# Security Lab – Work Factor

## VMware

No VM or programming is needed.

## 1 Introduction

In this lab, you will compute various work factors. In particular, first you are presented with the work factor concept and then you study the influence of non-uniformly distributed keys on the difficulty of guessing a key by trial and error. Secondly, you investigate some properties of a secret key's work factor based on a simple case study.

**Note**: If a number without decimal places is given in this practical, it is the exact value. Example: 0, 4, 17/2. If a number with decimal places is specified, the correct value is rounded to the specified number of decimal places. Example: correct value 1.9994872 becomes 2.00.

**Note**: Some calculations are required in this lab. Often the correct result is already indicated in the exercises. This should help you to check your calculation for correctness. This is because the assessment of the task is more about whether you have done the calculation correctly than whether you can present the numerically correct result. In Exercise 1, for example, you are asked to prove that the sum of certain probabilities given in a table is 1. Since the given sum is given as "1" and not as "1.00", it is clear that the result must be *exactly* 1 and not just *approximately* 1. Furthermore, it is not sufficient to simply write as an answer for example $\sum_{k=1}^{16} 1/16 = 1$ as an answer, because this only reformulates the question. Furthermore, it is not acceptable to enter the equation into Wolfram Alpha or a similar system and accept the answer unchecked. You must do the calculation yourself, by hand, and as long as possible without the aid of a calculator.

**Note**: Two equations helpful for this lab are.

$$\sum_{k=0}^{n} x^k = \frac{x^{n+1} - 1}{x - 1}$$

$$\sum_{k=0}^{n} k = \frac{n(n+1)}{2}$$

## 2 Work Factor – The Good, The Bad ~~and The Ugly~~ ...

The Work Factor denotes the average number of attempts needed to guess a secret. Formally let $X = \{x_1, \dots, x_n\}$ is a finite set of possible outcomes of an experiment. This can be, for example, the six sides of a dice, which can lie on top after a throw; the two sides of a coin; the used keys of a cryptosystem or the different blocks of a ciphertext. According to the random oracle model these blocks should be chosen randomly. Let it be now further for $1 \le i \le n$ with $p_i$ denotes the probability that $x_i$ appears as the result of the experiment. For a fair die, for example $X = \{1,2,3,4,5,6\}$ and it is because of fairness $p_i = 1/6$. If the die is not fair, then $X$ is unchanged, but the $p_i$ are then no longer all equal. If one now wants to guess the result of the experiment, and if one does this in the order $x_1, \dots, x_n$then the Work Factor is the expected number of trials:

$$\mathrm{WF}(X) = \sum_{k=1}^{n} k\, p_k.$$

The Work Factor is sometimes specified in bits. The Work Factor in bit is $\log_2 \mathrm{WF}(X)$.

In the following we consider an unspecified toy encryption with 4 bit key length. You now do experiments with two different systems G (for *good*) and B (for *bad*) and find that the different keys in the two systems are encrypted with different probabilities $p_{i,G}$ and $p_{i,B}$ are selected:

| Key | $p_{i,G}$ | $p_{i,B}$ | Key | $p_{i,G}$ | $p_{i,B}$ |
|---|---|---|---|---|---|
| 0000 | 1/16 | $1/2^1$ | 1000 | 1/16 | $1/2^9$ |
| 0001 | 1/16 | $1/2^2$ | 1001 | 1/16 | $1/2^{10}$ |
| 0010 | 1/16 | $1/2^3$ | 1010 | 1/16 | $1/2^{11}$ |
| 0011 | 1/16 | $1/2^4$ | 1011 | 1/16 | $1/2^{12}$ |
| 0100 | 1/16 | $1/2^5$ | 1100 | 1/16 | $1/2^{13}$ |
| 0101 | 1/16 | $1/2^6$ | 1101 | 1/16 | $1/2^{14}$ |
| 0110 | 1/16 | $1/2^7$ | 1110 | 1/16 | $1/2^{15}$ |
| 0111 | 1/16 | $1/2^8$ | 1111 | 1/16 | $1/2^{15}$ |

**Exercise 1.** As with any probability distribution, the sum of the probabilities should add up to one. Verify this for G and B. $\swarrow \cdot 16$

G: $\frac{1}{16} + \frac{1}{16} + \frac{1}{16} \dots \frac{1}{16} = 1$

B: $\frac{1}{2^1} + \frac{1}{2^2} + \frac{1}{2^3} + \frac{1}{2^4} \dots + \frac{1}{2^{14}} + \frac{1}{2^{15}} + \frac{1}{2^{15}} = 1$

**Exercise 2.** Now put yourself in the position of an attacker on System G. You want to crack the system, but for that, you need to try keys one by one. Justify why the order in which you try the keys has no influence on the expected number of samples you need until you have found the right key.

Weil die Wahrscheinlichkeiten für jeden Schlüssel gleich sind! $\left(\frac{1}{16}\right)$

**Exercise 3.** Now calculate the work factor of G (correct answer: 17/2). Calculate this work factor also in bits (correct answer: 3.09).

$WF(x) = \sum_{i=1}^{n} k \cdot p_k = 1 \cdot \frac{1}{16} + 2 \cdot \frac{1}{16} + 3 \cdot \frac{1}{16} + \dots + 14 \cdot \frac{1}{16} + 15 \cdot \frac{1}{16} + 16 \cdot \frac{1}{16} = \underline{\underline{\frac{17}{2}}}$

$\log_2 WF(x) = \log_2 \left(\frac{17}{2}\right) = 3.0875 = \underline{\underline{3.09}}$

So the keys of G have about 3.1 bit work factor.

**Exercise 4.** Now put yourself in the position of an attacker on system B. You want to crack the system again, but to do so you have to try out keys one after the other and you want to spend as little time as possible on the problem. Justify why the strategy of trying out keys in descending order of probability promises good success.

Das System B hat unterschiedliche Wahrscheinlichkeiten für ihre Schlüssel. In aufsteigender Form nehmen die Wahrscheinlichkeiten zu. 0000 hat die kleinste Wahrscheinlichkeit ⇒ mit 0000 anfangen!

**Exercise 5.** Now calculate the work factor of B (correct answer: 2.00). Calculate this work factor also in bits (correct answer: 1.00).

$$\sum_{i=0}^{n-1} k \cdot p_k = \left(1 \cdot \frac{1}{2^1} + 2 \cdot \frac{1}{2^2} + 3 \cdot \frac{1}{2^3} + 14 \cdot \frac{1}{2^{14}} + 15 \cdot \frac{1}{2^{15}}\right) + 16 \cdot \frac{1}{2^{15}}$$

$$= 1.999969 = \underline{\underline{2.00}}$$

$$\log_2(1.999969) = 0.99997738 = \underline{\underline{1.00}}$$

System B therefore has only about 1.0 bit work factor.

We have thus seen that, depending on the distribution of the keys and therefore naturally also depending on the attacker's level of knowledge, a system can have a significantly lower work factor than the value to be expected based on the key length alone.

**Exercise 6.** Now make a conjecture for which distribution of the keys the work factor is the largest. A justification is not necessary, but your conjecture must fit the observed facts.

Alle Schlüssel mit gleicher Wahrscheinlichkeit!

**Exercise 7.** Ideally, an encryption transforms a plaintext into a ciphertext that is indistinguishable from purely random text. In this case let $X = \{0,1\}$ be the set of bits appearing in the text. What is true for the probabilities $p_0$ and $p_1$ with which a zero- or one-bit occurs? What is the work factor in bit of one bit of the plaintext in this case (correct result: about 0.6 bit)?

$p_0 = 0.5 \qquad p_1 = 0.5$

$WF(X) = 1 \cdot 0.5 + 2 \cdot 0.5 = 1.5$

$\log_2(1.5) = 0.585 = \underline{\underline{0.6}}$

## 3 Work Factor in Secret Key Cryptography

Recall that the work factor of a problem is defined as the average number of tries one needs to do in order to solve that problem by trial and error. In this part, we will study the work factor for keys in secret key cryptography and see its meaning in terms of work, i.e., time needed for finding keys.

**Exercise 8.** We start with the general case. Assume that you want to find the key for a ciphertext by trying out the various possible keys. There are N keys.

a) If those keys are *uniformly distributed* (each key is equally probable), compute the work factor. (The correct answer is, (N+1)/2).
b) Argue that if N is very large, it does not matter numerically whether we use (N+1)/2 or N/2.

$$EF = 1 \cdot \frac{1}{N} + 2 \cdot \frac{1}{N} + \dots N \cdot \frac{1}{N} = \frac{1}{N} \cdot \underbrace{\sum_{k=1}^{N} k}_{\frac{\cancel{N}(N+1)}{2} = \frac{N+1}{2}}$$

b) Work Factor ist so gross, macht keinen Unterschied mehr.

**Exercise 9.** Now let's look into a specific case: We know that the cipher for the key that the attackers want to break is simply "AES" (with no extensions to the name). Look up AES's default key length. Assuming the key was chosen uniformly at random, what is that key's work factor? Also give the work factor in bits. Show and explain your work.

Key Length = 128 Bit

$$WF(AES) = \frac{2^{128}}{2} = 2^{127}$$

$$\log_2 (WF(AES))$$

**Exercise 10.** Make a *reasonable assumption* about how fast your computer can try a single key on a file of 512 bytes. You should do this by looking up your computer's clock speed and then looking up approximate values for "cycles per byte" for AES on your CPU. Cycles per byte is a standard performance figure for cryptographic speed. Then compute how long that computer would need to exhaust the work factor. Can you get it done by Friday? How about next Friday? **Important note:** If you can't find the cycles per byte for your particular computer, use the numbers for *any* modern CPU. If you're stuck, one reasonable answer can be found on Wikipedia. Even then, *absolute precision is not essential, order-of-magnitude estimates will suffice*. If you still don't know which of the numbers to choose, choose the one that means that your CPU is faster (higher cycles per second, lower cycles per byte).

3.6 GHz = Processor clock frequency
512 byte = Message length

bytes per second = 2200 MB/s
= 2.2 GB/s

$$\frac{3.6\ GHz}{2.2\ GB/s} = \frac{3.6 \cdot 10^9}{2.2 \cdot 10^9} \approx 1.6 \text{ Cycles per Byte}$$

$1.6 \cdot 3.6 \cdot 10^9$ = Bytes per second

$\frac{512}{1.6 \cdot 3.6 \cdot 10^9} =$

**Exercise 11.** As the previous calculation shows, you cannot hope to break the key by brute force alone. What assumptions would have to change so that you have a good chance of recovering the key nevertheless?

- (WF lower)
- More Performence (Speed)

**Exercise 12.** Compute the maximum work factor that the key could have so that your computer (under the assumptions from above) could break the key in 1 minute.

## Lab Points

You can earn **2 lab points** in this lab**:**

- You will receive two points if you show the supervisor your answers to the questions in the internship guide and these answers are mostly correct. You must also answer any control questions from the supervisor correctly.