**Ministry of Higher Education and Scientific Research**
**University of Manouba**
**Ecole Nationale des Sciences de l'Informatique**


Mastère Ingénierie des Composants et
Informatique Structurelle: ICIS

## Master Project Report :

# Formal Analysis of Information Flow Using Min-Entropy and Belief Min-Entropy

Prepared at the Hardware Verification Group,
Department of ECE, Concordia University,
Montreal, Quebec, Canada.


Ghassen HELALI

**Supervisors :** Dr. Narjes BELLAMINE - Laboratoire RIADI
Dr. Sofiène TAHAR - Concordia University

# Abstract

Quantitative theories in information flow is becoming nowadays very important in the area of information system security. It is so indispensable in different fields such as secure information flow, anonymity protocols and side-channel analysis. In fact, there is a growing interest in applying these theories in electronic communication, auctioning, voting and payment.

The consensus of quantitative information flow was introduced under the context of *Shannon entropy* and *mutual information*. The main goal of quantitative information flow is to compute the bounds of the threat that a secret information is leaked due to an external attack.

Our major focus in this work is to model the risk that the secret is correctly guessed in *one* try. Considering this model, we argue that the proposed consensus based on *Shannon entropy* failed to give good security guarantees; it sometimes leads to a confusion, this was mentioned by G. Smith, where the problem is that a random variable with high vulnerability to be guessed can have a large Shannon entropy.

We propose to use *min-entropy* and *belief-min-entropy* as better alternatives. The latter one is taking into account the attackers' extra knowledge. Both of these notions will be used in order to model and analyze the information leakage in deterministic and probabilistic systems. We will conduct our work in the core of the Higher-Order-Logic Theorem Proving in which we are going to formalize the new concepts previously presented. We will then apply our theory to analyze the information behavior in a cascade of channels. We prove that the leakage of two cascade channels can not exceed the leakage of the first channel.

**Key Words:** *Information Flow, Security Systems, Min-Entropy, Belief-Min-Entropy, Information Theory, Quantitative and Probabilistic Models, Vulnerability.*

# Résumé

La notion de théories quantitatives dans le flux d'information est de plus fréquente, elle a une grande importance dans le domaine de la sécurité des systèmes d'information. Elle est donc indispensable dans différents domaines tels que le flux d'informations sécurisé, protocoles anonymat et l'analyse des canaux litéraux. En fait, il y a un intérêt croissant pour l'application de ces théories dans la communication électronique, vente aux enchères, les systèmes de vote et le paiement en ligne.
Le flux quantitatif d'information a été introduit dans le contexte de *l'Entropie de Shanon* et *l'Information Mutuelle*. L'objectif principal de circulation de l'information quantitative est de calculer les limites de la menace qu'une information secrète soit révélée.
Notre objectif majeur dans ce travail est de modéliser le risque que le secret est bien deviné à premier coup. Considérant ce modèle, nous soutenons que le consensus proposé, basée sur *l'entropie de Shanon*, omis de donner quelques bonnes garanties de sécurité, il conduit parfois à des confusions, nous pouvons le voir dans l'exemple proposé dans, où le problème est qu'une variable aléatoire avec une forte vulnérabilité à deviner peut avoir une grande entropie de Shanon.
Nous allons donc utiliser de meilleures alternatives au lieu des traditionnelles proposées par Shanon, l'une basée sur l'entropie Rényi, qui est *l'entropie minimale* et l'autre est en tenant compte des croyances supplémentaires des attaquants, ce qui est *l'entropie minimale à croyance*. Ces deux notions seront utilisées afin de modéliser les fuites d'informations et de les analyser pour les systèmes déterministes ainsi que ceux probabilistes.
**Mots clés:** *Flux d'Information, les systèmes de sécurité, Entropie Minimale, Entropie Minimale à Croyance, Théorie de l'Information, Modèles Quantitatifs et Probabilistes, Vulnérabilité.*

# Acknowledgements

First of all, I would like to thank Dr. Sofiene Tahar for renewing his confidence on me and offering me the opportunity to continue my master project inside the Hardware Verification Group, Department of Electrical and Computer Engineering, Concordia University, Montreal, Quebec, Canada. I am also grateful regarding his helpful advices and remarks. Then, I would like to express my sincere thanks to Dr. Narjes Bellamine for her support, her remarks and advices were so pertinent. And finally, I would like to thank all my colleagues of the HVG, for their grateful help, support, discussions and time. And finally, special thanks go to Tarek Mhammdi, who was really taking care of me since my first steps in HOL. He was very helpful and always supporting me by sharing some interesting ideas and especially better ways to deal with HOL problems. Briefly, he is more than a colleague ...

To my parents, and those who
always support me...

# Table of Contents

# List of Figures

# Acronyms

| | |
|---|---|
| **HOL** | Higher-Order Logic |
| **PDF** | Probability Density Function |
| **CDF** | Cumulative Distribution Function |
| **PMF** | Probability Mass Function |
| **ML** | Meta Language |
| **RV** | Random Variable |
| **CPU** | Central Processing Unit |
| **MAX** | Maximum |
| **MIN** | Minimum |

# Chapter 1

# Introduction

## 1.1 Motivation

In critical systems, protecting the confidentiality of sensitive information is one of the most fundamental security issues. It becomes increasingly important in many fields such as communication, auction, online merchant services and voting. One of the classical way is to try to enforce noninterference, this approach says that low outputs and high inputs are independent; i.e. seeing the low output, the system should reveal nothing about the high input.

Unfortunately, avoiding the interference is not always easy, because sometimes we have to reveal information that depends on the high inputs. In a password checker for example, we have to reject an incorrect password, but this reveals information about what the secret password is not. One approach to relaxing noninterference is to develop a quantitative theory of information flow that lets us talk about "how much" information is being leaked.

For that reason, many protocols for security and protection of the confidentiality have been proposed. Recently, all the frameworks aiming to analyze and verify these kinds of protocols have been designed based on probabilistic behaviors and approaches. In fact, the data to be protected often range in real domains which means that they are characterized by statistical properties. These protocols often use randomised strategies to put out the relation between the information to be protected and the observable outcomes.

To illustrate this phenomena, we can cite DCNets (Dining Cryptographers) [2], Crowds [17], Onion Routing [20], and Freenet [13] that are protocols using the principle of hiding the secret information.

From the formal point of view, the degree of protection is the inverse of the leakage, i.e. the quantity of the secret information that can be revealed from the observable. Recent techniques of information hiding using possibilistic approaches have been replaced by nondeterminism. Some examples of these approaches are those based on epistemic logic [8], which is a logic of knowledge and belief, and on process algebra, also called process calculi [18] which is a family of approaches for formally modelling concurrent systems. It has recently been revealed that the possibilistic view is too abusive, it uses to consider systems of different degrees of protection equivalent. For that reason, using the probabilistic approaches is therefore more appropriate due to their high level of protection. These techniques express the property of not revealing "quantitative" information about the secrets. In order to express the degree of protection in a quantitative way, we have to go through the notions of *Information Theory* and *Statistics* which are going to be detailed in the next chapters.

## 1.2 Problem Description

The main problem in this project is how to quantify the information flow and analyze it in order to minimize or even avoid, when it is possible, the information leakage and then decrease the vulnerability of the system. In other words, having two distributions, the higher input and the low output, we try to quantify the system uncertainty with an appropriate model that offer a better way to evaluate the security of the confidential data of the secret input.

The model proposed should cover the different type of programs, both deterministic and probabilistic which is the more common. Another issue that we should take care of is the behavior of the program when an adversary's extra knowledge about the input is introduced. That means the attacker has an initial "belief" about the secret information in addition to the observable output. The challenge is to decrease the vulnerability of critical systems, in which a confidential information is correctly guessed in one try.

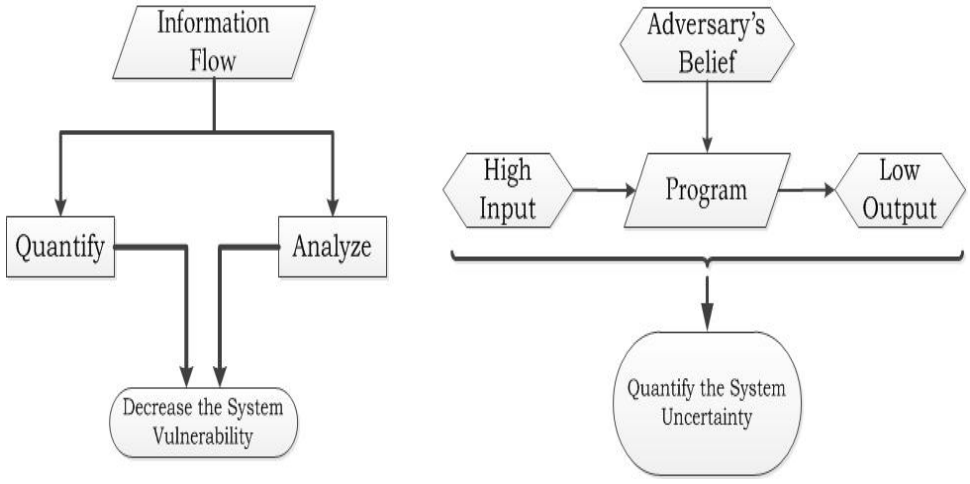


Figure 1.1: Problem Description

## 1.3 State of the Art

In this section we are going to give an overview about some techniques that could be used in the analysis of probabilistic systems and especially, the information systems. We will show how suitable each technique is in relation to our problem. Then we will present some related work that have been conducted in the area formalization of information theory.

### 1.3.1 Probabilistic Analysis Techniques

Due to the large domain of probability, many researchers around the world are trying to improve the quality of computer based probabilistic analysis. In the information theory, the notion probability theory is strongly recommended since all the quantities of information are measured and analyzed using probabilities. The challenge within the probabilistic analysis is to afford a framework that ensure precise and accurate analysis methods, can be used to analyze a variety of problems, and of course can be used easily. So, we provide a brief account of the state of-the-art probabilistic analysis approaches that have been used in the analysis of critical systems.

### Simulation

Today, one of the major technique used is simulation [5] which is the most commonly used computer based probabilistic analysis technique. Simulation is an important feature in engineering systems or any system that involves many processes. For example in electrical engineering, delay lines may be used to simulate propagation delay and phase shift caused by an actual transmission line.

Computer simulation has become a useful part of modeling many natural systems as well as in engineering to gain insight into the operation of those systems. A good example of the usefulness of using computer simulation can be found in the field of network traffic simulation and information flow transactions. In such simulations, the model behavior will change each simulation according to the set of initial parameters assumed for the environment.

As an example of works related to the information analysis using simulation we can mention the "Reconciling belief and vulnerability in information flow" [9].

### Model Checking

Probabilistic model checking resolves the problem of: given a model of a system, test automatically whether this model meets a given specification. Typically, the systems one has in mind are hardware or software systems, and the specification contains safety requirements such as the absence of deadlocks and similar critical states that can cause the system to crash. Model checking is a technique for automatically verifying correct properties of finite-state systems.

Model checking can be also considered as a technique for verifying finite state concurrent systems such as sequential circuit designs and communication protocols. As far as we know there is no previous work related to the verification of the information flow that uses the model checking as a probabilistic technique.

It has a number of advantages over traditional approaches that are based on simulation, testing, and deductive reasoning. In particular, model checking is automatic and usually quite fast. Also, if the design contains an error, model checking will produce a counterexample that can be used to pinpoint the source of the error. But from another point of view it is limited to finite state spaces. Our problem can not be handled with finite state machine because sometimes the

state space is infinite. The model checking is not very expressive to tackle such problem; we are using many quantifiers over functions and variable, things that that this approach can not do.

### Theorem Proving

Theorem proving [7] is another widely used formal verification technique. We first mathematically model the problem to be analyzed and then verify some properties using computer based formal tools. Formal logics as a modeling medium makes theorem proving a technique that formally verifies any system that can be described mathematically, and this gives theorem proving verification approach more flexibility. Theorem provers are mainly based on some well-known axioms and inference rules. Any new theorems must be created from the basic axioms and inference rules or any other theorems already proved.

Probabilistic analysis based on higher-order logic theorem prover can be conducted by first modeling the behavior of the system that needs to be analyzed in higher-order logic, while expressing its random elements in terms of formalized random variables. The next step is using this model to express the probabilistic and statistical properties of the system. For this effect, we need to have to know definitions of probabilistic and statistical properties of random variables in the environment of higher-order logic, such as, PMF, CDF, expectation and variance, etc. Finally, theorems corresponding to the probabilistic and statistical properties of the system model can be mechanically checked for correctness in a theorem prover.

The above mentioned theorem proving based probabilistic analysis approach tends to overcome the limitations of the two previous approaches. Due to the formal nature of the models and properties, probabilistic analysis carried out in a theorem proving environment will be free from any approximation and precision issues. Similarly, the high expressibility of higher-order logic allows us to analyze a wider range of systems without any modeling limitations, so that it can overcome the state-space explosion problem in the case of probabilistic model checking.
In the environment of theorem proving no work has been conducted in the information hiding area.

### 1.3.2 Formalization of Information Theory

After presenting some approaches of the probabilistic analysis that could be useful to conduct the quantification and the verification of the information flow, we present here some of the works related to our project.

The foremost requirement for conducting the formal probabilistic analysis of the quantitative information flow problem in a theorem prover is to have access to a higher-order-logic formalization of probability and information fundamentals. Several formalizations of those notions have been reported in the open literature.

Coble [4] formalized the main concepts of Lebesgue integration and further used these fundamentals to formalize the information theory in HOL. He analyzed the quantitative information flow and in order to define the mutual information he utilized the product space as well as the Radon-Nikodym derivative. He used that formalized theory to analyze the privacy and the anonymity guarantees and proposed the Dining Cryptographers as a case study for that. However, Coble's formalization of Lebesgue integral can only consider finite-valued measures, functions and integrals.

Building on top of Coble's work, Mhamdi [15] generalized the formalizations of the probability and information theory by introducing the notion of extended real numbers, the Borel sigma algebra which covers larger classes of functions in terms of integrability and convergence. He further used these fundamentals to formalize the measure of entropy, relative entropy and mutual information.

In parallel to the previous work, J. Hölzl[12, 11] formalized also, in the environment of Isabelle/HOL, a generic version of the measure, probability and information theory. His definition was done in the same way in Coble's work. He used the measure and the probability theories to define the Kullback-Leibler divergence, entropy, conditional entropy, mutual information and conditional mutual information and verify the properties related to the quantification of the information represented by a random variable.

We are going to utilize in our project the theories, probability and information, formalized in Mhammdi's work due to their completeness and availability in HOL in order to formally verify the information flow using min-entropy and belief-min-entropy. The analysis results can be claimed to be 100% precise, which is an achievement that has not been reported in the open literature so far.

## 1.4 Proposed Method

After revealing the problem we want to deal with, the related works and the different techniques that could be used in order to overcome this problematic, we present in this section our proposed solution that we adopted within this project. Considering the foundations of quantitative information flow, our model will be

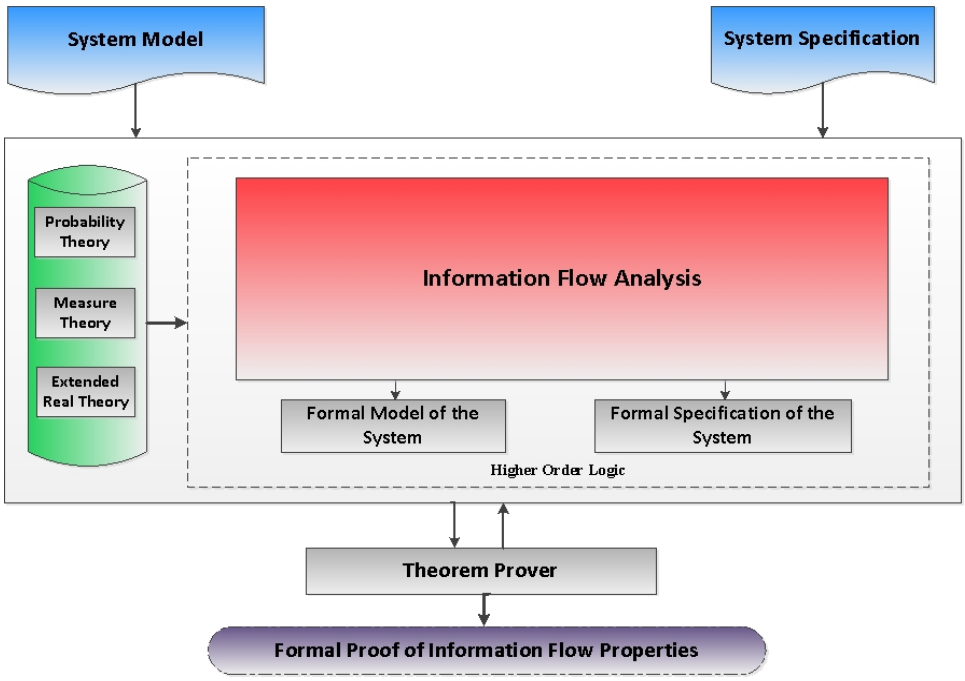


Figure 1.2: Proposed Methodology Overview

described as a program (protocol) having a high input $\mathcal{H}$ and produces a low output $\mathcal{L}$. An attacker $\mathcal{A}$ can observe $\mathcal{L}$ and may be able to get some information about $\mathcal{H}$. We would like to quantify the amount of the initial uncertainty ($\mathcal{H}$), the amount of the remaining uncertainty and the difference between them which is the amount of the information leacked from $\mathcal{L}$.
Our major goal can be described following the schema below:

"*initial uncertainty = information leacked + remaining uncertainty*"

In order to analyze the previous schemas we defined a new framework based on Min-Entropy and Belief Min-Entropy. We first formalize the entropies we need to quantify the previous uncertainties. Then we start proving the different related properties. We consider deterministic and probabilistic program $c$ having as input

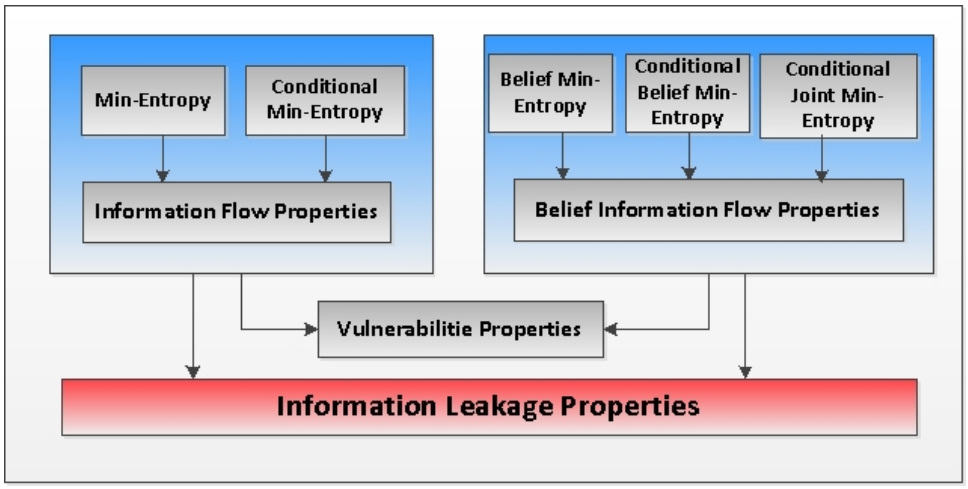


Figure 1.3: Information Flow Analysis

$\mathcal{H}$ and producing $\mathcal{L}$.

During the whole process, we focus our attention on a specific threat model: the probability that the value of $\mathcal{H}$ can be correctly guessed by an adversary $\mathcal{A}$ in one try.

Because of the limitations of the consensus related to Shanon entropy, we propose other alternatives that are based on the concept of vulnerability which is in close relation with the Bayes risk. The vulnerability $V(X)$ is the maximum of the distributions of $X$. This measure is considered to be the worst-case that the secret information is correctly guessed in one try.

We then generalize our model to take into account the attacker's belief. The idea says that the adversary assumes some knowledge about the a priori distributions of the hidden input and its correlation. We use here then the belief-vulnerability which is the expected probability of guessing the hidden input in one try given the adversary's belief. In other words, the adversary chooses the value of the secret input having the maximum a posteriori probability according to his belief. The belief-vulnerability of the secret information is then expressed as a function of the a posteriori distributions and the the adversary's knowledge.

We finally showed the strength of our model and definitions by applying them to various threat scenarios.

## 1.5 Outline of the Report

After presenting the problematic and the way to overcome it, the rest of the thesis will be organized as follows: In Chapter 2, we are giving an overview about the preliminaries and what we will need in order to tackle this work. We first introduce the environment of work, which is HOL theorem prover, the useful theories which are the Probability and the Information theories, and then a brief description of the entropy measures, the min-entropy and the belief-min-entropy.
In the next 2 chapters, Chapter 3 and Chapter 4, we will detail the work by describing the new definitions, proving the related properties, quantifying the amount of uncertainty and analysing the information flow for probabilistic and deterministic programs using Min-Entropy and Belief Min-Entropy.
Chapter 4 will then show the strength of our theoretical work by applying the formalizations on different scenarios of threat programs. We will also show the usefulness on a case study, Min-Entropy leakage of channels in cascade. In this application we prove that the min-entropy leakage of a cascade of two channels can not exceed the leakage of the first channel.
In the final chapter we will give an overview about what it has been done and present some future works that will be handled later on.

# Chapter 2

# Preliminaries

This section will cover the different utile notions related to our work that we are going to use in order to resolve our problem as well as the environment in which our project will be conducted.

## 2.1 HOL Theorem Proving

Higher-Order Logic (HOL) [7, 1] is an interactive theorem prover developed by Mike Gordon at the University of Cambridge for conducting proofs in higher-order logic. It utilizes the simple type theory of Church [3] along with Hindley-Milner polymorphism [16] to implement higher-order logic. HOL has been successfully used as a verification framework for both software and hardware as well as a platform for the formalization of pure mathematics. In mathematics and logic, a higher-order logic is a form of predicate logic that is distinguished from first-order logic by additional quantifiers and a stronger semantics. Higher-order logics with their standard semantics are more expressive, but their model-theoretic properties are less well-behaved than those of first-order logic.

The term "higher-order logic" is commonly used to mean higher order simple predicate logic. Here "simple" indicates that the underlying type theory is simple, not polymorphic or dependent.
The HOL System is an environment for interactive theorem proving in higher order logic. Its most outstanding feature is its high degree of programmability through

the meta-language ML. The system has a wide variety of uses from formalizing pure mathematics to verification of industrial hardware. Academic and industrial sites world-wide are using HOL. HOL denotes a family of interactive theorem proving systems sharing similar (Higher order) logics and implementation strategies. Systems in this family follow the LCF approach as they are implemented as a library in some programming language. This library implements an abstract data type of proven theorems so that new objects of this type can only be created using the functions in the library which correspond to inference rules in higher-order-logic. As long as these functions are correctly implemented, all theorems proven in the system must be valid. In this way, a large system can be built on top of a small trusted kernel.

One of the advantages of HOL is that it is not limited by the size of the state space. Large systems that cannot be verified using the model checker can still be verified by the theorem prover. The use of formal logics as a modeling medium makes theorem proving a very flexible verification technique as it is possible to formally verify any systems that can be described mathematically.

The soundness of HOL theorem proving guarantees that valid results are provable; hence, overcoming the inaccuracies of simulation and paper-and-pencil based techniques. Higher-order logic is a system of deduction with a precise semantics and is expressive enough to be used for the specification of almost all classical mathematics theories. Due to its high expressiveness, higher-order logic can be utilized to precisely model the behavior of any system, while expressing its random or unpredictable elements in terms of formalized random variables, and any kind of system property, including the probabilistic and statistical ones, as long as they can be expressed in a closed mathematical form.

## 2.2 Probability Theory

Probability provides mathematical models for random phenomena and experiments. The purpose is to describe and predict relative frequencies (averages) of these experiments in terms of probabilities of events. The classical approach to formalize probabilities, which was the prevailing definition for many centuries, defines the probability of an event A as $p(A) = \frac{N_A}{N}$ , where $N_A$ is the number of

outcomes favorable to the event A and N is the number of all possible outcomes of the experiment. Problems with this approach include the assumptions that all outcomes are equally likely (equiprobable) and that the number of possible outcomes is finite. KOLMOGOROV later introduced the axiomatic definition of probability, which provides a mathematically consistent way for assigning and deducing probabilities of events.

This approach consists in defining a set of all possible outcomes, $\Omega$, called the sample space, a set F of events which are subsets of and a probability measure $p$ such that $(\Omega, F, p)$ is a measure space with $p(\Omega) = 1$. Using measure theory to formalize probability has the advantage of providing a mathematically rigorous treatment of probabilities and a unified framework for discrete and continuous probability measures. In this context, a probability measure is a measure function, an event is a measurable set and a random variable is a measurable function. The expectation of a random variable is its integral with respect to the probability measure.

**Definition 2.2.1.** *(Probability Space)*

*$(\Omega, F, p)$ is a "probability space" iff it is a measure space and $p(\Omega) = 1$. A probability measure is a measure function and an event is a measurable set.*

**Definition 2.2.2.** *(Independent Events)*

*Two events $A$ and $B$ are independent iff $p(A \cap B) = p(A)p(B)$. Here $A \cap B$ is the intersection of $A$ and $B$, that is, it is the event that both events $A$ and $B$ occur.*

**Definition 2.2.3.** *(Random variable)*

*$X : \Omega \to \mathbb{R}$ is a "random variable" iff $X$ is $(F, \mathcal{B}(\mathbb{R}))$ measurable where $F$ denotes the set of events. Here we focus on real-valued random variables but the definition can be adapted for random variables having values on any topological space thanks to the general definition of the* BOREL *sigma algebra.*

**Definition 2.2.4.** *(Independent Random variable)*

Two random variables X and Y are independent iff $\forall A, B \in \mathcal{B}(\mathbb{R})$, the events $X \in A$ and $Y \in B$ are independent.
The set $X \in A$ denotes the set of outcomes $\omega$ for which $X(\omega) \in A$. In other words $X \in A = X^{-1}(A)$. Equivalently, $X$ and $Y$ are independent iff $\forall A, B \in B(R), p(X \in A \cap Y \in B) = p(X \in A).p(Y \in B)$. The event $X \in A$ is used to

define the probability mass function (PMF) of a random variable.

**Definition 2.2.5.** *(Probability Mass Function : PMF)*

*The "probability mass function" $p_X$ of a random variable $X$ is defined as the function assigning to $A$ the probability of the event $X \in A$.*

$$\forall A \in \mathcal{B}(\mathbb{R}), p_X(A) = p(X \in A) = p(X^{-1}(A)) \tag{2.1}$$

The joint PMF of two random variables and of a sequence of random variables are defined as

$$\forall A, B \in \mathcal{B}(\mathbb{R}), p_{XY}(A, B) = p(X \in A \cap Y \in B) = p(X^{-1}(A) \cap Y^{-1}(B)) \tag{2.2}$$

$$\forall A_1, ..A_n \in \mathcal{B}(\mathbb{R}), p_{X_1 X_2 .. X_n}(A_1, ..A_n) = p(\bigcap_{i=1}^{n} X_i \in A_i) = p(\bigcap_{i=1}^{n} X_i^{-1}(A_i)) \tag{2.3}$$

**Definition 2.2.6.** *(Expected value)*

*The "expected value" of a random value $X$ is defined as the integral of $X$ with respect to the probability measure. $E[X] = \int_\Omega X dp$ The properties of the expectation are the following*

- $E[X + Y] = E[X] + E[Y]$
- $E[aX] = aE[X]$
- $E[a] = a$
- $X \leq Y$ then $E[X] \leq E[Y]$
- X and Y are independent then $E[XY] = E[X]E[Y]$

**Definition 2.2.7.** *(Variance and Covariance)*

*The "variance" of a random variable $X$ is defined as $Var(X) = E[\mid X - E[X] \mid^2]$. The "covariance" of two random variables $X$ and $Y$ is defined as $Cov(X, Y) = E[(X - E[X])(Y - E[Y])]$. Two random variables $X$ and $Y$ are uncorrelated iff $Cov(X, Y) = 0$. The variance and covariance have the following properties*

- $Var(X) = E[X^2] - E[X]^2$
- $Cov(X, Y) = E[XY] - E[X]E[Y]$

- $Var(X) \geq 0$
- $\forall a \in R, Var(aX) = a^2 Var(X)$
- $Var(X+Y) = Var(X) + Var(Y) + 2Cov(X,Y)$
- If X,Y uncorrelated then $Var(X+Y) = Var(X) + Var(Y)$
- If $\forall i \neq j, X_i, X_j$ are uncorrelated,then $Var(\sum_{i=1}^{N} X_i) = \sum_{i=1}^{N} Var(X_i)$

## 2.3 Information Theory

In this section we briefly describe the different notions of the Information Theory that shore our work and illustrate our model.
Information Theory [14] is a branch of quantifying information. It is used in different area such as signal processing, data compression, storing and communicating data. Recently, it was commonly used in cryptography and information flow analysis [19]. We use different tools of information theory to reason about the uncertainty of a random variable.

The most important elements of the information theory are the *entropy*, *the mutual information*, *the relative entropy*, *the conditional entropy* and *the Rényi's entropy.*
Let $X, Y$ to denote discrete random variables and the corresponding $x, y$ and $\mathcal{X}, \mathcal{Y}$ for their values and set of values respectively. We denote by $p(x), p(y)$ the probability of $x$ and $y$ respectively and by $p(x,y)$ their joint probability. *The Shannon entropy* $(X)$ of $X$ is defined as:

$$H(X) = -\sum_{x \in \mathcal{X}} p(x) log\, p(x) \tag{2.4}$$

The entropy measures the uncertainty of a random variable. It takes its maximum value $log|\mathcal{X}|$ when $X$ is uniformly distributed and its minimum value 0 when $X$ is a constant. We take the logarithm with a base 2 and thus measure entropy in bits.
*The conditional entropy*:

$$H(X|Y) = -\sum_{y \in \mathcal{Y}} p(y) \sum_{x \in \mathcal{X}} p(x|y) log\, p(x|y) \tag{2.5}$$

measures the amount of uncertainty of $X$ when $Y$ is known. It can be shown that $0 \leq H(X|Y) \leq H(X)$ with the leftmost equality holding when Y completely determines the value of $X$ and the rightmost one when $Y$ reveals nothing about $X$, i.e., $X$ and $Y$ are independent random variables. Comparing $H(X)$ and $H(X|Y)$ give us the notion of *mutual information*, denoted $I(X;Y)$ and defined by:

$$I(X;Y) = H(X) - H(X|Y) \tag{2.6}$$

It is non-negative, symmetric and bounded by $H(X)$. In other words $0 \leq I(X;Y) = I(Y;X) \leq H(X)$. *The relative entropy* or Kullback-Leibler distance between 2 probability distributions $p$ and $q$ on the same set $\mathcal{X}$, denoted $D(p\|q)$, is defined as:

$$D(p\|q) = \sum_{x \in \mathcal{X}} p(x) log \frac{p(x)}{q(x)} \tag{2.7}$$

It is non-negative (but not symmetric) and it is 0 if and only if $p = q$. The relative entropy measures the inaccuracy or information divergence of assuming that the distribution is $q$ when the true distribution is $p$. *The guessing entropy* $G(X)$ is the expected number of tries required to guess the value of X optimally. The optimal strategy is to guess the values of $X$ in decreasing order of probability. Thus if we assume that $X = \{x_1, x_2, \ldots, x_n\}$ and $x_i$'s are arranged in decreasing order of probabilities, i.e., $p(x_1) \geq p(x_2) \geq \ldots \geq q(x_n)$, then

$$G(X) = \sum_{1 \leq i \leq n} ip(x_i) \tag{2.8}$$

*The min-entropy* $H_\infty(X)$ of a random variable is given by:

$$H_\infty(X) = -log\, max_{x \in \mathcal{X}}\, p(x) \tag{2.9}$$

It is an instance of the *Rényi-entropy*

$$H_\alpha(X) = \frac{1}{1-\alpha}\, log\, (\sum_{x \in \mathcal{X}} P[X = x]^\alpha) \tag{2.10}$$

with $\alpha = \infty$. The *min-entropy* measures the difficulty for an attacker to correctly guess the value of X in one try (obviously using the optimal strategy above). It can be shown that $H_\infty(X) \leq H(X)$ with equality when $X$ is uniformly distributed.

In general, $H(X)$ can be arbitrary higher than $H_\infty(X)$, since it can be arbitrary high even if X assumes a given value with probability close to 1.
*The belief-min-entropy* of two random variables $X$ and $Y$, such that $X$ describes the input distribution and $Y$ describes the adversary's additional information about $X$, is given by:

$$H_\infty(X:Y) = -log\,(\sum_{y\in\mathcal{Y}} p_\rho(y)\,\frac{1}{|\Gamma_y|}\sum_{x\in\Gamma_y} p_\rho(x|y)) \tag{2.11}$$

where $\Gamma_y = argmax_{x\in\mathcal{X}}\, p_\beta(x|y)$ such that $p_\rho$ denotes the *a priori input distribution* and $p_\beta$ denote the *a priori adversary's assumed distribution.*
The following table provides an overview about the HOL4 functions and symbols we are going to use
As we mentioned above, our main concern in this project is to evaluate the security

Table 2.1: HOL Symbols and Functions

| **HOL Symbol** | **Meaning** |
|---|---|
| $\forall$ | Logical *for all* |
| $\exists$ | Logical *exists* |
| $\wedge$ | Logical *and* |
| $\vee$ | Logical *or* |
| (a, b) | A pair of two elements |
| $\lambda$x.fx | Function that maps $x$ to $f(x)$ |
| $\emptyset$ | Empty Set |
| a $\in$ S | $a$ in $S$ |
| FINITE S | $S$ is a finite set |
| A $\subseteq$ B | A is a subset of B |
| A $\cap$ B | A intersection B |
| A $\cup$ B | A union B |
| disjoint A B | Sets A and B are disjoint |
| extreal_max_set A | The maximum element in a set A |
| IMAGE f A | Set with elements $f(x)$ for all $x \in A$ |
| SIGMA ($\lambda$n. f n) s | $\sum_{n\in s} f(n)$ |
| distribution p X | Probability function ($\lambda\ x.p(X = x)$) |
| random_variable X p Borel | Random variable function |

properties of the confidential information. So in the next two chapters we are going

to describe the work that we did. They will cover the different approaches and formalizations that are needed in order to conduct the analysis of the information flow. The first one will handle the different notions related to the analysis of the information leakage using the *min-entropy* and the second part will focus on the analysis using the *belief-min-entropy*.

# Chapter 3

# Formal Analysis of Information Flow Using Min-Entropy

Due to the lack of the consensus on the Shanon entropy we are going to explore other alternatives offering measures that better evaluate the security regarding the probability of guessing the secret input $H$ in one try.

## 3.1 The A Priori Behavior

In order to model that solution, we propose to use the notion of the *vulnerability* that will be used to define information leakage based on *min-entropy*.

**Definition 3.1.1.** *(The Vulnerability of a Random Variable)*

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

$$V(X) = \max_{x \in \mathcal{X}} P[X = x]. \tag{3.1}$$

The vulnerability $V(X)$ is then considered as the worst-case probability that the adversary $\mathcal{A}$ can guess the value of $X$ correctly in one try.

It is clear from this definition that it only depends on the maximum of the

distribution of $X$, which means that it is only focusing on the greatest risk of guessing $X$. Here, this measure is a probability so it is always between 0 and 1. But since we would like to quantify the information flow, which is in *bits*, we extend the definition of the vulnerability into an entropy that maps $V(X)$. This measure is known as the min-entropy.

**Definition 3.1.2.** *(The Min-Entropy)*

*The min-entropy of a random variable X, denoted $H_\infty(x)$, is given by*

$$H_\infty(X) = -log(V(X)) = -log(\max_{x \in \mathcal{X}} P[X = x]) \tag{3.2}$$

In Higher-Order-Logic (HOL), in order to formalize that notion, we need first to define a function that returns the maximum of a set and the base-2 logarithm. Once done the min-entropy is expressed in HOL by

```
⊢ min_entropy H p = -log(extreal_max_set(IMAGE(λh.
distribution p H {h}) (H(Ω)))
```

where the `extreal_max_set` refers to the function returning the maximum of a set, and `IMAGE f s` return the image of a set $s$ by $f$.

After defining the notion of *min-entropy*, we prove its related properties. We first prove the following property

**Theorem 3.1.1.** *(Upper bound of the Min-Entropy)*

$$\forall\, x\, b.\, x \in \mathcal{X}\, and\, (P[X = x] \leq 2^{-b}) \Rightarrow b \leq H_\infty(x)$$

This property determines an upper bound of the min-entropy.By just reasoning about the value of the distribution of the random variable, we can get the maximum value that $H_\infty$ can reach.

In case the random variable is uniformly distributed among $n$ values, where $n$ is the cardinal of the set $\mathcal{H}$ ($|\mathcal{H}|$), the min-entropy $H_\infty$ is equal to *log* n. In Higher-Order-Logic we formalize this theorem as following

```
⊢ ∀p H. FINITE (Ω) ∧ (random_variable H p Borel) ∧
    ∀h.  h ∈ H(Ω) ⇒ distribution p H {h}= 1/|H(Ω)| ⇒
    min_entropy H p = log |H(Ω)|
```

where $\Omega = p\_space p$. In this theorem, the first assumption is needed for the

computation of the maximum of a set, because this function is only defined for finite sets. The proof of that property is simple, we first rewrite the distribution by its value expressed in the 4th assumption and then with the logarithm property we get the expected result.

## 3.2 The A Posteriori Behavior

In the previous section, we only considered the high input which is the initial uncertainty, that was modeled by the Min-Entropy ($H_\infty(H)$). In this section, in order to quantify the information leakage, we need another quantity that models the remaining uncertainty. Corresponding to the schema we considered before

"*information leakage = initial uncertainty - remaining uncertainty*"

the remaining uncertainty will be modeled by the *conditional min-entropy*.
As a first step, we consider the *conditional vulnerability*, which gives the expected probability of guessing $X$ in one try, given $Y$:
**Definition 3.2.1.** *(The Conditional Vulnerability)*

*Given (jointly distributed) random variables $X$ and $Y$, the conditional vulnerability $V(X|Y)$ is defined as*

$$V(X|Y) = \sum_{y \in \mathcal{Y}} P[Y = y] V(X|Y = y) = \sum_{y \in \mathcal{Y}} P[Y = y] \max_{x \in \mathcal{X}} P[X = x|Y = y] \quad (3.3)$$

For the probabilistic programs we notice that it is easier to calculate $V(H|L)$ from the a priori distribution on $H$ and the matrix of conditional distribution $P[L = l|H = h]$. This relation is expressed by the Bayes' rule
**Theorem 3.2.1.** *(The Bayes' Rule)*

$$P[H = h|L = l]P[L = l] = P[L = l|H = h]P[H = h] \quad (3.4)$$

The formal version of this theorem in HOL is

```
⊢ ∀p H L h l.  FINITE (Ω) ∧ random_variable H p Borel ∧
```

```
   random_variable L p Borel ∧
   (∀h.  h ∈ Ω ⇒ {h} ∈ events p) ⇒
   conditional_distribution p H L (h,l) * distribution p
L l =
   conditional_distribution p L H (l,h) * distribution p
H h
```

Using the Bayes' rule, we can conduct the following analysis that gives us another form of the conditional vulnerability (later the conditional min-entropy)

$$\begin{aligned} V(H|L) =& \sum_{l\in\mathcal{L}} P[L=l]V(H|L=l) \\ =& \sum_{l\in\mathcal{L}} P[L=l] \max_{h\in\mathcal{H}} P[H=h|L=l] \\ =& \sum_{l\in\mathcal{L}} \max_{h\in\mathcal{H}} P[H=h|L=l]P[L=l] \\ =& \sum_{l\in\mathcal{L}} \max_{h\in\mathcal{H}} P[L=l|H=h]P[H=h] \end{aligned} \tag{3.5}$$

We next define the conditional min-entropy that will model the remaining uncertainty, which is the probability of guessing the value of $X$ after observing the output.

**Definition 3.2.2.** *(The Conditional-Min-entropy)*

*The conditional min-entropy of two random variables $X$ and $Y$, denoted $H_\infty(X|Y)$ is defined as*

$$\begin{aligned} H_\infty(X|Y) =& log\, \frac{1}{V(X|Y)} \\ =& -log(V(X|Y))0 \\ =& -log(\sum_{y\in\mathcal{Y}} \max_{x\in\mathcal{X}} P[Y=y]P[X=x|Y=y]) \end{aligned} \tag{3.6}$$

Formally, this definition is expressed in HOL as follows

```
⊢ ∀p H L.
```

```
   conditional_min_entropy p H L = - log( ∑
                                          l∈L(Ω)
extreal_max_set(
      IMAGE (λh.  distribution p L{l} *
      conditional_distribution p H L ({h},{l})) (H(Ω))))
```

where `SIGMA f s` means $\sum_{x \in s} f(x)$
We will then use our new definitions to formalize the information leakage. This quatity is expressed as the difference between the quantity of information in the input (initial uncertainty) and the one in the output (remaining uncertainty), previously modeled respectively by the Min-Entropy ($H_\infty(H)$) and the conditional min-entropy ($H_\infty(H|L)$).

**Definition 3.2.3.** *(The Information Leakage)*

*The information leaked between two distributions of random variables $X$ and $Y$, denoted $IL(X;Y)$, is*

$$IL(X;Y) = H_\infty(X) - H_\infty(X|Y) \tag{3.7}$$

We focus next on some specific programs, especially deterministic programs with a uniformly distributed input. In order to do that, we have to express the condition that we defined as follows

**Definition 3.2.4.** *(The Determinism Condition)*

```
⊢ ∀L c.  deterministic_cond L c = (L = (λx.  c))
```

We established this condition over the output random variable which is the best way to express the determinism which means that the output value is always known. This definition operates over a random variable and assign to it a constant. It leads us to extract a number of consequences that will guide us through our analysis. The following results will be based on the determinism condition. As a first result, the conditional-distribution ($P(L|H)$) will only take the values 0 and 1.

**Theorem 3.2.2.** *(Deterministic Conditional Distribution Cases)*

```
⊢ ∀p H L h l.  (prob_space p) ∧
   ∀h.  h ∈ (Ω) ⇒ {h} ∈ events p ∧
   FINITE (Ω) ∧
```

```
deterministic_cond L c ⇒
conditional_distribution p L H ({l},{h}) = 1 ∨
conditional_distribution p L H ({l},{h}) = 0
```

As a direct consequence of the previous result, the input state space can be described as the union of two sets depending on the value of P(L|H)
**Theorem 3.2.3.** *(Deterministic Events' Space)*

$$\forall L\ l.\ \mathcal{H} = \{h | h \in \mathcal{H} \wedge P(L = l | H = h) = 0\} \cup \{h | h \in \mathcal{H} \wedge P(L = l | H = h) = 1\} \tag{3.8}$$

The proof of this theorem is based on *Theorem 3.2.2* and some notions from the set theory.

After proving all the properties we need to compute the information leakage for deterministic programs. The mathematical description of that theorem is
**Theorem 3.2.4.** *(Information Leakage for Deterministic Program)*

*If a program is deterministic modeled by an initial uncertainty H and a remaining uncertainty L whenre H is uniformly distributed, then the information leaked is log|$\mathcal{L}$|*

$$IL_\infty(H; L) = log|\mathcal{L}| \tag{3.9}$$

This result is expressed in HOL as

```
⊢ ∀H L p.  FINITE (Ω) ∧ random_variable H p Borel ∧
   random_variable L p Borel ∧ ∀h.h∈(Ω)⇒{h}∈ events p ∧
   ∀h.h∈ H(Ω)⇒(distribution p H {h} = 1/|ℋ|) ∧
   deterministic_cond L c ⇒
   information_leakage p H L = log|ℒ|
```

The proof of this result was conducted using *Theorem 3.2.2* and *Theorem 3.2.3* that have been extracted from the determinism condition, rewriting techniques and notions from the set theory as well as the following property related to the maximum of two sets that was proved in the meanwhile

$$\max(s \cup t) = \mathbf{max}(\max s, \max t) \tag{3.10}$$

where the $max$ in bold refers to the maximum of two extended reals and the other max refers to the maximum of a set. The interpretation of the result proved in $Theorem4$ is that for a deterministic program with a uniformly distributed input, the information leaked depends only on the output of the program. That says the larger the output space is, the greater the quantity of information leaked will be.

We mentioned in this chapter a model for quantifying the information flow using the consensus of the min-entropy. The high secret input or the initial uncertainty was modeled by $H_\infty(H)$, the remaining uncertainty by $H_\infty(H|L)$ and the difference between these two measures gives us how mach information has been leaked. We also tackled a special case of the deterministic programs and proved that for this kind of programs the quantity of information leaked depends only on the observable output, the larger the output state space is, the more vulnerable the system is.

In the next chapter we will present another model to analyze the information flow. This approach will take into consideration the attacker's belief defined as an extra knowledge about the system behavior.

# Chapter 4

# Formal Analysis of Information Flow Using Belief Min-Entropy

In this chapter we are going to tackle the information flow analysis using a different approach taking into consideration the adversary's extra knowledge about and the behavior of the program.
As mentioned in the previous section , we are going also to use the notion of vulnerability as a starting point and we are going to define the *belief-vulnerability*. For that quantity we are going to distinguish between the *a priori belief-vulnerability* and the *a posteriori belief-vulnerability*. The first one take into account only the input and the adversary's belief before observing the output and the second one considers in addition the observable output.

## 4.1 The A Priori Behavior

Let $B$ be teh random variable modeling the adversary's additional information about a a high level random variable $X$. Then the belief-vulnerability of $X$ is the expected probability of guessing $X$ in one try given the adversary's belief $B$. Since we have two behavior in this model, one related to the system and another one related to the belief behavior. We then represent these two aspects by two differen distributions $p_\rho$ and $p_\beta$.
Given an additional information $B = b$, the adversary will choose a value having

the maximal conditional probability according to her belief, that is a value $x' \in \Gamma_b$, where $\Gamma_b = arg \max_{x \in \mathcal{X}} p_\beta(x|b)$, where the $arg \max_{x \in \mathcal{X}} p_\beta(x|b)$ returns the elements from $\mathcal{X}$ having the maximal conditional-distribution. The vulnerability of $X$ given $b$ is then the real probability that the adversary's choice is correct, which is the conditional probability $p_\rho(x'|b)$. As there might be many values of $X$ with the maximal conditional probability, the attacker will pick uniformly at random one element in $\Gamma_b$. Hence we have the following definition.

**Definition 4.1.1.** *(The A Priori Belief Vulnerability)*

*Let $X$ be a random variable and $B$ the adversary's extra knowledge about $X$. Then the belief-vulnerability of $X$, denoted $V(X : B)$, is defined as*

$$V(X : B) = \sum_{b \in \mathcal{B}} \frac{1}{|\Gamma_b|} p(b) \sum_{x \in \mathcal{X}} p(x|b) \tag{4.1}$$

In order to define this quantity in HOL we need first to formalize the set $\Gamma_b$, denoted in our case `belief_set p q X B b`, where $p$ et $q$ are two probability spaces since we have two different distributions $p_\rho$ and $p_\beta$. Hence the belief vulnerability is defined as

$$\vdash \forall \texttt{p q X B.}\ \texttt{belief\_vulnerability p q X B} = \sum_{b \in \mathcal{B}} \frac{1}{|\Gamma_b|} \star \texttt{distribution p B \{b\}} \star \sum_{x \in \Gamma_b} \texttt{conditional\_distribution p X B (\{x\},\{b\})}$$

From the belief vulnerability we can easily define the *Belief Min-Entropy*

$$\begin{aligned} H_\infty(X : B) &= - log(V(X : B)) \\ &= - log(\sum_{b \in \mathcal{B}} \frac{1}{|\Gamma_b|} p(b) \sum_{x \in \mathcal{X}} p(x|b)) \end{aligned} \tag{4.2}$$

Let $Belief_\perp$(A,B) be the set of totally inaccurate beliefs,

$$Belief_\perp(X, B) = \{(x, b) | b \in \mathcal{B}, x \in \Gamma_b \, and \, p_\rho(x|b) = 0\} \tag{4.3}$$

Then the following result holds

**Theorem 4.1.1.** *(Infinite Belief Min-Entropy)*

*Let $X$ be a random variable and $B$ the adversary's extra knowledge about $X$*

*then*

$$(x, b) \in Belief_{\perp}(X, B) \Rightarrow H_{\infty}(X : B) = +\infty$$

The proof of this result was a direct consequence of the rewriting of the belief min-entropy using the Bayes' rule.
Next, we show that under some constraints the Min-Entropy of a random variable is less or equal then belief min-entropy and under some others the belief min-entropy is equal to the conditional min-entropy. These two results were formalized and proved in HOL by computing different notions from the set theory and using some properties related to the maximum of a set and also the Bayes' rule

**Theorem 4.1.2.** *(Relation between Min-Entropy and Belief Min-Entropy)*

$$If\, \forall b \in \mathcal{B}, x \in \Gamma_b \Rightarrow p_{\rho}(b|x) \leq \tfrac{1}{|\mathcal{B}|}\, then\, H_{\infty}(X) \leq H_{\infty}(X : B)$$

**Theorem 4.1.3.** *(Belief Min-Entropy and Conditional Min-Entropy)*

$$If\, \forall b \in \mathcal{B}, a \in \Gamma_b \Rightarrow p_{\rho}(b|a) = \max_{a' \in \mathcal{A}} p_{\rho}(a'|b)\, then\, H_{\infty}(A : B) = H_{\infty}(A|B)$$

After verifying these few results, we are going to reason about the effect of the adversary's initial belief accuracy. We first give a definition of the accurate initial belief.

**Definition 4.1.2.** *(Accurate Initial Belief)*

*An adversary's initial belief is c-accurate (*$0 < c \leq 1$*) if that holds*

$$\forall\, b \in \mathcal{B}, \frac{1}{\Gamma_b} \sum_{a \in \Gamma_b} p_{\rho}(a|b) \geq c * \max_{a \in \mathcal{A}} p_{\rho}(a|b)$$

The formal version of the previous definition in HOL is

```
⊢ ∀X B p q b c.  c_accurate_initial_belief X B p q b c =
   0<c ∧ c≤1 ∧ c *
   extreal_max_set
      (IMAGE (λx.  conditional_distribution p X B
({x},{b})) X(Ω1))≤
      1/|Γ_b| * Σ_{x∈Γ_b} conditional_distribution p X B ({x},{b})
```

The c-accurate belief has an impact on the vulnerability of $X$ in presence of extra information by a factor at least $c$, this result is proved in the following theorem

**Theorem 4.1.4.** *If the adversary's belief is c-accurate then*

$$H_\infty(X:B) \leq H_\infty(X|B) + log(\frac{1}{c}) \tag{4.4}$$

In HOL we formalize this property as follows

```
⊢ ∀p q X B c.  (FINITE(Ω1)) ∧
  (∀x.  x ∈ Ω1 ⇒ {x} ∈ events p) ∧
  (random_variable X p Borel) ∧
  (random_variable B p Borel) ∧
  (∀b.  c_accurate_initial_belief X B p q b c) ⇒
  (belief_min_entropy p q X B ≤
  (conditional_min_entropy p X B + log(1/c)))
```

Next we proved the relation between the conditional min-entropy and the belief min-entropy in the general case saying that the conditional min-entropy is always less than or equal to the belief min-entropy thus we have

$$H_\infty(X|B) \leq H_\infty(X:B) \tag{4.5}$$

And finally, we show that when $X$ is uniformly distributed, we can obtain a better upper bound. We begin by recalling the result saying, *if $X$ is uniformly distributed and the program is deterministic then*

$$H_\infty(X|B) = log(\frac{|\mathcal{X}|}{|\mathcal{B}|}) \tag{4.6}$$

Thus we have the following theorems

**Theorem 4.1.5.** *(Upper Bound of the Belief Min-Entropy in a deterministic program)*

*If $X$ is uniformly distributed and the actual correlation $p_\rho(b|x)$ is deterministic then*

$$log(\frac{|\mathcal{X}|}{|\mathcal{B}|}) \leq H_\infty(X:B) \tag{4.7}$$

In the environment of Higher Order Logic this result is

```
⊢ ∀X B sp ev p1 p2 c.  FINITE (Ω1) ∧
```

$\Omega 1 \neq \emptyset \wedge$ random_variable X $\Omega 1$ Borel $\wedge$
random_variable B $\Omega 1$ Borel $\wedge$
$\forall$x b. x $\in$ belief_set (sp,ev,p1) (sp,ev,p2) X B b $\wedge$
$\forall$b. b $\in$ B($\Omega 1$) $\wedge$
$\forall$x. x $\in \Omega 1 \Rightarrow$ {x} $\in$ events (sp,ev,p1) $\wedge$
$\forall$x. x $\in$ belief_set (sp,ev,p1) (sp,ev,p2) X B b $\Rightarrow$
distribution (sp,ev,p1) X {x} = $\frac{1}{A(\Omega 1)} \wedge$
deterministic_cond B c $\Rightarrow$ log $\frac{|A(\Omega 1)|}{|B(\Omega 1)|} \leq$
belief_min_entropy sp ev p1 p2 X B

The proof of this theorem of this theorem is a direct consequence from (4.6) and (4.5).

## 4.2 The A Posteriori Behavior

Let $B$ denote the adversary's observation. We define the belief-vulnerability conditioned to the low observations of the adversary and we notice that in this case, the low observed output could be helpful for the adversary in order to select the inaccurate beliefs. In addition, if an observation is contradicting with his initial belief about the extra information $b$, that means that there is no high input $x \in \Gamma_b$ such that $p_\rho(y|x) > 0$. In the other hand, a belief $b$ is compatible to an observation $y$, if there exists a high input $x \in \Gamma_b$ verifying $p_\rho(y|x) > 0$. Let $y$ and $b$ be the adversary's observation and initial belief respectively. He will then only try values $x \in \Gamma_b$ for which $p_\rho(y|x) > 0$ if his belief and observation are compatible. Otherwise, as the evidence contradicts his belief, he will throw it away and only use the observation.
Let $\Gamma_{b,y}$ denote the set of possible adversary's choices according to both his belief and his low observation. Then

$$\Gamma_{b,y} = \begin{cases} arg\max\limits_{x \in \mathcal{X}} \ p_\beta(x|b,y) & \text{if } b \text{ and } y \text{ are compatible} \\ arg\max\limits_{x \in \mathcal{X}} \ p_\beta(x|y) & \text{otherwise} \end{cases} \tag{4.8}$$

In HOL we defined this set as belief_conditionned_set p q X B Y b y. Then we define the a posteriori belief-vulnerability as follows

**Definition 4.2.1.** *(A Posteriori Belief-Vulnerability)*

*Let $X$ be the high input of a program, $Y$ its low output and $B$ the adversary's initial belief about $A$. Then the belief-vulnerability of $X$ given $Y$, denoted $V(X|Y : B)$, is defined as*

$$\sum_{y \in \mathcal{Y}} \sum_{b \in \mathcal{B}} p_\rho(y, b) \, \frac{1}{|\Gamma_{b,y}|} \sum x \in \Gamma_b \, p(x|y, b) \tag{4.9}$$

```
⊢ ∀X B p q.  conditional_belief_vulnerability p q X B Y =
```
$\sum_{y \in Y(\Omega)} \sum_{b \in B(\Omega)}$ `joint_distribution p B Y ({b},{y}) *`
$\frac{1}{|\Gamma_b|}$ `*`
$\sum_{x \in \Gamma_b}$ `belief_conditional_distribution p X Y B`
`({x},{y},{b})`

Next we define the remaining uncertainty ($H_\infty(X|Y : B)$) from the a posteriori belief-vulnerability the same way as previously. As in the previous section, we then define the notion of adversary's post belief's accuracy.

**Definition 4.2.2.** *(Post-Belief's Accuracy)*

*An adversary's post-belief is c-accurate if* $c * \max_{x \in \mathcal{X}} p_\rho(x|b, y) \leq \frac{1}{|\Gamma_{b,y}|} \sum_{x \in \Gamma_{b,y}} p_\rho(x|b, y)$ for all $y \in \mathcal{Y}$ and $b \in \mathcal{B}$. We can show then that a 1-accurate post-belief is an information (100% accurate)

**Theorem 4.2.1.** *(100% Accurate Post-Belief)*

*Let $A$ be the high input of a program, $O$ its low output and $B$ be an additional information about $A$. If the adversary's post-belief is 1-accurate then*

$$H_\infty(A|O : B) = H_\infty(A|B, O) \tag{4.10}$$

```
⊢ ∀X B Y p q.
  FINITE (Ω) ∧ random_variable X p Borel ∧
  random_variable B p Borel ∧ random_variable Y p Borel ∧
  ∀x.  x ∈ (Ω) ⇒ {x} ∈ events p ∧
  ∀b y.  c_accurate_post_belief X B Y p q b y 1 ⇒
  conditional_belief_min_entropy p q X B Y =
conditional_joint_min_entropy p X B Y
```

We then establish the following bound for the remaining uncertainty based on the belief-vulnerability and we get a relation between the remaining uncertainty and another quantity that we defined as conditional joint Min-Entropy, the last mentioned quantity will help us to establish the deterministic uncertainty in the output.

**Theorem 4.2.2.** *(Lower Bound for Remaining Uncertainty)*

*Let $X$ be a random variable, $B$ the additional information about $X$ and $Y$ be the low output of the program. Then*

$$H_\infty(X|Y,B) \leq H_\infty(X|Y:B) \tag{4.11}$$

which is formalized in HOL as follows

```
⊢ ∀X B Y p q.
  FINITE (Ω) ∧ random_variable X p Borel ∧
  random_variable B p Borel ∧ random_variable Y p Borel ∧
  ∀x.  x ∈ (Ω) ⇒ {x} ∈ events p ⇒
  conditional_joint_min_entropy p X B Y ≤
  conditional_belief_min_entropy p q X B Y
```

From the previous result we can verify the same property related to the special case of deterministic program where in addition the input $X$ is uniformly distributed. Hence

**Theorem 4.2.3.** *(Lower Bound for Remaining Uncertainty in Deterministic Program)*

*If $X$ is uniformly distributed and both the protocol and the actual correlation between $X$ and $B$ are deterministic then*

$$log(\frac{|\mathcal{X}|}{|\mathcal{Y}|.|\mathcal{B}|}) \leq H_\infty(X|Y:B) \tag{4.12}$$

From the above result, we conclude that the belief behavior helps the adversary in choosing more reliable initial knowledge based on the observations. The above mentioned properties have been verified before [9] but the main novelty of our work was to re-verify these results using an interactive theorem prover. Based on the soundness of theorem proving, the formally verified theorems are guaranteed

to be accurate and contain all the required assumptions. Moreover, these formally verified results can be built upon to reason about information flow analysis of various applications within the sound core of a theorem prover. For illustration purposes, the information leakage of cascade of channels is formally analyzed in the next section. These added advantages have been attained at the cost of human effort in formalizing and interactively verifying the above mentioned results. The proof script [10] is composed of 3400 lines of code and took about 1000 man-hours of development time [10].

# Chapter 5

# Case Study

The previous section establishes the reasonableness of our definitions in terms of their theoretical properties. Now we show the utility of our approach by applying it to various threat scenarios. And then we will extend the range of these scenarios to tackle a heavier case study: Min-Entropy leakage of Channels in Cascade. In this application we are going to use our theories to evaluate the information leakage of cascade channels.

## 5.1 Small Scenarios

In this section we are targeting a simple scenarios of attacks. We define the input and the output spaces as well as the transition function and we will quantitatively analyze the information flow.

**Example 1:** Let $A$ be a random variable with publicly-known uniform a priori distribution over $\mathcal{A} = \{0,\ 1,\ 2,\ 3\}$. Assume that the adversary's additional observable is the parity of $A$, i.e. $B = \{0,\ 1\}$, with the following deterministic belief's correlation $p_\beta(b_k|a_i) = p(a_i mod 2 = b_k)$. In other words, the adversary believes that her additional information accurately reflects that the value of $A$ is an even number if $B = 0$ and odd otherwise. Now suppose that $A$ is the high

input of the deterministic program $C1$ below, whose low output is

$$O = \begin{cases} 1 & \text{if } a \in \{0,1\} \\ 2 & \text{otherwise.} \end{cases}$$

---

PROG 1

**BEGIN**

$O \leftarrow log(A+2)$

**END**

---

In the case of wrong belief, i.e., when the attacker believes that the value of $A$ is even (resp. odd) when it actually is odd (resp. even), his low observation of PROG C1 does not allow him to correct his belief. Indeed, both observations can be induced by any number under the different conditions.

**Example 2:** Suppose that $A$ is uniformly distributed over {0, 1, 2, 3} and the adversary's extra information is about the parity of $A$. Assume that the a priori distribution of $A$ is publicly-known, i.e., $\forall a \in \mathcal{A}, p_\beta(a) = p_\rho(a)$. Assume also that the adversary believes that his extra information is accurate, that is he assumes the following correlation:

| $p_\beta(b\|a)$ | $b_0$ | $b_1$ |
|---|---|---|
| $a_0$ | 1 | 0 |
| $a_1$ | 0 | 1 |
| $a_2$ | 1 | 0 |
| $a_3$ | 0 | 1 |

Then we have here, $\Gamma_0 = \{0, 2\}$ and $\Gamma_1 = \{1, 3\}$. Considering the following program $a$ and $B$ previously defined and the output is $\mathcal{O} = \{0,\ 1,\ 2\}$.

---

PROG 2

**BEGIN**

$O \leftarrow log(A+1)$

**END**

---

Both the program and the adversary's assumed correlation are deterministic, it is therefore easy to compute the adversary's belief conditional distribution $p_\beta(a|o,b)$ and the associated possible choices $\Gamma_{o,b}$.

| $p_\beta(a\|o,b)$ | $a_0$ | $a_1$ | $a_2$ | $a_3$ | $\Gamma_{b_k,o_j}$ |
|---|---|---|---|---|---|
| $b_0, o_0$ | 1 | 0 | 0 | 0 | $\{a_0\}$ |
| $b_0, o_1$ | 0 | 0 | 1 | 0 | $\{a_2\}$ |
| $b_0, o_2$ | 0 | 0 | 0 | 1 | $\{a_3\}$ |
| $b_1, o_0$ | 1 | 0 | 0 | 0 | $\{a_0\}$ |
| $b_1, o_1$ | 0 | 1 | 0 | 0 | $\{a_1\}$ |
| $b_1, o_2$ | 0 | 0 | 0 | 1 | $\{a_3\}$ |

We proceed now to the analysis of two programs described above. Each of these two programs is analysed under the following hypothesis.

- The high input A is uniform and publicly-known. Thus

$$\forall a \in \mathcal{A}\, p_\rho(a) = p_\beta(a) = \frac{1}{|\mathcal{A}|}$$

- The adversary believes that her extra info is accurate, that is she assumes the correlation shown in table below.
  Thus $\Gamma_0 = \{0, 2\}$ and $\Gamma_1 = \{1, 3\}$.

- The real correlation between $A$ and $B$ is of the form of the matrix shown in table below. It is easy to see that the adversary's initial belief is therefore $c$-accurate.

- $B$ and $O$ are independent.

| $p_\beta(b\|a)$ | $b_0$ | $b_1$ | $p_\rho(b\|a)$ | $b_0$ | $b_1$ |
|---|---|---|---|---|---|
| $a_0$ | 1 | 0 | $a_0$ | $\frac{c}{1+c}$ | $\frac{1}{1+c}$ |
| $b_1$ | 0 | 1 | $a_1$ | $\frac{1}{1+c}$ | $\frac{c}{1+c}$ |
| $b_2$ | 1 | 0 | $a_2$ | $\frac{c}{1+c}$ | $\frac{1}{1+c}$ |
| $b_3$ | 0 | 1 | $a_3$ | $\frac{1}{1+c}$ | $\frac{c}{1+c}$ |

We denote by $IU_x$ the initial uncertainty computed using approach $x \in \{c, v, bv\}$ where $c$, $v$ and $bv$ denote the consensus, vulnerability and belief-vulnerability approaches respectively.
We begin by `PROG 1` of Example 1. Since $A$ is uniformly distributed then $IU_c = IU_v = log|\mathcal{A}| = 2$. Furthermore, $RU_v = log(|\mathcal{A}|/|\mathcal{O}|) = log\frac{4}{2} = 1 = RU_c$ since `PROG 1` is deterministic.

**Theorem 1:**

$\vdash \forall$X p.
(random_variable X p Borel) $\wedge$
($\forall$x. x $\in$ X($\Omega$)) $\Rightarrow$ (distribution p X {x} = $\frac{1}{|X(\Omega)|}$) $\wedge$
FINITE ($\Omega$) $\wedge$
IMAGE X (p_space p) = {0;1;2;3} $\Rightarrow$
min_entropy X p = 2

**Theorem 2:**

$\vdash \forall$X Y p c.
FINITE ($\Omega$) $\wedge$ (X($\Omega$) = {0;1;2;3}) $\wedge$
(Y($\Omega$) = {1;2}) $\wedge$ (random_variable X p Borel) $\wedge$
(random_variable Y p Borel) $\wedge$
($\forall$x. x $\in$ $\Omega$ $\Rightarrow$ {x} $\in$ events p) $\wedge$
($\forall$x. x $\in$ X($\Omega$) $\Rightarrow$ (distribution p X {x} = $\frac{1}{|X(\Omega|}$)) $\wedge$
deterministic_cond Y c $\Rightarrow$
conditional_min_entropy p X Y = 1

Thus, when we do not take into account the attacker's belief, then $IL_c = IL_v = 1$.

**Theorem 3:**

$\vdash \forall$X Y p.
FINITE ($\Omega$) $\wedge$ (X($\Omega$) = {0;1;2;3}) $\wedge$
(Y($\Omega$) = {1;2}) $\wedge$ random_variable X p Borel $\wedge$
random_variable Y p Borel $\wedge$
$\forall$x. x $\in$ ($\Omega$) $\Rightarrow$ {x} $\in$ events p) $\wedge$
$\forall$x. x $\in$ X($\Omega$) $\Rightarrow$ (distribution p X {x} = $\frac{1}{|X(\Omega)|}$) $\wedge$
deterministic_cond Y c $\Rightarrow$
information_leakage p X Y = 1

Now let consider the uniformly $c$-accurate attacker's belief. Then we have $IU_{bv} = -log(\frac{c}{2(c+1)})$. And then we get $RU_{bv} = -log(\frac{c}{1+c})$. Therefore for all $c$, $IL_{bv} = 1$. Thus, the adversary's initial knowledge about the parity of $A$ does not affect the quantity of information leaked by `PROG 1`.
However, the real question is not how much information is leaked by this program, but what the remaining uncertainty represents in term of security threat to the high input. Even though the adversary's belief does not affect the quantity of information leaked, it dramatically affects both the initial and remaining uncertainty. As a consequence from that we deduce that inaccurate beliefs strengthen the security of the program (by confusing the adversary), whilst accurate beliefs may weaken it. Thus, a deliberate randomization of the parity of the high input in order to confuse the adversary is a good strategy to strengthen the security of this program.
We continue our analysis with `PROG 2` of Example 2 which is a slight modification of `PROG 1`. Again $IU_c = IU_v = log|\mathcal{A}| = 2$ and $IU_{bv} = -log(\frac{c}{2(c+1)})$. For the remaining uncertainty we have $RU_c = 0.585, RU_v = 0.415$ and $RU_{bv} = -log(\frac{2c+1}{2(1+c)})$. Therefore, $IL_c = 1.415, IL_v = 1.585$ and $IL_{bv} = log(\frac{2c+1}{c})$ .

Unlike `PROG 1`, the information leakage of this program can be arbitrary high when the inaccuracy of the adversary's belief is high whilst its remaining uncertainty $RU_{bv}$ remains very low even for inaccurate beliefs. As already noticed in Example 2, this program leaves $A$ highly vulnerable of being guessed and a deliberate padding of $A$ in order to confuse the adversary is of little help. It means that highly inaccurate beliefs slightly strengthen the security of `PROG 2`.

## 5.2 Leakage in Cascade Channels

After dealing with small attacks scenarios in the previous section, we will present in this section a very important application. The major goal is to reason about the information flow of channels in cascade and analyze the leakage in such systems. We will first expose the notions of channels and cascade of channels. We will then show how to measure the quantity of information using our Min-Entropy theory developed previously and we will finally verify the property related to the information leakage in a cascade of channels.

### 5.2.1 Channels and Cascade of Channels

#### Channels

As defined in [6], the channel is a triplet $(\mathcal{A}, \mathcal{B}, \mathcal{C}_{\mathcal{AB}})$, where $\mathcal{A}$ is defined as a finite set of the critical inputs, $\mathcal{B}$ the observable output and $\mathcal{C}_{\mathcal{AB}}$ is the channel matrix representing the transitional probabilities from the input to the output of the channel. In order to make it easier we are going to use a function that maps this matrix which is the conditional probability of obtaining the output $b$ such that the input is $a$.
In HOL our definition will be as follows
**Definition 5.2.1.** *(Channel)*

```
⊢ ∀A A B p f.  channel p X Y f =
  (random_variable A p Borel) ∧
  (random_variable A p Borel) ∧
  ∀a b.  (a ∈ (IMAGE A (p_space p))) ∧
         (b ∈ (IMAGE B (p_space p))) ∧
      (f(a,b) = (conditional_distribution p B A (b,a)))))
```

Noting the property that for every row the sum of probabilities is 1, we proved using our definitions that for every input $a \in \mathcal{A}$,

$$\sum_{b \in \mathcal{B}} P(B = b | A = a) = 1.$$

The formalization of this theorem in Higher Order Logic environment is

**Theorem 1:**

```
⊢ ∀ p A B a.  FINITE (p_space p) ∧
  p_space p ≠ ∅ ∧ random_variable A p Borel ∧
  random_variable B p Borel ∧
  ∀x.  x ∈ p_space p ⇒ {x} ∈ events p ∧
  0 < distribution p A a ∧
  events p = POW(p_space p) ⇒
  ∑_{b∈B} conditional_distribution p B A (b,a)) = 1
```

**Cascade of Channels**

We talk about cascade of channels when we have a communication system composed of two channels in sequence, $(\mathcal{A}, \mathcal{C}', \mathcal{C}_{\mathcal{AC}'})$ and $(\mathcal{C}', \mathcal{B}, \mathcal{C}_{\mathcal{C}'\mathcal{B}})$ where the outputs of the first one are the inputs of the second one. In such systems, the

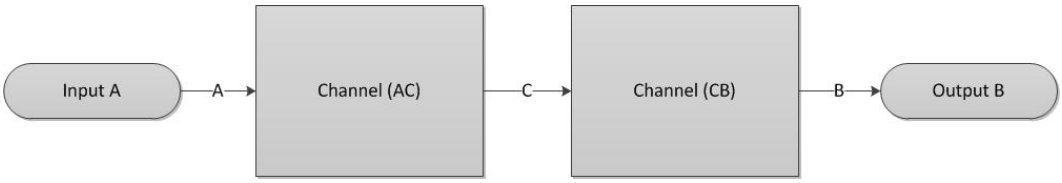


Figure 5.1: Channels in Cascade

final output is produced in two steps, it first passes through the first channel then flows through the second one. We then can say that the whole system behavior is controlled by the first channel, in terms of quantity of information. By definition, the cascade of channels $(\mathcal{A}, \mathcal{C}', \mathcal{C}_{\mathcal{AC}'})$ and $(\mathcal{C}', \mathcal{B}, \mathcal{C}_{\mathcal{C}'\mathcal{B}})$ is the channel $(\mathcal{A}, \mathcal{B}, \mathcal{C}_{\mathcal{AC}'} * \mathcal{C}_{\mathcal{C}'\mathcal{B}})$.

**Definition 5.2.2.** *(Cascade Channel)*

```
⊢ ∀A A C B p f.  cascade_channel p A C B f g =
  channel p A C f ∧
  channel p C B g ∧
  ∀a b.  joint_distribution p A B (a,b) =
    ∑c joint_distribution p A C ({a}, {c}) *
    conditional_distribution p B C ({b}, {c})
```

In this definition the cascade condition is expressed by the last equation since the matrix channel of the system is the multiplication of the two channel matrices.

### 5.2.2 Measuring Information Flow using Min-Entropy

Let $(\mathcal{A}, \mathcal{B}, \mathcal{C}_{\mathcal{AB}})$ be a channel, $A$ and $B$ are 2 random variables modeling respectively the secret input and the observable output, and let $\mathcal{A}$ be an adversary who is trying to attack the system and get the secret $\mathcal{A}$. As we mentioned previously, the amount of information flowing from $\mathcal{A}$ to $\mathcal{B}$ considering the fact that $\mathcal{A}$ is observing the output $\mathcal{B}$ can be expressed as follows:

*leakage = initial uncertainty - remaining uncertainty*

In our project, we are considering the worst case scenario, $\mathcal{A}$ will recover the critical information in one guess. So the Min-Entropy theory is used in order to quantify information flow. The a priori distribution will be modeled as a function of the maximum input distribution and the a posteriori behavior is expressed as a function of the maximum over $\mathcal{A}$ of the distribution of guessing $a$ such that observing $b$ as well as the initial distribution. Thus our schemas will be

$$\textit{leakage = min-entropy(A) - conditional min-entropy(A|B)}$$
$$IL_\infty(A,B) = H_\infty(A) - H_\infty(A|B)$$

### 5.2.3 Leakage in Cascade Channel

After presenting the foundations of channels, cascade channels and the measures we are going to use to quantify the information flow, we will now show how the min-entropy leakage behave in a cascade of channels. Considering the structure of such system, two channels in cascade, we can anticipate that the maximum quantity of information leaked through the system can not exceed the leakage of the first channel. We then prove this property in the following theorem.

**Theorem 2:**
*Let* $(\mathcal{A},\mathcal{B},\mathcal{C}_{\mathcal{AB}})$ *be the cascade of* $(\mathcal{A},\mathcal{C}',\mathcal{C}_{\mathcal{AC}'})$ *and* $(\mathcal{C}',\mathcal{B},\mathcal{C}_{\mathcal{C}'\mathcal{B}})$*. Then we have* $\mathcal{IL}_\infty(\mathcal{A},\mathcal{B}) <= \mathcal{IL}_\infty(\mathcal{A},\mathcal{C}')$

In the core of Higher Order Logic the previous theorem is expressed as

```
⊢ ∀ p A C B f g.
  cascade_channel p A C B f g ∧
  FINITE (p_space p) ∧
  (p_space p) ≠ ∅ ∧
  events p = POW (p_space p) ∧
  ∀x.  0 < distribution p B {x} ∧
  ∀x.  0 < distribution p C {x} ∧
  (∀x.  x IN (p_space p) ⇒ {x} ∈ events p) ⇒
  information_leakage p A B ≤ information_leakage p A C
```

In order to prove this theorem we used some real analysis and properties of the logarithm function. We simplified our goal until we reached the level of vulnerabilities. In this case our theorem will by in the following form

$$V_\infty(A|B) \leq V_\infty(A|C)$$

We follow up by using the constraints of our goal and the probability theorems previously proved in order to reach our final result. The major steps of the proof of this property are

$$\begin{aligned} V(A|B) &= \sum_b max_a \;\; p(B=b) * p(A=a|B=b) \\ &= \sum_b max_a \;\; p(A=a, B=b) \end{aligned}$$

Using the property of cascade we defined at the beginning of this section we get

$$\begin{aligned} p(A{=}a \;/\; B{=}b) &= \sum_c p(A{=}a \;,\; C{=}c) \;*\; p(B{=}b \;/\; C{=}c) \\ &\leq \sum_c max_a \;\; p(A{=}a \;,\; C{=}c) \;*\; p(B{=}b \;/\; C{=}c) \end{aligned}$$

We replace the previous results in our main goal. After we utilized the property we already proved before of swapping the sums, `EXTREAL_SWAP_SIGMA_SIGMA` and then theorem `conditional_distribution_sums1` saying: *The sum of the conditional distribution over the first state space of the random variable is equal to 1*, we got

$$V(A \;/\; B) \leq \sum_c max_a \;\; p(A{=}a \;,\; C{=}c)$$

When we reach this point, we simply use some real analysis related to the multiplication and division as well as some of definition we got our main goal proved. The property cited above needed almost 850 lines of HOL code with the related theorems.

### 5.2.4 Discussion

Due to the formal nature of the model and the soundness of the mechanical theorem prover, the analysis is guaranteed to be free of approximation and precision errors and thus the results obtained are mathematically precise and confirmed the results of paper-and-pencil based analysis approaches. This precision of analysis is a novelty that, to the best of our knowledge, has not been achieved by any

other existing computer-based probabilistic analysis approaches. In the Definition 6 of the cascade channel behavior, the transition functions, f and g, are general functions that provide generic results. In model checking approach parameters and functions should be specified. Furthermore the result verified in Theorem 9 can be extended to the Min-Entropy analysis of information leakage of $n$ channels in cascade using induction techniques. We can prove that the Min-Entropy leakage of $n$ channels in cascade will not exceed the leakage of the first channel. The main key to verify this property is the definition of the cascade condition. Mathematically, we can express the connection of $n$ channels as follows

Let $X_0$ be the random variable modeling the input of the system and $X_n$ the one modeling the output, thus

$\forall$ i. $(0 \leq \text{i} \leq \text{n}) \Rightarrow \text{P}(X_0,X_i) = \sum_{\mathcal{X}_{i-1}} \text{P}(X_0,X_{i-1})$ * $\text{P}(X_i|X_{i-1})$

Based on what we defined previously and what already existed, this condition can be formalized in HOL4 as

```
⊢ ∀X p f n.  n_cascade_channel p X n f =
   ∀i.  (1≤i≤n) ⇒ channel p (X (i−1)) (X i) (f i) ∧
   ∀x y i.  joint_distribution p (X 0) (X i) (x,y) =
      ∑ joint_distribution p (X 0) (X (i−1)) (x,z) *
      z
      conditional_distribution p (X (i−1)) (X i) (z,y)
```

The ability to express and verify generic properties, quantified for all values of the variables, is the main strength of theorem proving as can be seen from the above definition and the property related to the information leakage of n channels in cascade. This property is an ongoing task, once verified, can hold for any number of cascade of channels and can be specialized to obtain expression and values for particular scenarios. Probabilistic model checking, which is the other main stream formal method, cannot provide such generic results due to the inherent state-space explosion problem.

# Chapter 6

# Conclusions

In this project, we tried to focus on a specific threat model: the expected probability that an adversary could guess the secret input value in one try, given the observable output. We used for that model a new definition based on vulnerability, belief-vulnerability, min-entropy and belief min-entropy instead of the traditional consensus definitions of quantitative information flow based on Shanon entropy which do poorly with the threat model in question.

This project was conducted in the environment of HOL theorem proving in order to overcome the different limitations of simulation and paper and pencil techniques. This approach provides accurate results and a minimum of errors that could occur due to human behavior or the us of a lot of approximations.

The main purpose of that project is to formally analyse the information flow in different programs and protocols by evaluating the security of confidential information. We used for the analysis two approaches, one based on the min-entropy and the other based on the belief min-entropy. In the first model we defined the initial uncertainty by the min-Entropy `H`and the remaining uncertainty by the conditional min-entropy, that means the fact of guessing `H` given `L`, where `L` maps the low output. After formalizing those two notions in HOL, we verified different results on how to calculate vulnerability which seems encouraging, especially for the special case of a deterministic program mapping a uniformly distributed `H` to an output `L`. For there we found that the leakage is simply $log|\mathcal{L}|$.

The second approach incorporates the attacker's beliefs. We tried to investigate the impact of such extra knowledge on the security of the secret information. The

analysis in the core of Higher-Order-Logic theorem proving reveals that inaccurate extra information tends to confuse the adversary by increasing his uncertainty about the hidden secret while accurate information may increase its vulnerability. We also showed the strength of the proposed definitions both theoretically and by applying them to various threat scenarios.

In order to point out the usefulness of our theory, we applied our formalizations into an case study, the information flow in the cascade of channels. This kind of system is a very interesting structure since it is a small model of most of the real communication systems such us protocols and networks. The information flows through a channels in cascade in two steps such that the output of one channel will be the input of the next channel. In this application we verified the Min-Entropy information leakage and we proved that the leakage of a cascade of channels cannot exceed the leakage of the first channel. This result could be extended to tackle systems with deterministic behavior and prove that the vulnerability such systems will only depend on the vulnerability of the second channel.

Due to the formal nature of the model and the soundness of the mechanical theorem prover, the analysis is guaranteed to be free of approximation and precision errors and thus the results obtained are 100% precise and confirmed the results of paper-and-pencil based analysis approaches. This precision of analysis is a novelty that, to the best of our knowledge, has not been achieved by any other existing probabilistic analysis approaches.

The proposed higher-order-logic theorem proving based probabilistic analysis approach could be very useful in different future directions. The reasonableness of these definitions could be assessed in different kind of applications having a threat scenarios. We are aiming to apply it on the Crowds protocol [17] and maybe further directions such as Freenets [13]. Both of these applications present a threat scenario, we try to use our model in order to analyse the threat and verify some results related to the quantification of the information leaked and then based on the results of analysis we can take the appropriate measures that offer better way to evaluate the security of the confidential information.

As a future direction, our work could be developed to analyze information flow in a way that starting from a specific bound of information leakage that shouldn't be exceeded it evaluates the input set considering the output set. This work could be conducted in order to ensure a specific level of security of critical information.

# Bibliography


[1] C. E. Brown. *Automated Reasoning in Higher-Order Logic*. Studies in Logic, Logic and Cognitive Systems. College Publications, 2007.

[2] D. Chaum. The dining cryptographers problem: Unconditional sender and recipient untraceability. *Cryptology*, 1(1):65–75, 1988.

[3] A. Church. A formulation of the simple theory of types. *Journal of Symbolic Logic*, 5:56–68, 1940.

[4] A. R. Coble. Anonymity, information, and machine-assisted proof. Technical report, University of Cambridge, Computer Laboratory, Cambridge UK, July 2010.

[5] O. Grumberg E. Clarke and D. Long. Tools for finite state concurrent. *Formal Methods in System Design*, 1(2-3):151–238, 1992.

[6] B. Espinoza and G. Smith. Min-entropy leakage of channels in cascade. In *Formal Aspects in Security and Trust*, volume 7140 of *LNCS*, pages 70–84. Springer, 2011.

[7] M.J.C. Gordon. *Mechanizing programming logics in higher order logic*. University of Cambridge, Computer Laboratory, 1988.

[8] J.Y. Halpern and K.R. O'Neill. Anonymity and information hiding in multiagent systems. *Journal of Computer Security*, 13(3):483–514, 2005.

[9] S. Hamadou, V. Sassone, and C. Palamidessi. Reconciling belief and vulnerability in information flow. In *Proceedings IEEE Symposium on Security and Privacy*, pages 79–92. IEEE Computer Society, 2010.

[10] G. Helali. http://hvg.ece.concordia.ca/projects/prob-it/min_beliefInfo.php.

[11] J. Hölzl. *Construction and Stochastic Applications of Measure Spaces in Higher-Order Logic*. PhD thesis, Institut für Informatik, Technische Universität München, October 2012.

[12] J. Hölzl and Armin Heller. Three chapters of measure theory in Isabelle/HOL. In Marko C. J. D. van Eekelen, Herman Geuvers, Julien Schmaltz, and Freek Wiedijk, editors, *Interactive Theorem Proving (ITP 2011)*, volume 6898 of *LNCS*, pages 135–151, 2011.

[13] B. Wiley T.W. Hong I. Clarke, O. Sandberg. Freenet: A distributed anonymous information storage and retrieval system. In *Designing Privacy Enhancing Technologies*, volume 2009 of *LNCS*, pages 46–66. 2001.

[14] James L. Massey. Guessing and entropy. In *In Proceedings of the 1994 IEEE International Symposium on Information Theory*, page 204, 1994.

[15] T. Mhamdi, O. Hasan, and S. Tahar. Formalization of entropy measures in hol. In *Interactive Theorem Proving*, volume 6898 of *LNCS*, pages 233–248. Springer, 2011.

[16] R. Milner. A Theory of Type Polymorphism in Programming. *J. Comput. Syst. Sci.*, 17(3):348–375, 1978.

[17] Michael K. Reiter and Aviel D. Rubin. Crowds: anonymity for web transactions. *ACM Transactions on Information Systems Security*, 1(1):66–92, 1998.

[18] S. Schneider and A. Sidiropoulos. CSP and anonymity. In *Proceedings of the European Symposium on Research in Computer Security: Computer Security*, volume 1146 of *LNCS*. Springer, 1996.

[19] G. Smith. Quantifying information flow using min-entropy. In *Quantitative Evaluation of SysTems*, pages 159–167, 2011.

[20] P.F. Syverson, D.M. Goldschlag, and M.G. Reed. Anonymous connections and onion routing. In *In IEEE Symposium on Security and Privacy*, pages 44–54, Oackland, California, 1997.