Journal of Applied Mathematics & Bioinformatics, vol.3, no.1, 2013, 181-191
ISSN: 1792-6602 (print), 1792-6939 (online)
Scienpress Ltd, 2013

Minimum key length for cryptographic security

George Marinakis[1]

Abstract

The security of a symmetric cryptographic algorithm depends on the strength of the algorithm and the length of the cryptographic key. When the algorithm does not have a known and exploitable flaw in its internal structure, the only cryptanalytic attack that can be applied to it is the method of Exhaustive Key Search (Brute Force Attack). This process is extremely time consuming and if the cryptographic key has an adequate length, then the Exhaustive Key Search is practically inapplicable and therefore we say that the algorithm is practically secure. In this study, we examine the various parameters which influence the time for the Exhaustive Key Search, and based on them we calculate the minimum key length of a symmetric cryptographic algorithm in order to be secure against the cryptanalytic attacks which use the current computer technology (in software and hardware). After this, we calculate the minimum key length for the years in the future, according to the expected technological progress.

Mathematics Subject Classification: Cryptography
Keywords: Algorithm; Key; Exhaustive Key Search; Brute Force Attack

[1] Telecommunications and Electronics School of Military Signal Officers, Athens, e-mail: gmari@tee.gr

Article Info: *Received* : November 1, 2012. *Revised* : January 30, 2013
Published online : March 30, 2013

1 Introduction

In 1996, a study from a group of cryptographers [1], showed that the minimum key length should be 75 bits in order to be secure against the Brute Force Attack for that era and 90 bits in order to be secure for the next 20 years. Since then, many new cryptographic algorithms have been published with more complexity and bigger key length, but also their implementation techniques have been advanced in speed and performance. Therefore, because of these advances in technology, today there is a need for a new estimation of the minimum key length in order the cryptographic algorithms be secure against the Brute Force Attack.

In the beginning of this study, we give the theoretical formulas and then we make the practical calculation of the Brute Force Attack times, for various cryptographic keys of modern algorithms. And in order to be as much practical as possible, in the calculations we use the execution times of existing and published algorithm implementations in software and hardware.

Throughout this study, we do not estimate the cost of the various Brute Force Attack implementations, but we assume that the adversary has the motivation to make the necessary investments in the available state of the art computer technology in order to cryptanalyze the encrypted information. Of course, some big investments (especially those who need massive parallelization of specific hardware) can not be realized by an individual hacker, but they are affordable by a Big Company or an Intelligence Agency. Some characteristic examples concerning the estimated cost of the different Brute Force Attack methods, as well as the estimation of the value of the encrypted information (depending on its kind and the motivation of the adversary), are given in [2].

2 Single Search

The simplest case of Brute Force Attack is the Single Search, in which we use only one at a time algorithm implementation (in software or hardware). In this case, the necessary time for Brute Force Attack (BFA) is :

$$T_{BFA} = T_{MDL} \cdot N = T_{MDL} \cdot 2^L \tag{1}$$

where T_{BFA} is the Brute Force Attack time, T_{MDL} is the time which is needed by the implementation to execute a Main Decryption Loop (MDL) of the algorithm, N is the total number of the keys and L is the length of the key (in bits)[2].

2.1 Software implementation

When the algorithm implementation is done in software, the necessary time for a general purpose computer to execute a Main Decryption Loop (MDL) of the algorithm is :

$$T_{MDL} = C_{MDL} \cdot T_C = \frac{C_{MDL}}{F_{MAX}} \tag{2}$$

where C_{MDL} is the necessary CPU cycles for a MDL, T_C is the duration of each CPU cycle (T=1/F) and F_{MAX} is the current maximum clock speed of the general purpose computers.

Because of equation (2), equation (1) becomes :

$$T_{BFA} = \frac{C_{MDL} \cdot 2^L}{F_{MAX}} \tag{3}$$

Modern tools and methods of software development led to significant decrease of cryptographic algorithms execution time. According to [3], the fastest software implementation of AES-128 algorithm until the year 2008, was 193 CPU cycles for the decryption of one block of data. As far as computer speed is concerned, nowadays the maximum clock frequency of commercial computers is 3 GHz (September 2012). Therefore, if we put $C_{MDL} = 193$ cycles/block and $F_{MAX} = 3 \cdot 10^9$ Hz in the equation (3), we can calculate the Brute Force Attack time (T_{BFA}) when the algorithm is implemented in software, for different values of the key L[3]. With these values of T_{BFA} we created the first

[2]In practice, when we use the Brute Force Attack method it is possible to find the key before we exhaust the total key space. But in this study we consider the worst case, in which we must examine all the key combinations in order to finally decrypt the encrypted message.

[3]For the simplification of the calculations, we assume that the necessary CPU cycles are the same for the different key lengths. In practice, when the key increases, the time needed for the algorithm to run is also increasing. However, these time differences are relatively small and they do not affect the general conclusions of this study.

column of Table 1 (Single Search/Software)[4].

2.2 Hardware implementation

Cryptographic algorithms in hardware can be implemented either in FPGAs (Field Programmable Gate Arrays) or in ASICs (Application Specific Integrated Circuits). A brief description regarding the technology of the above integrated circuits is given in [4], as well as a comparison between them. To sum up that comparison, it can be said that FPGAs are reprogrammable and cheaper, whereas ASICs can not be reprogrammed and are more costly. On the other hand, ASICs are much more faster than FPGAs.

When implementing algorithms in hardware, the time T_{MDL} that needs the computer to execute a Main Decryption Loop (MDL) of the algorithm is called Latency and according to [4], its defined by the following equation:

$$L_{atency} = \frac{B_{lock\_size} \cdot S_{imultaneus\_blocks}}{T_{hroughput}} \tag{4}$$

where:
B_{lock_size} = size of the input block of the algorithm (in bits)
$S_{imultaneous\_blocks}$ = number of blocks which can be processed simultaneously
$T_{hroughput}$ = number of bits encrypted (or decrypted) per unit of time.

If only one block can be processed each time ($S_{imultaneous\_blocks} = 1$), due to (4), (1) converts to:

$$T_{BFA} = L_{atency} \cdot 2^L = \frac{B_{lock\_size} \cdot 2^L}{T_{hroughput}} \tag{5}$$

Today, very fast implementations of cryptographic algorithms have been accomplished in FPGAs and ASICs , where the number of bits encrypted (or decrypted) per time unit is very high. Such an example is the implementation of the AES-128 algorithm on ASIC that is referred in [5], in which a Throughput of 40 Gbps is achieved. If in equation (5) we assign the values :
$B_{lock\_size} = 128$ bits and Throughput $= 40 \cdot 10^9$ bits/sec , then we can calculate the T_{BFA} time for the exhaustive key search when the algorithm is implemented

[4]For indicative reasons, the calculations of Table 1 are done for block ciphers, which are more well known. But the calculated values are proportional to the corresponding values of stream ciphers for the same key length.

in hardware, for various lengths of the cryptographic key L[5]. The second column of Table 1 was created using those values (Single Search/Hardware).

3 Parallel Search

The Brute Force Attack time can be significantly decreased if we use parallelization. This means that we use concurrently many systems which implement the algorithm and we distribute the total number of the keys by giving in each implementation different key values. In this way, the total search time is divided by n, which is the number of the parallel implementations that we use. Today, it is feasible to use one million general purpose computers, or FPGAs, or ASICs, in order to conduct a parallelized Brute Force Attack.

3.1 Software implementation

If we apply a parallel BFA search by simultaneously using n general purpose computers which will be sharing the key values, then (3) becomes:

$$T_{BFA} = \frac{C_{MDL} \cdot 2^L}{n \cdot F_{MAX}} \tag{6}$$

Assigning the following values in the above equation:
$C_{MDL} = 193$ cycles/block, $F_{MAX} = 3 \cdot 10^9$ Hz, n= 1000000,
we can calculate the time needed for the exhaustive key search T_{BFA} for various L lengths of the cryptographic key, with the parallel use of one million computers which implement the algorithm in software. Using the above values, the third column of Table 1 was created (Parallel Search/Software).

[5]In practice, the block size is not the same for all the block ciphers. For example, the 3DES and IDEA algorithms have a block size of 64 bits, but the most modern block ciphers like the three AES final candidates Rijndael, Serpent and Twofish have a block size of 128 bits. For simplicity reasons we use the block size of 128 bits for all the different key calculations in the rows of Table 1. This simplification does not induce a significant difference in the tables values, neither alters the general conclusions of this study.

3.2 Hardware implementation

If we apply a parallel BFA search by simultaneously using n FPGAs or ASICs which will be sharing the key values, then (5) becomes:

$$T_{BFA} = \frac{B_{lock\_size} \cdot 2^L}{n \cdot T_{hroughput}} \tag{7}$$

Assigning in the above equation the values :
$B_{lock_size} = 128$ bits , $T_{hroughput} = 40 \cdot 10^9$ bits/sec and $n = 1000000$,
we can calculate the time needed for the exhaustive key search T_{BFA} for various L lengths of the cryptographic key, with the parallel use of one million integrated FPGA or ASIC circuits that implement the algorithm. Using the above values, the fourth column of Table 1 was created (Parallel Search/Hardware).

Table 1: Minimum BFA Time using current technology

Key (bits)	Minimum BFA Time (in years)			
	Single Search		Parallel Search (10^6)	
	Software	Hardware	Software	Hardware
75	77068788 y	3833475.3 y	77.068 y	3.833 y
90	$2.525 \cdot 10^{12}$ y	$1.256 \cdot 10^{11}$ y	$2.525 \cdot 10^{6}$ y	$1.256 \cdot 10^{5}$ y
128	$6.941 \cdot 10^{23}$ y	$3.452 \cdot 10^{22}$ y	$6.941 \cdot 10^{17}$ y	$3.452 \cdot 10^{16}$ y
192	$1.280 \cdot 10^{43}$ y	$6.369 \cdot 10^{41}$ y	$1.280 \cdot 10^{37}$ y	$6.369 \cdot 10^{35}$ y
256	$2.362 \cdot 10^{62}$ y	$1.174 \cdot 10^{61}$ y	$2.362 \cdot 10^{56}$ y	$1.174 \cdot 10^{55}$ y

4 Future Evolution

Equations (6) and (7) express T_{BFA} using todays computers top performance. Computers performance though is increased throughout the years, since the technology with which the integrated circuits are built is improved. According to Moore's Law, as stated in 1965 [6], the number of transistors in the integrated circuits doubles every year. In 1975 Moore himself restated that the transistor density doubles every two years. In the next years it appeared that the doubling time varies from 18 months to three years. According to several publications such as [7] and [8], the Moore's Law about the doubling

every two years as an average, still holds to the present day and it will still be valid for many more decades to come. This fact comes not only from the expected increase in the number of transistors in the integrated circuits, but also because of the incorporation of new materials, processes and device structures which will be combined with CMOS transistor modules. And if the number of transistors in the integrated circuits will double every two years, this will have as effect that in the same amount of time their performance will be doubled as well (and this is proven in practice). Therefore, with the passing of d years, the speed/performance of the computers will have doubled for $d/2$ times, in terms of a geometrical progress expressed by the following equation:

$$F_d = F_{2012} \cdot 2^{d/2} \tag{8}$$

where F_d is the maximum speed of a general purpose computer after d years and F_{2012} is that maximum speed today (2012).

For the same reasons, the $T_{hroughput}$ in the future hardware implementations will be:

$$T_{hroughput-d} = T_{hroughput-2012} \cdot 2^{d/2} \tag{9}$$

where $T_{hroughput-d}$ is the maximum $T_{hroughput}$ after d years and $T_{hroughput-2012}$ is the maximum $T_{hroughput}$ today (2012).

Finally, because of the Moore's Law we must expect that the number n of the parallel implementations will also increase. This comes from the fact that due to the increased transistor density, the integrated circuits except from becoming faster, they will become smaller in size and cheaper in price. In addition to this, it is expected that in the future there will be significant advances in the parallelization techniques and in computer networking. Therefore, in order to consider a bigger technological evolution (and therefore a bigger risk of cryptanalytic attack) we can assume that the number n will also increase with the same rate like F_d and $T_{hroughput-d}$ and it will be :

$$n_d = n_{2012} \cdot 2^{d/2} \tag{10}$$

where n_d is the maximum number of parallel implementations after d years and n_{2012} is the maximum n today (2012).

Using equations (8) and (10), equation (6) converts to (11). And using equations (9) and (10), equation (7) converts to (12):

$$T_{BFA} = \frac{C_{MDL} \cdot 2^L}{n_{2012} \cdot F_{2012} \cdot 2^d} \tag{11}$$

$$T_{BFA} = \frac{B_{lock\_size} \cdot 2^L}{n_{2012} \cdot T_{hroughput} \cdot 2^d} \tag{12}$$

Assigning in equation (11) the values: C_{MDL}= 193 cycles/block, $F_{2012} = 3 \cdot 10^9$ Hz and $n_{2012} = 1000000$, we calculated the times of Brute Force Attack T_{BFA} with the parallel use of n_d general purpose computers, for various lengths L of the cryptographic key, using different chronic distances from today (d = 30, 50, 70, 90 years) and using the equivalent technology of that era. With the above values of T_{BFA}, Table 2 was created (where y=years, d=days, h=hours, m=minutes, s=seconds).

Similarly, assigning in equation (12) the values: B_{lock_size} = 128 bits , $T_{hroughput} = 40 \cdot 10^9$ bits/sec and $n_{2012} = 1000000$, we calculated the times of Brute Force Attack T_{BFA} with the parallel use of n_d FPGAs or ASICs , for various lengths L of the cryptographic key, using different chronic distances from today (d = 30, 50, 70, 90 years) and using the equivalent technology of that era. With the above values of T_{BFA}, Table 3 was created.

Looking the equations (11) and (12), we see that because of the Moore's Law every year their denominator is multiplied by 2, which means that the T_{BFA} is divided by two. In order to compensate this reduction of T_{BFA} , the numerator of the equations must be also multiplied by two. This means that the key length L must increase by one bit every year (12 months). This is a slightly stricter conclusion than that of reference [1], in which was stated that the key length must increase by one bit every 18 months.

Table 2: Minimum BFA Time while software technology evolves

Key (bits)	Minimum BFA Time			
	n_d parallel computers			
	d=30	d=50	d=70	d=90
75	2.263 s	$2.158 \cdot 10^{-6}$ s	$2.058 \cdot 10^{-12}$ s	$1.963 \cdot 10^{-18}$ s
90	20.6 h	0.07 s	$6.745 \cdot 10^{-8}$ s	$6.433 \cdot 10^{-14}$ s
128	$6.465 \cdot 10^{8}$ y	616.55 y	5.15 h	0.017 s
192	$1.192 \cdot 10^{28}$ y	$1.137 \cdot 10^{22}$ y	$1.084 \cdot 10^{16}$ y	$1.034 \cdot 10^{10}$ y
256	$2.199 \cdot 10^{47}$ y	$2.098 \cdot 10^{41}$ y	$2 \cdot 10^{35}$ y	$1.908 \cdot 10^{29}$ y

Table 3: Minimum BFA Time while hardware technology evolves

Key (bits)	Minimum BFA Time			
	n_d parallel FPGA or ASIC			
	d=30	d=50	d=70	d=90
75	0.112 s	$1.073 \cdot 10^{-7}$ s	$1.024 \cdot 10^{-13}$ s	$9.765 \cdot 10^{-20}$ s
90	61.489 m	$3.518 \cdot 10^{-3}$ s	$3.355 \cdot 10^{-9}$ s	$3.2 \cdot 10^{-15}$ s
128	$3.215 \cdot 10^{7}$ y	30.667 y	15.372 m	$8.796 \cdot 10^{-4}$ s
192	$5.932 \cdot 10^{26}$ y	$5.657 \cdot 10^{20}$ y	$5.395 \cdot 10^{14}$ y	$5.145 \cdot 10^{8}$ y
256	$1.094 \cdot 10^{46}$ y	$1.043 \cdot 10^{40}$ y	$9.952 \cdot 10^{33}$ y	$9.491 \cdot 10^{27}$ y

5 Conclusion

From Table 1 of section 3, we see that even when we use 10^6 parallel hardware implementations of the current technology, a key of 90 bits is enough in order to be protected against the Brute Force Attack today, because it will need $1.25 \cdot 10^5$ years to break the key. But from Tables 2 and 3 of section 4, it is obvious that due to the Moore's Law and due to the massive software and especially hardware parallelization, the Brute Force Attack times can be significantly decreased in the future. From Table 3, we see that if we start a Brute Force Attack in 50 years from now, it will be feasible to break a key of 128 bits in almost 30 years. Also, if we start a BFA in 70 years from now, it will be feasible to break a key of 128 bits in almost 15 minutes and after 90 years from now, it will be feasible to break a key of 128 bits in 0.87 ms. Therefore, if we want to keep our encrypted informations secret for 10, 20 or 30 years, the 128 bits will be enough. But if we want to keep them secret for more than 30 years, the 128 bits key will not be enough.

As we saw in paragraph 4, one practical rule in order to protect the cryptographic algorithms from the technological evolution due to Moore's Law, is to increase their key length by one bit every year. This means that if today (2012) a key length of 90 bits is considered secure, after 50 years (2062) the key must be 135 bits in order to be secure against the Brute Force Attacks of that era.

From the above it is obvious that, although today some modern cryptographic algorithms offer key lengths of 192 and 256 bits (like the AES), these lengths seems to be redundant and excessively big at least for the next 50 years.

Because until that time, the technological evolution in computer software and hardware does not put any serious threat, when the attacker uses the Brute Force Attack method. Of course, all the above discussion will subvert if some very revolutionary technological advance appears in the near future (like the practical exploitation of quantum computers).

As we mentioned in the beginning of this study, the Exhaustive Key Search (Brute Force Attack) is the most time consuming and expensive cryptanalytic attack and it is applied only when the algorithm does not have a known and exploitable flaw in its internal structure. This means that except the strive for increasing the key length, much effort must be done in the area of finding and analyzing the possible weaknesses and backdoors inside the cryptographic algorithms. Because if these weaknesses are properly exploited, they can bypass the most or even the total complexity of the key.

References

[1] M. Blaze, W. Diffie, R.L. Rivest, B. Schneier, T. Shimomura, E. Thompson and M. Wiener, Minimal key lengths for symmetric ciphers to provide adequate commercial security, (January, 1996), www.bsa.org/policy/encryption/cryptographers c.html.

[2] Bruce Schneier, *Applied Cryptography*, John Wiley, New York, 1996.

[3] Daniel J. Bernstein and Peter Schwabe, New AES software speed records, http://cr.yp.to/aes-speed/aesspeed-20080926.pdf

[4] Kris Gaj and Pawel Chodowiec, FPGA and ASIC Implementations of AES, http://teal.gmu.edu/courses/ECE746/project.

[5] Helion Technology, Overview Datasheet High Performance AES (Rijndael) cores for ASIC, http://www.heliontech.com/downloads.

[6] G.E. Moore, Cramming more components onto integrated circuits, *Electronics*, **38**(8), (Apr. 1965).

[7] Intel, *Moores Law: An Intel Perspective*, Intel Corporation, 2005.

[8] Ralph K. Cavin, Paolo Lugli and Victor V. Zhirnov, Science and Engineering Beyond Moores Law, *Proceedings of the IEEE*, **100**, (May, 2012).