# What is the definition of Perfect security?

**Definition 5.1:** A cryptosystem has perfect security if for all $x \in \mathcal{P}$ and $y \in \mathcal{C}$, it holds that $P[x|y] = P[x]$.
**TLDR:** Information about the ciphertext gives you *no* information about the plaintext.

# What are the requirements in order to acheive Perfect Security?

**Theorem - $|\mathcal{K}| \geq |\mathcal{C}| \geq |\mathcal{P}|$:** If you have perfect security then $|\mathcal{K}| \geq |\mathcal{C}| \geq |\mathcal{P}|$.
**TLDR:** If you have perfect security your key can not be shorter than your ciphertext, which cannot be shorter than your plaintext.

# What is Entropy?

**Definition 5.6:** Let $X$ be a random variable that takes values $x_1, ..., x_n$ with probabilities $p_1, ..., p_n$. Then the entropy of $X$, written $H(X)$, is defined to be:

$$H(X) = \sum_{i=1}^{n} p_i \log_2(1/p_i)$$

**TLDR:** If an event $A$ occurs with probability $p$ and you are told that $A$ occurred, then you have learned $\log_2(1/p)$ bits of information. **TLDR:** The entropy $H(X)$ can be described as:

- How many bits we need to send on average to communicate the value of $X$.
- The amount of uncertainty you have about $X$ before you are told what the value is.

# What are the bounds for Entropy?

**Theorem 5.7:** For a random variable $X$ taking $n$ possible values, it holds that $0 \leq H(X) \leq \log_2(n)$. Furthermore, $H(X) = 0$ **iff** one value $X$ has probability 1 (and the others 0). $H(X) = log_2(n)$ **iff** it is uniformly distributed, i.e., all probabilities are $1/n$.
**TLDR:** If the entropy of $X$ is 0 there is no uncertainty, meaning that we know the value of $X$. If the entropy of $X$ is 1 then the uncertainty of $X$ is highest meaning that all possible values of $X$ have the same probability.

## What is the definition for Conditional Entropy?

**Definition 5.9:** Given the above definition of $H(X \mid Y = y_j)$, we define the conditional entropy of X given Y to be:

$$H(X \mid Y) = \sum_j P[Y = y_j] H(X \mid Y = y_j)$$

## For deterministic cryptosystems, what is the entropy of the key given the ciphertext ($H(K \mid C)$)?

**Theorem 5.11:** For any cryptosystem with deterministic encryption function, it holds that:

$$H(K \mid C) = H(K) + H(P) - H(C)$$

**TLDR:** Answers how much uncertainty remains about the key given the ciphertext

## What is Redundancy in a language

**Definition - Redundancy:** Given a language $L$ and a plaintext space $\mathcal{P}$, the *redundancy* of the language is the amount of superflous information is contained, on avarage in the language $L$.

$$R_L = \frac{\log(|\mathcal{P}|) - H_L}{\log(|\mathcal{P}|)} = 1 - \frac{H_L}{\log(|\mathcal{P}|)}$$

$H_L$ is a measure of the number of bits of information each letter contains in the language $L$, on average. For English, we have that $H_L$ is (very approximately) 1.25 bits per letter.

$$H_L = \lim_{n \mapsto \infty} H(P_n)/n$$

**TLDR:** A language contains redundancy, which is how much duplicate information there is on avarage in the language.
**Example:** The following sentance displays redundancy in english:

> *"cn y rd th fllwng sntnc, vn f t s wrttn wtht vcls?"*

## What is the definition for Spurious Keys?

**Definition - Spurious Keys:** If an adversary has a ciphertext $y$ that he wants to decrypt, he can try all keys and see if $y$ decrypts to meaningful english. If $y$ decrypts to meaningful english under the *wrong* key, then that key is said to be a *spurious key.*
**TLDR:** A spurious key is a key that *seems* to be the correct key for a ciphertext but is not.

## What is the formula for the number of Spurious Keys?

**Definition - Number of Spurious Keys:** The average number of spurious keys, taken over all choices of ciphertexts of length $n$:

$$sp_n = \sum_{\boldsymbol{y} \in \mathcal{C}^n} P[y]|K(\boldsymbol{y})| - 1 = \sum_{\boldsymbol{y} \in \mathcal{C}^n} P[y]|K(\boldsymbol{y})| - 1$$

Given a ciphertext $\boldsymbol{y}$, we use $K(\boldsymbol{y})$ to denote the set of keys that are possible given this ciphertext. More precisely, a key $K$ is in this set if decryption of $\boldsymbol{y}$ under $K$ yields a plaintext that could occur with non-zero probability:

$$K(\boldsymbol{y}) = \{K \in \mathcal{K} \mid P[D_K(\boldsymbol{y} > 0)]\}$$

**TLDR:** This formula for $sp_n$ describes the average number of spurious keys of a ciphertext $\boldsymbol{y}$ of length $n$.

## What is the definition for Unicity Distance?

**Definition 5.12:** The unicity distance $n_0$ of a cryptosystem is the minimal length of plaintexts such that $sp_{n_0} = 0$, if such a value exists, and $\infty$ otherwise. **TLDR:** The unicity distance tells you how many times you can encrypt something where multiple keys seem to be valid keys.

## For a deterministic cryptosystem, what is the bound for the Unicity Distance?

**Theorem 5.13:** Assume we have a cryptosystem with deterministic encryption function, where the plaintext and ciphertext alphabets have the same size ($|\mathcal{C}| = |\mathcal{P}|$), and where keys are uniformly chosen from $\mathcal{K}$. Assume we use the system to encrypt sequences of letters from language $L$. Then

$$n_0 \geq \frac{\log(|\mathcal{K}|)}{R_L \log(|\mathcal{P}|)}$$

**TLDR:** If we reuse keys, our unconditional security will always be gone, once we encrypt enough plaintext under the same key. The only exception is the case where $R_L = 0$ which leads to $n_0$ being $\infty$. Which makes sense, if every sequence of characters is a plaintext that can occur, the adversary can never exclude a key.