, Catuscia Palamidessi, and Geoffrey Smith

Presented by Tegan Brennan

October 20, 2016

Motivation

Motivation

Protecting confidentiality of secret information is a fundamental issue.

However, non-interference is often too strong a condition $\rightarrow$ interest in theories that allow information leakage to be quantified so that a small amount of leakage is tolerable.

Min Entropy Leakage

Leakage measure based on how much a channel increases the vulnerability of a secret to being correctly guessed in one try.

Question - how applicable in this across various scenarios?

Min Entropy Leakage

Leakage measure based on how much a channel increases the vulnerability of a secret to being correctly guessed in one try.

Question - how applicable in this across various scenarios?

- What if an adversary benefits by guessing part of the secret?

Min Entropy Leakage

Leakage measure based on how much a channel increases the vulnerability of a secret to being correctly guessed in one try.

Question - how applicable in this across various scenarios?

- What if an adversary benefits by guessing part of the secret?
- Guessing the secret approximately?

Min Entropy Leakage

Leakage measure based on how much a channel increases the vulnerability of a secret to being correctly guessed in one try.

Question - how applicable in this across various scenarios?

- What if an adversary benefits by guessing part of the secret?
- Guessing the secret approximately?
- Is allowed multiple guesses?

Min Entropy Leakage

Leakage measure based on how much a channel increases the vulnerability of a secret to being correctly guessed in one try.

Question - how applicable in this across various scenarios?

- What if an adversary benefits by guessing part of the secret?
- Guessing the secret approximately?
- Is allowed multiple guesses?
- Penalized for guessing incorrectly?

Motivating Example

X is an array containing 10-bit uniformly distributed passwords for 1000 users.

Consider the channel:

$$u \overset{?}{\leftarrow} \{0 \ldots 999\}$$

$$Y = (u, X[u])$$

Some user's password is always leaked!! Would this threat be captured using min entropy?

Motivating Example

Using X as the secret, we compute the prior vulnerability, posterior vulnerability and min entropy leakage:

- $V(\pi) = 1/2^{10000}$

Motivating Example

Using X as the secret, we compute the prior vulnerability, posterior vulnerability and min entropy leakage:

- $V(\pi) = 1/2^{10000}$
- $V(\pi, C) = 1/2^{9990}$

Motivating Example

Using X as the secret, we compute the prior vulnerability, posterior vulnerability and min entropy leakage:

- $V(\pi) = 1/2^{10000}$
- $V(\pi, C) = 1/2^{9990}$
- $\mathcal{L} = \log \frac{2^{-9990}}{2^{-10000}} = 10$ bits

Motivating Example

Using any specific user's password as the secret, we compute the prior vulnerability, posterior vulnerability and min entropy leakage:

- $V(\pi) = 1/2^{10}$

Motivating Example

Using any specific user's password as the secret, we compute the prior vulnerability, posterior vulnerability and min entropy leakage:

- $V(\pi) = 1/2^{10}$
- $V(\pi, C) = \frac{1}{1000} * 1 + \frac{999}{1000}\frac{1}{2^{10}} \approx .00198$

Motivating Example

Using any specific user's password as the secret, we compute the prior vulnerability, posterior vulnerability and min entropy leakage:

- $V(\pi) = 1/2^{10}$
- $V(\pi, C) = \frac{1}{1000} * 1 + \frac{999}{1000}\frac{1}{2^{10}} \approx .00198$
- $\mathcal{L} \approx \log \frac{.00198}{2^{-10}} \approx 1.106$ bits

Motivating Example

Using any specific user's password as the secret, we compute the prior vulnerability, posterior vulnerability and min entropy leakage:

- $V(\pi) = 1/2^{10}$
- $V(\pi, C) = \frac{1}{1000} * 1 + \frac{999}{1000}\frac{1}{2^{10}} \approx .00198$
- $\mathcal{L} \approx \log \frac{.00198}{2^{-10}} \approx 1.106$ bits

Do these results capture the vulnerability of this channel?

Proposed Solution

- Introduce a generalization of min-entropy leakage, *g-leakage*.
- Parametrize leakage by a gain function that models the benefit an adversary gets by making a guess.
- Goal - model a wide range of scenarios.

Preliminaries

Preliminaries

Definition

Channel A channel is a triple $(\mathcal{X}, \mathcal{Y}, C)$, where $\mathcal{X}$ is a finite set of secret input values, $\mathcal{Y}$ a finite set of observable output values and C is an $|\mathcal{X}| \times |\mathcal{Y}|$ matrix where $C[x, y]$ is the probability of getting output y when the input is x.

- Rows sum to 1
- Each entry is between 0 and 1

Vulnerability

Definition

Given a prior distribution π distribution on $\mathcal{X}$ and channel C, the prior vulnerability is

$$V(\pi) = \max_{x \in X} \pi[x]$$

and the posterior vulnerability is

$$\begin{aligned} V(\pi, C) &= \sum_{y \in Y} \max_{x \in X} \pi[x] C[x, y] \\ &= \sum_{y \in Y} p(y) V(p_{X|y}) \end{aligned}$$

Entropy

Definition

$$H_\infty(\pi) = -\log V(\pi)$$
$$H_\infty(\pi, C) = -\log V(\pi, C)$$

Entropy is a measure of bits of uncertainty.

Entropy

Note this this is not Shannon entropy, which measures the average unpredictability of the output.

Why don't we use Shannon entropy? Because it's operational significance can be quite weak:

$$\pi = (\frac{1}{2}, 2^{-1000}, 2^{-1000}, \ldots, 2^{-1000})$$

Here the Shannon entropy is 500.5 bits, but the adversary can correctly guess the secret half the time.

Leakage

Definition

$$\mathcal{L}(\pi, C) = H_\infty(\pi) - H_\infty(\pi, C) = log\,\frac{V(\pi, C)}{V(\pi)}$$

Leakage is the amount by which C decreases the uncertainty about the secret.

Min Capacity

Definition

$$\mathcal{ML}(C) = \sup_{\pi} \mathcal{L}(\pi, C)$$

Min-capacity is the maximum min-entropy leakage over all priors.
Can be thought of as a worst-case leakage of C.

Gain Functions

Gain Functions

Min entropy operates under the assumption that the adversary only benefits by guessing the exact value of the secret.

Generalize min entropy leakage by introducing gain functions to model the operational scenario.

Gain Functions

Definition

Given a set $\mathcal{X}$ of possible secrets and a set $\mathcal{W}$ of allowable guesses, a gain function specifies the gain that the adversary gets by choosing $w \in \mathcal{W}$ when the secret is $x \in \mathcal{X}$.

$$g : \mathcal{W} \times X \to [0, 1]$$

Note that $\mathcal{W}$ does not have to be $\mathcal{X}$.

Gain Functions

Example: The identity gain function $g_{id} : \mathcal{X} \times \mathcal{X} \rightarrow [0, 1]$ is given by

$$g_{id}(w, x) = \begin{cases} 1 & w = x \\ 0 & w \neq x \end{cases}$$

Gain Functions

Definition

Given a gain function g and a prior π, the prior g-vulnerability is

$$V_g(\pi) = \max_{w \in W} \sum_{x \in X} \pi[x] g(w, x)$$

Intuitive is that adversary should make a guess w that maximizes the expected gain.

Gain Functions

Definition

Given a gain function g, prior π, and a channel C the posterior g-vulnerability is

$$\begin{aligned} V_g(\pi, C) &= \sum_{y \in Y} \max_{w \in W} \sum_{x \in X} \pi[x] C[x, y] g(w, x) \\ &= \sum_{y \in Y} \max_{w \in W} \sum_{x \in X} p(x, y) g(w, x) \\ &= \sum_{y \in Y} \max_{w \in W} p(y) \sum_{x \in X} p(x|y) g(w, x) \\ &= \sum_{y \in Y} p(y) V_g(p_{X|y}) \end{aligned}$$

The posterior g-vulnerability is the weighted average of the g-vulnerabilities of the posterior distributions $p_{X|y}$

Gain Functions

Definition

$$
\begin{aligned}
H_g(\pi) &= -\log V_g(\pi) \\
H_g(\pi, C) &= -\log V_g(\pi, C) \\
\mathcal{L}_g(\pi, C) &= H_g(\pi) - H_g(\pi, C) = \log \frac{V_g(\pi, C)}{V_g(\pi)} \\
\mathcal{ML}(C) &= \sup \mathcal{L}_g(\pi, C)
\end{aligned}
$$

Gain Functions

Given these definitions, we can make the following observation.

Proposition

Vulnerability under g_{id} coincides with vulnerability.

Proof.

For any w, $\sum_{X} \pi[x] g_{id}(w, x) = \pi[w]$. Hence

$V_{g_{id}}(\pi) = \max_w \pi[w] = V(\pi)$ $\square$

Examples of Gain Functions

Distance-based gain functions

Given a metric d on $\mathcal{X}$, first divide all distances by the maximum value of d to obtain a normalized metric, $\bar{d}$.

Then the gain function g_d can be defined

$$g_d(w, x) = 1 - \bar{d}(w, x)$$

The gain is based on the distance between the guess and the secret. Allows us to model the case where guessing the secret approximately benefits the adversary.

Binary Gain Functions

The family of gain functions that return either 0 or 1 are called binary gain functions.

In this case, each guess corresponds to the subset of $\mathcal{X}$ for which that guess gives 1. This means that we can use think of the subsets themselves as guesses.

Definition

Given $\mathcal{W} \subseteq 2^{\mathcal{X}}$, $\mathcal{W}$ nonempty, the binary gain function g_W is

$$g_{\mathcal{W}}(W, x) = \begin{cases} 1 & \text{if } x \in W \\ 0 & \text{otherwise} \end{cases}$$

Binary Gain Functions

Different choices for W can lead to interesting gain functions.

Examples: $\mathcal{W} = \{W, X \backslash W\}$,
$\mathcal{W} = X \backslash \sim$,
$\mathcal{W}_k = \{W \in 2^X | \ |W| \leq k\}$

Return to Motivating Example

Recall the channel:

$$u \xleftarrow{?} \{0 \ldots 999\}$$

$$Y = (u, X[u])$$

Return to Motivating Example

The intuition is that the adversary just wants to guess some user's password with no preference as to whose.

Let

$$\mathcal{W} = \{(u, x) | \ 0 \leq u \leq 999 \text{ and } 0 \leq x \leq 1023\}$$

and define

$$g((u, x), X) = \begin{cases} 1 & \text{if } X[u] = x \\ 0 & \text{otherwise} \end{cases}$$

Return to Motivating Examples

$$V_g(\pi) = \max_{w \in W} \sum_{x \in X} \pi[x] g(w, x) = 2^{-10}$$

$$V_g(\pi, C) = 1$$

$$\mathcal{L}(\pi, C) = \log \frac{V_g(\pi, C)}{V_g(\pi)} = 10$$

So we get 10 bits again! But is the meaning any different?

Return to Motivating Example

Converting to entropy,

$$H_g(\pi) = 10 \qquad\qquad H_{g_{id}}(\pi) = 10000$$

The channel leaks 10 out of 10 bits of information under g as compared with 10 out of 10000 under g_{id}.

More accurately models the threat to a structured secret

One More Example

Consider these two channels:

$$\textbf{if } (X \,\%\, 8 == \ 0) \ Y = \ X; \textbf{else } \ Y = 1$$

$$Z = X|07$$

One More Example

Consider these two channels:

$$\textbf{if } (X \,\%\, 8 == 0)\ Y = X; \textbf{else } Y = 1$$

$$Z = X|07$$

Both channels have a min-entropy leakage of 61 bits.

One More Example

Consider these two channels:

$$\textbf{if } (X \,\%\, 8 == 0)\ Y = X; \textbf{else } Y = 1$$

$$Z = X | 07$$

Both channels have a min-entropy leakage of 61 bits.

They can be distinguished by gain functions!

Properties of g-leakage

Comparing min-entropy leakage and g-leakage

Observation No general relation between min-entropy leakage and g-leakage holds. Each may be greater than the other.

Comparing min-entropy leakage and g-leakage

Observation No general relation between min-entropy leakage and g-leakage holds. Each may be greater than the other.

Theorem

For any channel C and gain function g, $\mathcal{ML}_g(C) \leq \mathcal{ML}(C)$

Comparing min-entropy leakage and g-leakage

Observation No general relation between min-entropy leakage and g-leakage holds. Each may be greater than the other.

Theorem

For any channel C and gain function g, $\mathcal{ML}_g(C) \leq \mathcal{ML}(C)$

Min-capacity is an upper bound on g-capacity for *every* gain function g.

Comparing min-entropy leakage and g-leakage

This means that if the min-capacity of C is small, then the leakage under any gain function and under any prior is also small.

However, g can affect the prior vulnerability..... Leakage bounds only address the conservation of confidentiality.

Corollary: The capacity of C under the k-tries scenario is no greater than under the 1-try scenario.

Calculating G-capacity

Min-capacity is always realized on a uniform prior and hence easy to calculate.

The same does not hold for g-capacity.

Cited as an area for future study.

Comparing Channels

Comparing Channels

Say we have two channel C_1 and C_2 with the same input space $\mathcal{X}$.

An interesting question to ask is whether the leakage of C_1 is less than or equal to that of C_2 on every prior.

Definition
Given C_1 from $\mathcal{X}$ to $\mathcal{Z}$ and C_2 from $\mathcal{X}$ to $\mathcal{Y}$ and a leakage measure m, we write $C_1 \leq_m C_2$ if the m-leakage of C_1 never exceeds that of C_2 for any prior.

Comparing Channels

How does this ordering depend on m?

Comparing Channels

A deterministic channel C from $\mathcal{X}$ to $\mathcal{Y}$ induces a partition on $\mathcal{X}$. x_1 and x_2 are in the same partition iff they map to the same output ($C(x_1) = C(x_2)$)

We can order these equivalence relations by partial refinement!

Lattice of Information

Definition

Partial Refinement Given deterministic channels C_1 and C_2, write $C_1 \sqsubseteq C_2$ if the partition of C_1 is refined by the partition of C_2, meaning that each equivalence class of C_2 is contained within some equivalence class of C_1.

For deterministic channels, $\leq_m$ coincides with $\sqsubseteq$ for Shannon, min-entropy and guessing entropy!

This means that $C_1 \sqsubseteq C_2$ iff C_1 never leaks more than C_2 on any prior under any of the usual measures.

Comparing Probabilistic Channels

Can this be generalized to probabilistic channels?

Theorem

Let C_1 from $\mathcal{X}$ to $\mathcal{Z}$ and C_2 from $\mathcal{X}$ to $\mathcal{Y}$ be deterministic channels. Then $C_1 \sqsubseteq C_2$ iff there exists a deterministic channel C_3 from $\mathcal{Y}$ to $\mathcal{Z}$ such that $C_1 = C_2 C_3$

Proof.

Assume $C_1 = C_2 C_3$. Then $C_2(x_1) = C_2(x_2)$ implies that $C_1(x_1) = C_3(C_2(x_1)) = C_3(C_2(x_2)) = C_1(x_2)$. Conversely, assume $C_1 \sqsubseteq C_2$. For every $y \in \mathcal{Y}$, C_1 maps all $x \in C_2^{-1}(y)$ to the same value, say z_y. Define C_3 to map each $y \in \mathcal{Y}$ to z_y. $\square$

Comparing Probabilistic Channels

Can we general partition refinement to probabilistic channels?

Definition

Given C_1 from $\mathcal{X}$ to $\mathcal{Z}$ and C_2 from $\mathcal{X}$ to $\mathcal{Y}$, we say $C_1 \sqsubseteq_0 C_2$ (C_1 is composition refined by C_2) if there exists C_3 from $\mathcal{Y}$ to $\mathcal{Z}$ such that $C_1 = C_2 C_3$

On deterministic channels, $\sqsubseteq_0$ coincides with $\sqsubseteq$

Theorem

If $C_1 \sqsubseteq_0 C_2$*, then* $C_1 \leq_G C_2$*.*

The converse, if $C_1 \leq_G C_2$, then $C_1 \sqsubseteq_0 C_2$, is conjectured. (later resolved)

So we have a partial order on probabilistic channels, with both structural and leakage-testing significance.

Summary

Introduce the idea of gain functions, which allow us to model operational scenarios more precisely.

Give some nice results about how channels can be ordering based on their leakage.