

SCIENCE & MILITARY

No 1 | Volume 15 | 2020

Dear readers,

You are holding the first issue of the *Science & Military* in the fifteenth year of its existence. As the chief editor, I am pleased to say that our journal is attracting more and more readers and authors, which results in higher quality of articles submitted to the editorial board for publication.

Academic writing has become an increasingly important part of scientific work. It is impossible to carry out any research without sharing information and its results. The current trends in academic writing involve particularly new models that emerge in scientific communication and publication. They are influenced by digital tools, the Open Access Movement and the open science concept.

Academic writing significantly increases human knowledge. It is an important part of doctoral studies and it helps senior researchers gain prestige in academic circles and promote their research.

Dear readers, let me briefly introduce the latest issue of the *Science & Military* 1/2020, which contains interesting articles that underwent a rigorous review process.

The first among the peer-reviewed articles in this issue is the article written by Jozef Kostelanský and Lubomír Dedera titled **„Evaluation of Custom Virtual Machine Instruction Set Emulator“**. The main goal of the article is to evaluate performance characteristics of a custom virtual machine instruction set emulator. The authors compare performance characteristics of two implementations of the CRC16 algorithm – in the emulated custom virtual machine instruction set and the direct C-to-x86-compiled executable.

Another article titled **“Simulation Analysis of Planetary Transmissions in MATLAB Environment“** was written by Matúš Riečičiar and Peter Droppa. This paper deals with verification of the Simscape driveline module of the Matlab software as a tool of vehicle transmission and driveline designing. The authors compare different parameters results of analytical computing and Simscape simulations output values.

The authors Hung Nguyen Manh, Marie Richterová and Tomáš Břinčil wrote the article titled **„Performance Evaluation of Downlink Satellite Broadcast Stream Under Rainy Conditions in a Laboratory – Controlled Environment“**. This article analyses the basic technical properties and principles of the DVB-S2 system and the performance of the BER system under rainy conditions simulated by air in a room with an anechoic chamber. The results are analysed and used to design future research to determine the optimal coding rates and

modulation schemes as well as their impact on the entire signal transmission.

The author Anton Mydliar wrote the article titled **“Optimization of Ultrasonic Cutting Tool Geometry”**. This paper is concerned with the optimization of the ultrasonic cutting tool geometry. The author made the parameterizable model of Ultrasonic Cutting tool, which was created in finite element software Ansys. The eigen frequencies and modal shapes were extracted by modal analysis. The results and measurements are summarized in the conclusion of the article.

Another article titled **“Formal Model of Decomposition and Mapping in Accelerated Cluster Architecture”** was written by Miloš Očkay and Lubomír Dedera. It outlines basic elements of accelerated cluster architecture. It also explains decomposition and mapping in multistage architecture, using the data and task parallelism. The presented formal model describes decomposition in all stages, allowing more efficient mapping and achieving an accelerated solution to a complex problem.

The authors Martin Droppa and Marcel Harakal' wrote the article titled **“Cyber Security State in Real Environment”**. The purpose of this document is to give an insight into the wide area of world of cyber security and state of the cyber security in AFA (Armed Forces Academy in Liptovský Mikuláš) environment. In the end of the article, the findings from the state of cyber security at AFA are described. Based on a simple analysis of the situation in the AFA environment, the proposed recommendations, measures and methods of eliminating cyber attacks are briefly described.

The series of articles is closed with the paper titled **“Fundamentals of Static Malware Analysis: Principles, Methods and Tools”** written by Andrej Fedák and Jozef Štulrajter. One of the goals of this paper is to make readers familiar with the malware analysis, specifically with the complex subject such as static analysis. This includes clarifying what the analysis is and what it is used for. Another contribution of this paper is further description of these analytical tools and methods.

Dear readers, these are the articles, which have been selected for our first issue in 2020. We hope you will find them interesting and that they will motivate and inspire you to create new opinions, conduct research or react by writing new papers.

Col. (ret.) Prof. Eng. Marcel HARAKAL, PhD.
Chairman of the editorial board

Reviewers

Assoc. Prof. Dipl. Eng. Anton BALÁŽ , PhD.	The Technical University of Košice, SK
Lt. Col. Prof. Dipl. Eng. Jan FURCH , Ph.D.	University of Defence Brno, CZ
Lt. Col. Assoc. Prof. Dipl. Eng. Petr HRŮZA , Ph.D.	University of Defence Brno, CZ
Assoc. Prof. Dipl. Eng. Juraj JABLONICKÝ , PhD.	Slovak University of Agriculture in Nitra, SK
Prof. Dipl. Eng. Ján KOLLÁR , CSc.	The Technical University of Košice, SK
Prof. Dipl. Eng. Tomáš KRATOCHVÍL , PhD.	Brno University of Technology, CZ
Assoc. Prof. RNDr. Milan LEHOTSKÝ , CSc.	Catholic University in Ružomberok, SK
Assoc. Prof. Dipl. Eng. Ján OCHODNICKÝ , PhD.	Armed Forces Academy of General M. R. Štefánik, Liptovský Mikuláš, SK
Assoc. Prof. Dipl. Eng. Jaroslav PORUBĀN , PhD.	The Technical University of Košice, SK
Assoc. Prof. Dipl. Eng. Lýdia SOBOTOVÁ , PhD.	The Technical University of Košice, SK
Prof. Dipl. Eng. Jiří STODOLA , DrSc.	University of Defence in Brno, CZ
Assoc. Prof. Dipl. Eng. Martin TOMÁŠEK , PhD.	The Technical University of Košice, SK
Lt. Col. Assoc. Prof. Dipl. Eng. Michal TURČANÍK , PhD.	Armed Forces Academy of General M. R. Štefánik, Liptovský Mikuláš, SK
Mgr. Janka URAMOVÁ , PhD.	University of Žilina, SK

EVALUATION OF CUSTOM VIRTUAL MACHINE INSTRUCTION SET EMULATOR

Jozef KOSTELANSKÝ, Ľubomír DEDERA

Abstract: The main goal of the article is to evaluate performance characteristics of a custom virtual machine instruction set emulator. The instruction set has been designed as part of research aimed at utilization of custom virtual machines in the area of obfuscation techniques for software protection and malware detection, with the aim to efficiently implement the particular algorithm (CRC16). In the paper we compare performance characteristics of two implementations of the CRC16 algorithm – in the emulated custom virtual machine instruction set and the direct C-to-x86-compiled executable. The aim is to show that the emulation process of such a simple virtual machine has only minor influence on execution time in comparison with the C-to-x86-compiled code.

Keywords: Instruction set; Virtual machine; Performance; Compilers; Time measurements.

1 INTRODUCTION

Emulators allows to run code written for one architecture on another architecture. There are many emulators. For example, QEMU (1), Android Emulator (2) and many more. Emulation is also a promising method of software obfuscation (3). In (4) was custom reconfigurable instruction set proposed. The main goal of that custom virtual machine instruction set is to research static properties of reconfigurable instruction set emulators. It concludes with presenting findings that, using presented approach, it is possible to generate binaries with the same dynamic properties, but with different static properties. After reviewing static properties of the proposed emulator, the focus is switched to its dynamic properties. One of the requirements on software obfuscators from dynamic point of view is code efficiency or performance efficiency. It means that obfuscated code should not run much slower than the original code (5). Time measurements are as well used as methods for anomaly detections (6), (7), (8). The goal of this paper is to evaluate performance impact of the proposed custom virtual machine instruction set emulator running on top of the x86 architecture operating system with comparison towards direct x86 implementation. For this evaluation, CRC-16 algorithm implementation is used.

2 VIRTUAL MACHINE INSTRUCTION SET DESIGN

In (4) a custom virtual machine instruction set was proposed. Requirements were focused on simple and extensible instruction set creation. However, it was insufficient for the CRC16 (9) algorithm implementation. That is why the whole emulator has been rewritten specifically for CRC16 implementation with the following changes:

- **Registers** – The number of general-purpose registers has been increased from eight to sixteen registers.
- **Instructions** – The previously proposed instruction set lacked instructions that

are necessary for the CRC16 implementation.

Here we sum up the updated RISC-like architecture emulator attributes:

- **Architecture** – virtual machine template design is based on von Neumann architecture with shared memory between data and instructions.
- **Memory size** – virtual machine template design memory size is 32-bit. It consists of 16-bit addresses pointing to 16-bit values.
- **Instruction type** – virtual machine template design uses instructions of fixed length (16 bits per instruction).
- **Registers** – virtual machine template design uses 16 registers. Fifteen of them are general purpose registers. In addition, there is the instruction pointer register.
- **Flags** – virtual machine template design uses 3 flags. It updates them after every instruction execution. The flags are – FLAG_ZERO, FLAG_POSITIVE and FLAG_NEGATIVE. Their names describe their purpose.
- **Instructions** – custom virtual machine instruction set consists just of instructions necessary to implement the CRC16 algorithm. Here is a list of previously implemented instructions for reference:
 - LA is an instruction used for loading 8-bit value into registers, for example LA R0, 0x00000001 will load value 1 into register R0. This instruction is used for loading smaller constant values or addresses pointing to bigger values in memory.
 - LV is an instruction used for loading value pointed by address in one register into another register, for example LV R0, R1 will load value from address where is register R1 pointing into register R0.

- XOR is an instruction used for performing logical xor operation between 2 registers and writing output to another register; for example, XOR R0, R1, R2 will do the following operation: $R0 := R1 \text{ XOR } R2$. This instruction is used in addition to XOR also for setting the registry values to 0.
- ADD is an instruction used for adding 2 registers and writing output to another register; for example, ADD R0, R1, R2 will do following operation: $R0 := R1 + R2$.
- MOV is an instruction used for copying a value from one register to another; for example, MOV R0, R1 will copy value from register R1 into register R0.
- CMP is an instruction used for comparing 2 registers; for example, CMP R0, R1 will compare value in register R0 with value in register R1 and based on the result, it will update flags. If R0 equals to R1 then the instruction will set the FLAG_ZERO. If R0 is greater than R1, the instruction will set the FLAG_POSITIVE and if R0 is smaller than R1, the FLAG_NEGATIVE is set.
- JNZ is an instruction used for conditional branching; for example, JNZ R6 will do conditional jump to the address in register R6 if FLAG_ZERO is not set, and otherwise, it will continue with another instruction. Together with CMP instruction can be used for cycle implementation.
- PRINT is an instruction used for printing the value stored in a register; for example, PRINT R0 will print the value in register R0 on the standard output.
- HALT is an instruction used for halting the custom virtual machine down. If the interpreter encounters this instruction, it will shut down.

Newly added instructions:

- SUB is an instruction used for subtraction; for example, SUB R0, R1, R2 will subtract value in R2 from R1 and write result to R0.
- JNE is an instruction used for conditional branching; for example, JNE R0 will do conditional jump to address stored in register R0 if FLAG_ZERO is not set, and otherwise, it will continue with next instruction.

Together with CMP instruction, can be used for cycle implementation.

- JB is an instruction used for conditional branching; for example, JB R0 will do conditional jump to address stored in register R0 if FLAG_NEGATIVE is set, and otherwise, it will continue with next instruction. Together with CMP instruction, can be used for cycle implementation.
- AND is an instruction used for AND logical operation between two registers; for example, AND R0, R1, R2 will do following operation $R0 := R1 \text{ AND } R2$.
- OR is an instruction used for OR logical operation between two registers; for example, OR R0, R1, R2 will do following operation $R0 := R1 \text{ OR } R2$.
- LSHIFT is an instruction used for bitwise left shift operation; for example, LSHIFT R0, R1, R2 will do following operation $R0 := R1 \ll R2$.
- RSHIFT is an instruction used for bitwise right shift operation; for example, RSHIFT R0, R1, R2 will do following operation $R0 := R1 \gg R2$.

3 PROPOSAL OF THE MEASURE

The goal of this paper is to measure execution time needed to perform CRC16 checksum computations. A cyclic redundancy check (CRC) is an error-detecting code that is usually being used for accidental data changes detection. It can be also used for intentional data changes detection as well (9).

CRC is typically implemented using logical shifts for polynomial divisions (10). This kind of implementation was used for our evaluation of the proposed architecture because of its higher computation complexity than the implementation used originally in (3). There are also more effective approaches for CRC computation such as Computation of Cyclic Redundancy Checks via table look-up (11).

CRC16 is easy to implement and can be used for changes detection. The same CRC16 algorithm was implemented in pure C and in the proposed custom virtual machine instruction set.

3.1 CRC16 ALGORITHM DETAILS

There are many CRC specifications and implementations. Description of a specific CRC code needs a characterization by defining a division polynomial. An M-bit long CRC is based on a primitive polynomial of degree M, called a generator polynomial. For example, CCITT has chosen the following polynomial: $x^{16} + x^{12} + x^5 + 1$.

This polynomial can also be expressed like 1 0001 0000 0010 0001.

The CRC computation algorithm for an input word I and a given generator polynomial G of degree D is as follows:

First, multiply I by X^M . In binary, it results in adding M zero bits to I. Second, divide G into Ix^M . Since division is done on binary level, all of the subtractions are done as modulo 2. The Modulo 2 subtraction operation is the same as the logical exclusive or (XOR) operation. Third, ignore the quotient. Fourth, the remainder is part of CRC. Let's call it C. C will be a polynomial of degree M-1. If C is of higher degree, the division process has not been finished yet (12). All of these operations can be defined using logical shifts and exclusive OR logical operations.

3.2 CRC16 CUSTOM ARCHITECTURE IMPLEMENTATION

The CRC-16-ANSI specification has been chosen to be used for the purpose of this research. This implementation is specified by the 0x8005 ($x^{16} + x^{15} + x^2 + 1$) (1 1000 0000 0000 0101) polynomial. Below is presented the CRC16 implementation in the proposed custom virtual machine instruction set.

```
LA R0, 0x2
LA R1, 0x4
LA R3, 0x1
LV R2, R3
XOR R3,R3,R3
XOR R4,R4,R4
XOR R5,R5,R5
LA R9, 12
LA R10, 15
LA R11, 1
LA R12, 8
RSHIFT R5,R3,R10
LSHIFT R3,R3,R11
LV R15,R0
RSHIFT R6,R15,R4
AND R6,R6,R11
OR R3,R3,R6
ADD R4,R4,R11
LA R13, 25
CMP R4,R12
JB R13
LA R4, 0
ADD R0, R0, R11
SUBTRACT R1,R1,R11
LA R13, 29
CMP R5,R11
JB R13
XOR R3,R3,R2
CMP R14, R1
JB R9
LA R9, 34
LA R6, 0
```

```
LA R4, 16
RSHIFT R5, R3, R10
LSHIFT R3, R3, R11
LA R13,40
CMP R5,R11
JB R13
XOR R3,R3,R2
ADD R6,R6,R11
CMP R6, R4
JNE R9
LA R7,0
LV R6,R14
LA R8, 0x0001
LA R9, 47
AND R10, R6, R3
LA R13, 52
CMP R10,R11
JB R13
OR R7,R7,R8
RSHIFT R6,R6,R11
LSHIFT R8,R8,R11
CMP R6, R14
JNE R9
PRINT R7
HALT
```

3.3 COMPILATION DETAILS

Both programs were compiled on Linux, Ubuntu 16.04 with 4.4 kernel using gcc compiler v7.5.0 (13) for Linux operating system. Different optimization settings were used for measurements:

- Default configuration;
- -O1;
- -O2;
- -O3;
- -Ofast.

Both pure C implementation and custom virtual machine-based implementation used bitwise shift-based implementations and not precalculated table implementation.

Based on the online gcc compiler documentation (14), the influence of used optimization flags on compilation process is as follows. The -O1 flag turns on optimization during compilation process. Compiler tries to decrease the code size and more notably, it tries to decrease the execution time. It uses the basic set of optimizations but does not use any optimizations that would probably take a lot of compilation time. The -O2 flag instructs the compiler to optimize even more. Using this flag, the gcc compiler performs almost all supported optimizations with exception of those, which outcome in space-speed compromises. The usage of -O3 flag means to turn on further optimizations. It uses all optimizations specified by -O2 flag and some other ones. The largest set of optimizations is applied, when the Ofast flag is set. It uses all -O3 optimizations, but

also it enables optimizations that are not valid for all standard-compliant programs.

Totally, three sets of measurements were made. As an input, different subsets of ROCKOU.txt (15) wordlist have been used. The input subsets consist of 100, 1000, 10 000, 100 000 and 1 000 000 words. The measurements were done on Intel Core i7 with 8 GB of RAM memory.

For execution time measurements, *Time* (16) tool was used. Its results consist of the elapsed time in the User mode, Kernel mode and the real elapsed

time. For the purpose of this paper measurements, only real elapsed time was taken into consideration.

4 RESULTS

As it has been mentioned in the previous chapter, totally, three distinct sets of measurements were taken. In all from three distinct sets of measurements, 50 separate measurements were done. All chosen inputs were run against whole set of generated binaries using entire flags.

Tab. 1 Results of measurements of execution time

	c	c -O1	c -O2	c -O3	c -Ofast	vm	vm -O1	vm -O2	vm -O3	vm -Ofast
1000000	32:39.05	32:21.00	32:54.01	32:33.01	32:24.05	33:58.01	32:53.10	32:42.04	32:48.03	32:40.03
100000	03:15.01	03:18.05	03:17.06	03:14.03	03:14.03	03:21.04	03:17.09	03:16.07	03:16.03	03:15.07
10000	19.13	18.89	18.55	19.49	20.18	19.51	19.44	19.98	19.86	19.00
1000	1.81	1.52	1.95	2.00	1.82	2.04	2.00	1.82	2.01	2.00
100	0.17	0.1	0.09	0.09	0.18	0.19	0.09	0.14	0.17	0.1

Table 1 shows dependency of the execution time to the number of processed input words. The first column describes number of processed input words in distinct measure. Table displays results of measurements of execution time in seconds. The header consists of all samples which were used in tests. Mark c stands for clean C implementation and mark vm stands for custom virtual machine instruction set based implementation. Mark is then followed with used optimization flag used in compilation process.

Figures 1 to 5 shows dependency graphs of average execution time of samples in seconds for processing different number of input words.

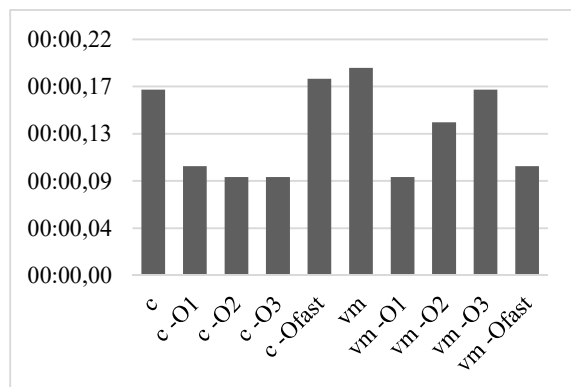


Fig. 1 Average execution time in seconds of samples for processing 100 input words
Source: authors.

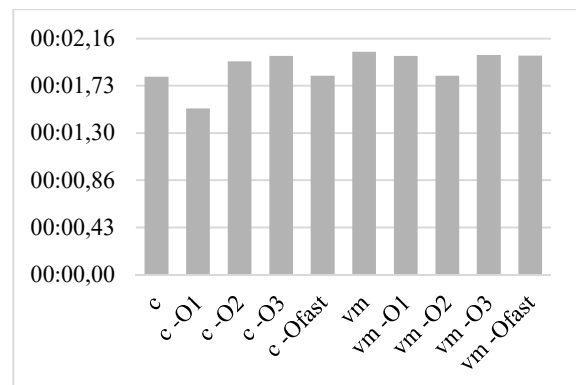


Fig. 2 Average execution time in seconds of samples for processing 1000 input words
Source: authors.

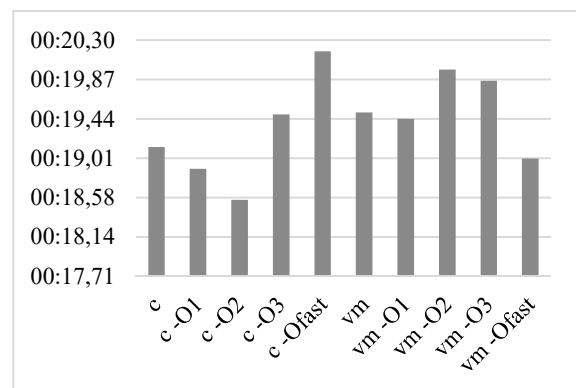


Fig. 3 Average execution time of samples for processing 10 000 input words
Source: authors.

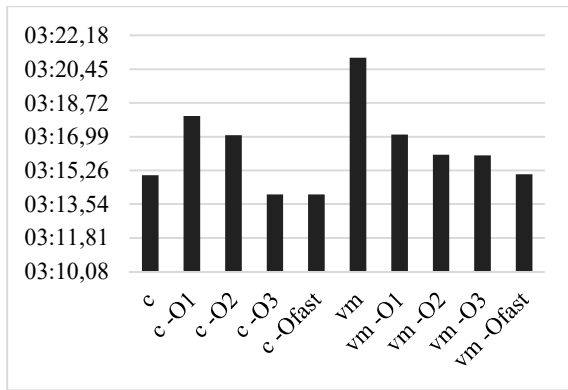


Fig. 4 Average execution time of samples for processing 100 000 input words
Source: authors.

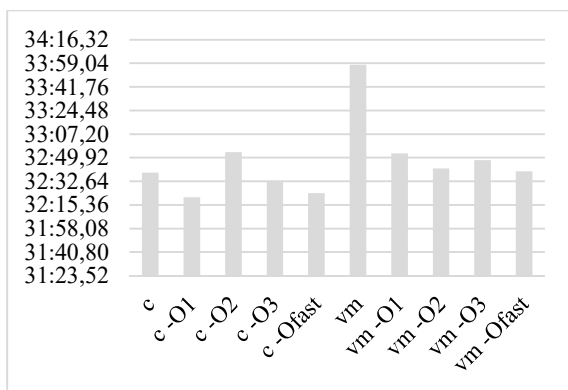


Fig. 5 Average execution time of samples for processing 1 000 000 input words
Source: authors.

It can be observed from the figures that, in general, there is only minor impact on performance between the pure C implementation and the custom emulator implementation for the selected algorithm and the input set. The maximal measured difference between samples without optimizations is only 13 % in favor for the pure C implementation. The maximal measured difference is 89 % in favor for pure C implementation. These results were measured between samples using -O3 optimizations flag on smallest input set. It is a lot, but it was measured on small input set, where operating system itself had huge influence on measures.

On the other hand, it is observable that using the smallest data input, sometimes the custom virtual machine instruction set based sample executes even quicker in some occasions.

Averagely, the difference was only 7 % in favor for the pure C implementation, but if we take into consideration only three biggest input datasets, difference was in average about 1 %.

We assume, that such small average difference of execution time between raw C implementation and custom RISC-based architecture emulator implementation is a result of effectively implemented

shellcode for emulator with many mentioned compiler optimizations done by hand.

5 CONCLUSION

The main goal was to evaluate execution time of custom virtual machine instruction set emulator compared to pure C implementation of CRC16 algorithm. The research has been motivated by the utilization of the custom virtual machines in the area of obfuscation techniques for software protection and malware detection.

Measured differences are really small. The average difference is only 7 % in favor to pure C implementation.

Algorithm implementations were in both cases based on bitwise shifts and not on precalculated tables, which is less effective, but leave space for more calculations and further optimizations.

Future measures may be focused on following research questions:

Validation of measures, so the follow-up research may be focused on different algorithms implementations, in order to minimize compiler impact on measures.

Measure the similarity score between different compilation flags usage, in order to minimize compiler impact.

To conclude, even though not all measurements results looks optimistic at initial view, but founded on assumptions that measured execution time in measures involving only small number of samples was highly influenced by operating itself, this custom virtual machine instruction set emulator overcame our assumptions, but it needs to be furthermore analyzed. Overall result of 7 % of execution time increase is inconsistent with (7) where was indicated that virtualization-based software will cause timing anomalies that will be detected "for free" by timing-based attestation. Since our proposed custom instruction set virtual machine meets the requirements mentioned in (5) for code efficiency, it does not have to be detected "for free" as stated above.

References

- [1] *QEMU a Fast and Portable Dynamic Translator*. Bellard, Fabrice. s. l.: USENIX Association, 2005. FREENIX Track: 2005 USENIX Annual Technical Conference. pp. 41-46.
- [2] *Run apps on the Android Emulator. Android Studio*. [Online] 12. 27, 2019. Available at: <<https://developer.android.com/studio/run/emulator>>.
- [3] YOU, I., KANGBIN, Y.: Malware Obfuscation Techniques: A Brief Survey. In: *Proceedings - 2010 International Conference on Broadband*,

- Wireless Computing Communication and Applications*, BWCCA 2010.
- [4] KOSTELANSKÝ, J., DEDERA, L.: Custom virtual machine implementation and its influence on executable static properties. In: *2019 Communication and Information Technologies (KIT)*. [online] Liptovský Mikuláš : Akadémia ozbrojených síl generála M. R. Štefánika, 2019. ISBN 978-80-8040-575-5.
 - [5] FANG, H., WU, Y., WANG, S., HUANG, Y.: Multi-stage Binary Code Obfuscation Using Improved Virtual Machine. In: Lai X., Zhou J., Li H. (eds) *Information Security*. ISC 2011. Lecture Notes in Computer Science, vol. 7001. Springer, Berlin, Heidelberg, 2011.
 - [6] LU, S., LYSECKY, R. L., ROZENBLIT, J. W.: Subcomponent timing-based detection of malware in embedded systems. In: *Proceedings - 35th IEEE International Conference on Computer Design, ICCD 2017*. pp. 17-24. [8119185] Institute of Electrical and Electronics Engineers Inc.. <https://doi.org/10.1109/ICCD.2017.12>
 - [7] KOVAH, X., KALLENBERG, C., WEATHERS, Ch.: New Results for Timing-Based Attestation. In: *IEEE Symposium on Security and Privacy*. 2012. pp. 239-253.
 - [8] LU, S., SEO, M., LYSECKY, R.: Timing-based anomaly detection in embedded systems. In: *20th Asia and South Pacific Design Automation Conference, ASP-DAC 2015*. pp. 809-814. [7059110] Institute of Electrical and Electronics Engineers Inc. <https://doi.org/10.1109/ASPDAC.2015.7059110>
 - [9] PETERSON, W. W., BROWN, D. T.: Cyclic Codes for Error Detection. Brown. In: *Proceedings of the IRE*, Vol. 49, Issue 1, 1961. pp. 228-235.
 - [10] *A Commonsense Approach to the Theory of Error-Correcting Codes*. s. l. : The MIT Press; n edition, 1988. 0262010984.
 - [11] SARWATE, D. V.: *Computation of Cyclic Redundancy Checks via Table Look-Up*. New York : Association for Computing Machinery, 1988. Commun. ACM, Vol. 31, p. 6. 0001-0782.
 - [12] Press, William, H., et al. *Numerical Recipes in C*. New York : Cambridge University Press, 2002. ISBN 0-521-43108-5.
 - [13] GRIFFITH, A.: *GCC: The Complete Reference*. New York : McGraw-Hill, Inc., 2002. ISBN 978-0-07-222405-4.
 - [14] GCC online documentation. *GCC*. [Online] Free Software Foundation, Inc., 11 28, 2019. Available at: <https://gcc.gnu.org/onlinedocs/>.
 - [15] Passwords. *Skullsecurity*. [Online] 5 18, 2015. Available at: <http://downloads.skullsecurity.org/passwords/rockyou.txt.bz2>.
 - [16] Time(1) - Linux man page. [Online] Available at: <https://linux.die.net/man/1/time>.

Dipl. Eng. Jozef **KOSTELANSKÝ**
 Armed Forces Academy of General M. R. Štefánik
 Department of Informatics
 Demänová 393
 031 01 Liptovský Mikuláš
 Slovak Republic
 E-mail: jozef.kostelansky@gmail.com

Assoc. Prof. RNDr. Ľubomír **DEDERA**, PhD.
 Armed Forces Academy of General M. R. Štefánik
 Department of Informatics
 Demänová 393
 031 01 Liptovský Mikuláš
 Slovak Republic
 E-mail: lubomir.dedera@aos.sk

Jozef Kostelansky was born in Slovakia. He received his engineer degree in 2016 in Communication and Information systems from the Military Academy in Liptovský Mikuláš with his thesis focused on Android malware analysis. He is currently the PhD. student researching hybrid malware analysis techniques.

Ľubomír Dedera works as an Associate Professor at the Department of Informatics, Armed Forces Academy in Liptovský Mikuláš. He graduated (RNDr.) from the Faculty of Mathematics and Physics, Comenius University in Bratislava in 1990. He received a PhD. degree in Artificial Intelligence from the Military Academy in Liptovský Mikuláš in 1997. His research interests include computer languages, computer security and artificial intelligence.

SIMULATION ANALYSIS OF PLANETARY TRANSMISSIONS IN MATLAB ENVIRONMENT

Matúš RIEČIČIAR, Peter DROPPA

Abstract: At present time three main widespread methods of planetary gearboxes analysis are used – analytical calculation, practical tests and simulations. This paper deals with verification of the Simscape driveline module of the Matlab software as a tool of vehicle transmission and driveline designing. Different parameters results of analytical computing and Simscape simulations output values are compared in this paper. Deviations of these values turned out to be insignificant and therefore Simscape module proved to be applicable, usable, cost and time reducing and very useful in transmissions designing process.

Keywords: Planetary gearbox; Transmission; Simulation; Simscape; Driveline.

1 INTRODUCTION

Planetary gearbox is a mechanism that has found wide application in a design of the both wheeled and tracked military vehicles transmission systems. A large scale of kinematic and dynamic effects appears in different designs of these mechanisms. These effects can seriously affect the dynamic and tractive effort characteristics of the vehicle and consequently can affect the applicability of the certain planetary gearbox in the vehicle powertrain. Therefore the determination or the computation of these effects and parameters values of each design at each mode is a necessity. Currently three main widespread computing approaches are used. Due to longer computing time and simplifications of the analytical approach and higher costs of practical gearboxes testing, different simulation tools are nowadays used more and more. Despite some cons like its simplifications, it is very effective and economically advantageous to prove applicability of certain gearbox design in vehicle powertrain by using simulation tools before practical test execution.

Purpose of this paper is to verify the Simscape Driveline module of the Matlab software as one of the possible ways of conducting planetary gearbox analysis.

Because of the impossibility of practical tests execution in our environment we have chosen the method of comparing results gained by analytical method and by simulation for verification.

We have chosen the systematical approach – from basic models and basic parameters to more complex ones. The goal is not yet to create the perfectly reality matching model, but just to verify if the Simscape module is usable for drivelines and gearboxes modeling and analyzing.

Our assumption is that the deviations of the values gained analytically and by successful models simulations should not exceed the value of 5 % deviation.

2 ANALYZED GEARBOX

Gearbox ZF 6HP26 is a part of the drivetrain of the armored vehicle Iveco LMV 4x4. It has three

degrees of freedom, five control elements and use 7 (6+1) of 10 theoretically possible gears. [1]

Gearbox is equipped with torque converter with lock-up clutch and is composed of one simple planetary gear train (PGT) and one Ravigneaux gear set (RPGT). It has five control elements – two multiple disk brakes (B1, B2) and three multiple plate clutches (C1, C2, C3). For further analysis we need to know design of the gearbox and connections between components. This information is represented by kinematic scheme of the gearbox (Fig. 1). [1]

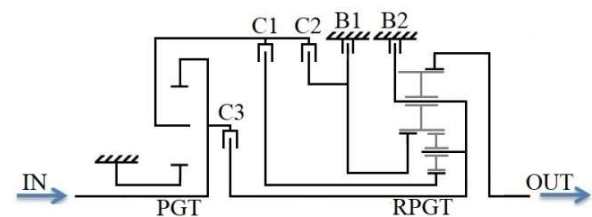


Fig. 1 Kinematic scheme of gearbox ZF 6HP26
Source: authors.

3 ANALYTICAL APPROACH

The analysis of any planetary gearbox consists of six basic steps: [2] [3]

1. Planetary gear train (PGT) function analysis.
2. Kinematic parameters analysis.
3. Dynamic parameters analysis.
4. Efficiencies computing and detection of parasitic power.
5. Basic principles of PGT construction.
6. Strength calculation [4].

For first step verification of usability of the simplified powertrain model have been chosen both basic kinematic parameter (revolutions) and basic dynamic parameter (torques) of PGT main components. (steps 2. and 3. of the analysis)

If simplified powertrain model proves to be usable for further computing we will try to create more complex powertrain model. Than we would be able to verify and compare also the efficiencies gained analytically and by a simulation. (step 4. of the analysis).

3.1 Revolutions analysis

By accomplishment of the first point of the analysis we were able to create computing scheme and PGT angular velocities graph (Fig. 2).

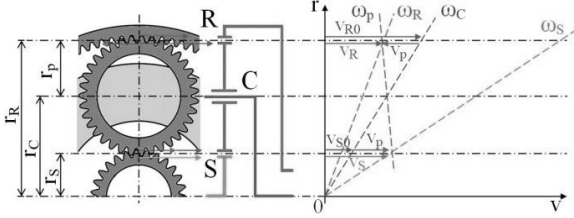


Fig. 2 PGT angular velocities graph
Source: authors.

Further investigation of this graph, computing scheme and PGT function has led to the derivation of the basic PGT kinematic equation (1). [2] [3]

$$n_S + n_R \alpha - n_C(1 + \alpha) = 0 \quad (1)$$

Where: n_S - sun gear revolutions
 n_R - ring gear revolutions
 n_C - carrier revolutions
 α - PGT parameter

Understanding and proper usage of this equation is fundamental for further computing of the revolutions (rpm) of the main PGT components.

3.2 Torques analysis

PGT dynamic parameters calculation is based on the basic mechanics principles and principle of the balance of forces and torques acting on the carrier. (Fig. 3) [2]

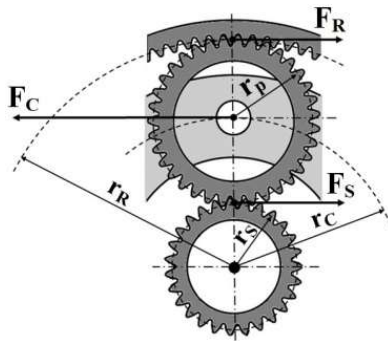


Fig. 3 Balance of forces in basic PGT
Source: authors.

By further investigation of the balance of forces and torques inside PGT and by application of the basic torque formula (2) we can derive following PGT torque formulas. (Tab. 1) [2] [3]

$$\tau = F \cdot r \quad (2)$$

Tab. 1 Basic PGT torque formulas [2]

$\tau_S = \frac{\tau_R}{\alpha}$	$\tau_R = \tau_S \cdot \alpha$	$\tau_C = \tau_S(1 + \alpha)$
$\tau_S = \tau_C \frac{1}{1 + \alpha}$	$\tau_C = M_0 \frac{\alpha}{1 + \alpha}$	$\tau_C = M' \frac{1 + \alpha}{\alpha}$

Where: τ_S - sun gear torque
 τ_R - ring gear torque
 τ_C - carrier torque
 α - PGT parameter

3.3 Efficiencies analysis

Considerable losses caused mainly by friction of gears, bearings friction and oil resistance can occur in every gearbox design. Simplified calculation of the efficiencies often takes only the most significant gear friction losses into account. [2] [3]

In the PGT the power is transmitted not only by relative motion (with gear losses), but also by carrier motion (without gear losses). Therefore it is not possible to use the simple efficiency equation of fixed shaft axle transmission. We have to take to account what part of the power is transmitted by relative motion and what by carrier motion. We start with basic gear efficiency equation (3). [2] [3]

$$\eta_m = \frac{P_2}{P_1} = \frac{i_p}{i_k} \quad (3)$$

Where: P_1 - input shaft power
 P_2 - output shaft power
 i_p - power ratio
 i_k - kinematic ratio

While kinematic ratio is defined as function of PGT parameters in action at particular gear (4) [2]

$$i_k = \frac{n_2}{n_1} = f(\alpha_1, \alpha_2, \dots, \alpha_n) \quad (4)$$

Power ratio is defined as function of PGT parameters and efficiencies of these PGTs (5). [2]

$$i_p = \frac{\tau_2}{\tau_1} = f(\alpha_1 \eta^{x_1}, \alpha_2 \eta^{x_2}, \dots, \alpha_n \eta^{x_n}) \quad (5)$$

Where: η - PGT efficiency (0,96 – 0,98)
 x - exponent with value ± 1

For finding +/- sign of the exponent x equation (6) has been derived. [2]

$$sngx_i = sng \frac{\alpha_i}{i_i} \cdot \frac{\partial i_{ck}}{\partial \alpha_i} \quad (6)$$

Where: i_i - kinematic gear ratio of particular gear

Using equation (3) we can get power efficiencies of the gearbox at each gear analytically.

4 NUMERICAL APPROACH

The selected tool for numerical gearbox analysis is Simscape driveline – subcomponent of the Matlab software, which is designed for iterative analysis and design processes with a programming language that express matrix and array mathematics directly. [5]

Simscape driveline is a tool used especially for modeling and simulating translational and rotational mechanical systems. Its most significant advantage is its components library. This library includes basic models of drivetrain components. Therefore there is no need to design these components, but just to determine their parameters and links between them to create a model. Driveline modeling employs a physical network approach, where Simscape blocks correspond to physical components, such as engines, gears, brakes, clutches, tires, pistons and so on. The lines that connect these components represent the physical connections that transmit power. The resulting models let us describe the physical structure of a system, rather than the underlying mathematics. [4]

In general, there are more benefits that make software simulations very effective tool for gearbox analysis:

1. Low cost.
2. Calculation time – lower due to computers improving.
3. Cover multiple parameters by one calculation.
4. Possibility of parameters values monitoring at each mode by simple model editing.
5. Simple change of input parameters.
6. Simple change of model components parameters.

On the other hand, in the simulation process, there are various simplifications which make the simulation results less accurate and less reliable. Therefore it is necessary to verify these results either by practical test execution or/and by analytical calculation.

4.1 Basic Simulation model

To verify applicability of the Simscape driveline as a tool of the gearboxes designing the systematical approach has been chosen. We have started with basic

model of the simplified drivetrain with ideal angular velocity source or ideal torque source (Fig. 3) and basic ZF 6HP26 gearbox model.

This way we were able to exclude possible mistakes and errors which could have occurred in other blocks (engine, torque converter, tires etc.).

If we could prove that the results gained by numerical and analytical approach are identical or with insignificant deviation, we would proceed to more complex models.

The simplified model (Fig. 4) consists of ideal source which is the input for the simulation and of the gearbox ZF 6HP26 itself. Outputs of the simulation are the rpms and torques of each main PGTs components measured by ideal rotational motion sensors and ideal torque sensors.

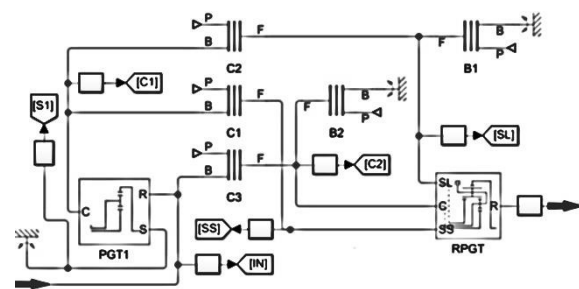


Fig. 4 Basic ZF 6HP26 gearbox model
Source: authors.

5 RESULTS COMPARISON

Table 2 represents values of both rpms and torques of main components of the simple planetary gear train and Ravigneaux planetary gear train at each one of seven used gears.

Each value in the table is a relative value to the input shaft revolutions or torque. There are the values of the main components rpms and torques calculated both analytically and by simulation.

Finally there is a deviation of these two values at each gear for all gearbox main components. As we can see, the values of the deviation are zero or insignificant. Even the largest values of the deviation occur only at the third decimal place and any of them exceeds 0,03 % of the particular value. This fact has led us to assumption that the basic model of the gearbox is correct and usable for further more complex modeling and simulations.

Tab. 2 Comparison of analytical and simulation rpms and torques results

Gear	Calculation	Revolutions of the main Components							Torques of the main Components							i
		S1	C1	R1	SS	SL	C2	R2	S1	C1	R1	SS	SL	C2	R2	
1st	Analytical	0,0000	0,6575	1,0000	0,6575	-0,5361	0,0000	0,2397	-0,5208	1,5210	1,0000	1,5210	0,0000	2,6460	4,1670	4,1670
	Simulation	0,0000	0,6575	1,0000	0,6575	-0,5375	0,0000	0,2400	-0,5210	1,5210	1,0000	1,5210	0,0000	2,6508	4,1717	4,1717
	Deviation	0,0000	0,0000	0,0000	0,0000	0,0014	0,0000	0,0003	0,0002	0,0000	0,0000	0,0000	0,0000	0,0048	0,0047	0,0047
2nd	Analytical	0,0000	0,6575	1,0000	0,6575	0,0000	0,2953	0,4273	-0,5208	1,5210	1,0000	1,5210	0,8167	0,0000	2,3380	2,3380
	Simulation	0,0000	0,6575	1,0000	0,6575	0,0000	0,2958	0,4275	-0,5210	1,5210	1,0000	1,5210	0,8191	0,0000	2,3400	2,3400
	Deviation	0,0000	0,0000	0,0000	0,0000	0,0005	0,0002	0,0002	0,0002	0,0000	0,0000	0,0000	0,0024	0,0000	0,0020	0,0020
3rd	Analytical	0,0000	0,6575	1,0000	0,6575	0,6575	0,6575	0,6575	-0,5208	1,5210	1,0000	0,9895	0,5314	0,0000	1,5210	1,5210
	Simulation	0,0000	0,6575	1,0000	0,6575	0,6575	0,6575	0,6575	-0,5210	1,5210	1,0000	0,9886	0,5324	0,0000	1,5210	1,5210
	Deviation	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0002	0,0000	0,0000	0,0009	0,0010	0,0000	0,0000	0,0000
4th	Analytical	0,0000	0,6575	1,0000	0,6575	1,2793	1,0000	0,8751	-0,1428	0,4171	0,2743	0,4171	0,0000	0,7257	1,1430	1,1430
	Simulation	0,0000	0,6575	1,0000	0,6575	1,2800	1,0000	0,8750	-0,1427	0,4166	0,2739	0,4166	0,0000	0,7261	1,1427	1,1427
	Deviation	0,0000	0,0000	0,0000	0,0000	0,0007	0,0000	0,0001	0,0001	0,0005	0,0004	0,0005	0,0000	0,0004	0,0003	0,0003
5th	Analytical	0,0000	0,6575	1,0000	1,4201	0,6575	1,0000	1,1532	0,1326	-0,3872	-0,2546	0,0000	-0,3872	1,2550	0,8674	0,8677
	Simulation	0,0000	0,6575	1,0000	1,4190	0,6575	1,0000	1,1530	0,1328	-0,3878	-0,2550	0,0000	-0,3878	1,2550	0,8672	0,8672
	Deviation	0,0000	0,0000	0,0000	0,0011	0,0000	0,0000	0,0002	0,0002	0,0006	0,0004	0,0000	0,0006	0,0000	0,0002	0,0005
6th	Analytical	0,0000	0,6575	1,0000	2,2265	0,0000	1,0000	1,4472	0,0000	0,0000	0,0000	0,0000	-0,3086	1,0000	0,6914	0,6914
	Simulation	0,0000	0,6575	1,0000	2,2230	0,0000	1,0000	1,4460	0,0000	0,0000	0,0000	0,0000	-0,3090	1,0000	0,6910	0,6910
	Deviation	0,0000	0,0000	0,0000	0,0035	0,0000	0,0000	0,0012	0,0000	0,0000	0,0000	0,0000	0,0004	0,0000	0,0004	0,0004
R	Analytical	0,0000	0,6575	1,0000	-0,8064	0,6575	0,0000	-0,2940	-0,5208	1,5210	1,0000	0,0000	1,5210	-4,9280	-3,4070	-3,4070
	Simulation	0,0000	0,6575	1,0000	-0,8043	0,6575	0,0000	-0,2935	-0,5210	1,5210	1,0000	0,0000	1,5210	-4,9223	-3,4013	-3,4013
	Deviation	0,0000	0,0000	0,0000	0,0021	0,0000	0,0000	0,0005	0,0002	0,0000	0,0000	0,0000	0,0000	0,0057	0,0057	0,0057

6 COMPLEX DRIVETRAIN MODEL

The comparison of the values of rpms and torques of the main ZF 6HP26 gearbox components gained both analytically and by simulation proved that this gearbox model can be integrated into the more complex simulation models.

In these simulations, instead of being the main part, the gearbox is just a subcomponent of the whole vehicle drivetrain model.

Modeled vehicle is the Iveco LMV 4x4 which is used in Slovak military forces and which actually uses ZF 6HP26 gearbox. We have tried to model all of the vehicle drivetrain subcomponents as realistic as we have been able to. The level of matching the reality is mostly based on the availability of the components parameters.

The purpose of this particular model is not to completely match the reality yet, but just to show the way we can observe the function of the vehicle drivetrain in much easier and much more intuitive way.

The more complex model of the Iveco LMV 4x4 drivetrain model consists of seven main subcomponents: (Fig. 5)

1. **Driver input** – which represents position of the accelerator pedal and is also an input for the engine.
2. **Engine** – engine model characterized by basic parameters.
3. **Torque converter** – generic model.
4. **Gearbox** – analyzed ZF 6HP26 model.
5. **TCU** – transmission control unit shifting gears and working with shift map based on vehicle speed and accelerator pedal position.
6. **Additional gearbox** – reduction and transfer gearbox.
7. **Vehicle body** – consists of chassis components (differentials, final gears, wheels etc.) and vehicle body represented by vehicle dimensions.

In these simulations, instead of being the main part, the gearbox is just a subcomponent of the whole vehicle drivetrain model.

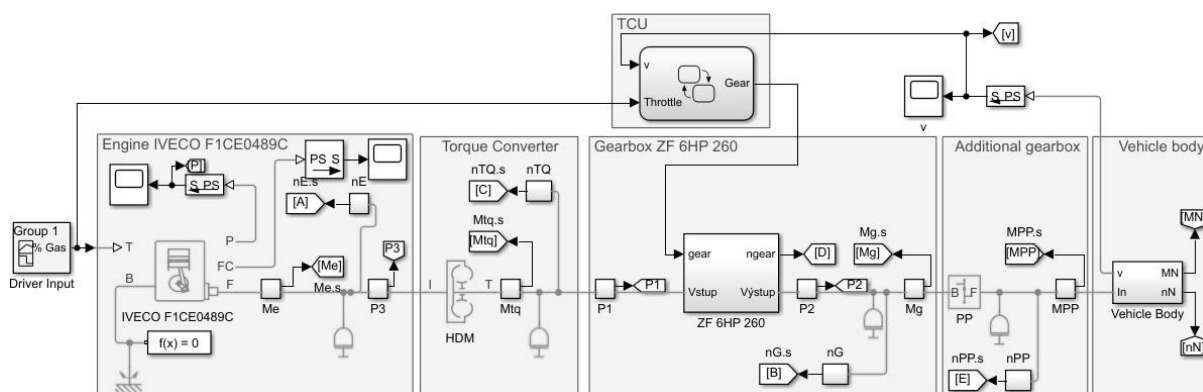


Fig. 5 Complex IVECO LMV 4x4 drivetrain model
Source: authors.

6.1 Complex drivetrain verification

For first step verification of the model applicability the fourth step of gearbox analysis has been chosen – comparison of the values of efficiencies gained both analytically and by the complex model simulation (part 3).

The complex model simulations are able to provide great number of different outputs and results. We are not able to reliably compute these

values analytically, especially with large number of unknown parameters influencing the results.

Therefore we have chosen the efficiencies values as the first step of the model verification. The analytical computing has been conducted using method stated in part 3.3. the gearbox in complex model (Fig. 5) is equipped with power sensors P1 at input shaft and P2 at output shaft. By comparison of these two power values at each gear we have got the following graph. (Fig. 6)

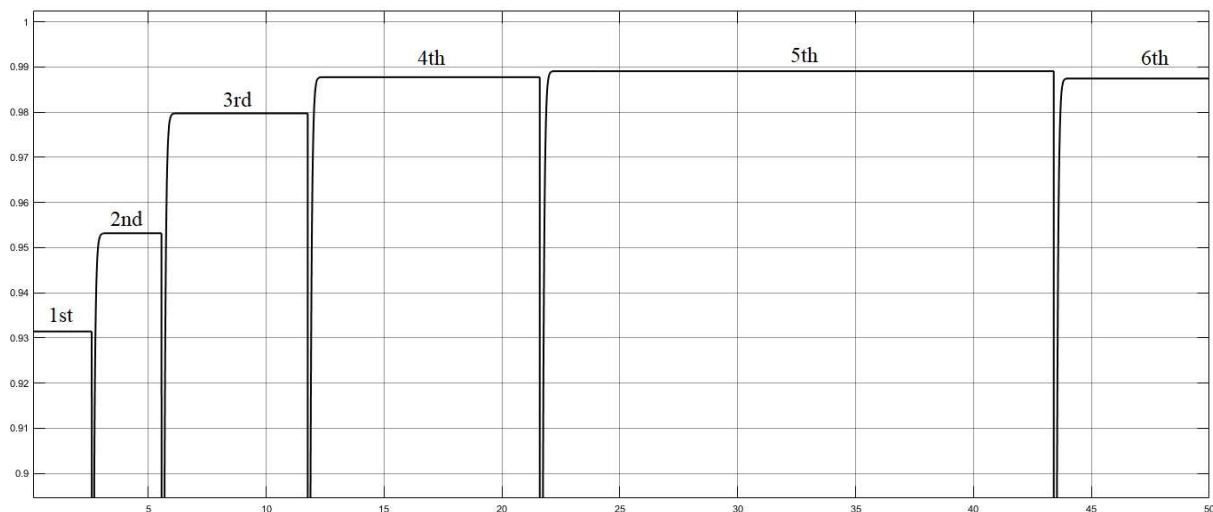


Fig. 6 Efficiencies of the ZF 6HP26 gearbox at each gear gained by simulation

Source: authors.

The following table (Tab. 3) represents the comparison of the efficiencies values gained both analytically and by the complex model simulation.

As we can observe from the table, the results are very similar with insignificant deviation which does not exceed 1 % of the efficiencies values.

Tab. 3 Comparison of analytical and simulation efficiencies results

Gear	1st	2nd	3rd	4th	5th	6th
Analytical	0,9245	0,9606	0,9887	0,9880	0,9925	0,9896
Simulation	0,9314	0,9532	0,9797	0,9877	0,9891	0,9874
Deviation	0,0069	0,0074	0,0090	0,0003	0,0034	0,0022
%	0,7442	0,7698	0,9100	0,0259	0,3404	0,2188

7 CONCLUSION

This paper deals with the verifying of the usability and applicability of the Simscape driveline software as a tool of the modeling and designing of the transmission mechanisms. To verify applicability of the software, the systematical approach has been chosen – from the simplified models to the more complex models and simulations. For this purpose the results of both analytical calculations and software simulations of the planetary gearbox ZF 6HP26 have been compared. We have created both the simplified and complex

model of the vehicle drivetrain and have conducted a simulation. We have also made an analytical calculation of three major parameters at each gear.

In first step we have compared values of revolutions and torques of each main gearbox component at each gear with values gained by simple model simulation. The largest deviation has been 0.03 % of the original value. We have found this deviation acceptable for further modeling of the more complex model.

In next step we have conducted comparison of efficiencies values at each gear gained by analytical calculation and by the complex model simulation. The largest deviation has been 0.91 % of original value.

Our assumption had been that the deviation between successful model values and analytical results should not exceed 5 % value.

Therefore we can confirm our hypothesis and we can claim that the Simscape driveline model is one of the possible ways of conducting transmissions and drivelines analysis. This tool proved to be applicable, usable, cost and time reducing and very useful in transmissions designing process.

This paper also indicates the possible path of further effort – to investigate another ways and parameters which could be observed, verified and analyzed in simulation and either analytical or real

conditions, what could lead to more reliable verification of the Simscape driveline module.

automatic control and numerical analysis of planetary gearboxes.

References

- [1] ZF6HP26 *Technicians Diagnostic Guide*. Online. Available: <<http://maybeme.com/Storage/JagTech/ManualsHandbooksTSBs/6hp26%20ZF%20Files/24267127-Audi-Jaguar-Bmw-Zf6hp26%E2%80%9D.pdf>>
- [2] ŽALUD, Z.: *Bojová a Speciální vozidla II: Převodová ústrojí bojových pásových vozidel – Planetové převody*. Brno : Univerzita obrany v Brně, 2010. s. 3-78. ISBN 978-80-7231-706-6.
- [3] HÁJEK, M., KABOUREK, J.: *Bojová pásová vozidla: Konstrukce a výpočet*. Brno : VA AZ, 1985. s. 136-473.
- [4] DIŽO, J., BLATNICKÝ, M., HARUŠINEC, J., FALENDYSH, A.: Modification and analyses of structural properties of a goods wagon bogie frame. In: *Diagnostyka*, 2019, 10(1): 41-48. Available at: <<https://doi.org/10.29354/diag/99853>>
- [5] *Model and simulate rotational and translational mechanical systems*. Online. Available at: <<https://www.mathworks.com/products/simdrive.html>>

Dipl. Eng. Matúš **RIEČIČIAR**
Armed Forces Academy of General M. R. Štefánik
Department of Mechanical Engineering
Demänová 393
031 01 Liptovský Mikuláš
Slovak Republic
E-mail: matus.rieciciar@gmail.com

Prof. Dipl. Eng. Peter **DROPPA**, PhD.
Armed Forces Academy of General M. R. Štefánik
Department of Mechanical Engineering
Demänová 393
031 01 Liptovský Mikuláš
Slovak Republic
E-mail: peter.droppa@aos.sk

Matúš Riečičiar was born in 1993 in Slovakia. He received his engineer degree in 2017 in Armament and technics of military forces from Department of Mechanical Engineering of the Military Academy in Liptovský Mikuláš with his thesis focused on Planetary gearboxes analysis. He is currently the PhD. student researching

Peter Droppa works as an Professor at the Department of Mechanical Engineering, Armed Forces Academy in Liptovský Mikuláš. In 1984 he graduated (Eng.) at the Military University in Brno in branch Mechanical technology. He received a PhD. degree in Armament and Technics of Land Forces from the Military Academy in Liptovský Mikuláš in 2003. Since 2007, he has been become the associate professor at the Transport Machines and Equipment, Faculty of Military Technology, University of Defence. And in 2013 he was appointed a professor. His research interests include combat and special vehicles, experimental measurement and simulations and their development trends.

PERFORMANCE EVALUATION OF DOWNLINK SATELLITE BROADCAST STREAM UNDER RAINY CONDITIONS IN A LABORATORY-CONTROLLED ENVIRONMENT

Hung NGUYEN MANH, Marie RICHTEROVÁ, Tomáš BŘINČIL

Abstract: The attenuation induced by rain greatly impacts the performance in the satellite communication at Ku band. In fact, it is very challenging to measure parameters of the satellite channel under rainy conditions in the laboratory environment. Due to the complex in terms of design, development, management and maintenance of the system and the dynamic characteristics of the satellite channel with respect to operators utilizing this frequency band, it becomes essential to study the key effects and causes of the signal attenuation. In order to have the accurate evaluation in the satellite channel, end-to-end performance of the system based on a description of the standard document under raining conditions was modelled by the Rician distribution and K-factor. The average bit error rate used for the performance evaluation of the system by the experimental test. Obtained results in the satellite channel condition are evaluated, analyzed and discussed. Moreover, specific future planned work is also mentioned in the article.

Keywords: Digital Broadcast Video; Sum-of-sinusoids principle; Rician distribution; Rain attenuation; Channel impairments.

1 INTRODUCTION

Digital Video Broadcasting - Satellite - Second Generation (DVB-S2) [1] standards address broadband interactive satellite systems for mainly providing high data rate multimedia services to fixed terminals. The candidate frequency bands of the DVB-S2 system based broadband satellite systems to fully support are the Ku-band in the Europe. Simulation of a suitable general model in a laboratory-controlled environment is very important to address an issue of the system. Space communication and navigation test-bed on the International Space Station provided an opportunity to evaluate the performance of the DVB-S2 system, as well as demonstrated advanced features of the system could be integrated and operated. Reliability of the DVB-S2 system such as variable and adaptive coding and modulation was evaluated. The test-bed [2] was based on the commercial gateway product, DVB-S2 modulator, demodulator and a channel simulator for emulating communication channels, that demonstrated that a DVB-S2 system can be operated close to saturation in quadrature phase shift keying (QPSK), 8-ary phase-shift keying (8-PSK) and 16-ary Amplitude Phase Shift Keying (16-APSK).

Software-Defined Radio (SDR) provides flexible radio functionality to develop a flexible wireless laboratory platform. In [3], performance measurements of the symbol timing recovery of DVB-S2 receiver circuits over the Additive White Gaussian Noise (AWGN) channel was presented and measured for the clear sky period. Real DVB-S2 signal [4] with different parameters was analyzed for reception and measurement. In the practice, atmospheric effect is a one of the key factors in the design of satellite-to-earth links operating at frequencies above 10 GHz. In the atmosphere, raindrops absorb cause the signal attenuation and reduction of the system availability and reliability. Hence, the rainy factor in the satellite connection

becomes more important in the successful connection. The Nakagami-m distribution is often modeled the satellite channel because of its advantages of a better fit for amplitude statistics. However, the Nakagami-m distribution alone, unlike the Rician model, has the disadvantage of being an amplitude only description, but even a slight phase error may lead to the problem of the simulation. X. Hao [5] and M. Cheffena [6] proved that the correlation exists between the atmospheric and the factor of Rician distribution. The practical scenario with a lot of parameters such as particularly the effects of different elevation angles, frequencies, geographical region, and weather parameters have not yet been described in the proposed correlation. Therefore, a modified reliable correlated equation that includes the effects of the mentioned parameters is led by Liolis [7] for filling the gap. The relationship between rain affects and the value of the K factor was showed in the following equation:

$$K_{rician} = K_{mod} - K_{rain} \quad (1)$$

where K_{mod} is the factor before adding the atmospheric effects. K_{rain} is the factor based on the rainy statistical analysis was obtained by Crane's model [8] for temperate regions. Effective path average factors were estimated by observations of the point rain rate statistics, of the horizontal structure of rainfall, and of the vertical temperature structure of the atmosphere. If the signal is totally blocked, then the model signal in the satellite channel was usually represented by a Rayleigh distribution [9]. Flat Rayleigh model can represent the worst-case environment in terms of bit error rate degradation. To test the performance of the system, a test-bed consisting of SDRs and laboratory measurement instrument allows the research of rainy scenarios that arise in the satellite link. This experimental platform also provides an opportunity to demonstrate how the DVB-S2 system could be operated in the laboratory environment.

2 SIMULATION METHODOLOGY

2.1 System model

In the following part, the detail of the system model that we adopted for the end-to-end DVB-S2 system analysis. Based on the specification of European Telecommunications Standards Institute (ETSI), a simulation of the DVB-S2 system was simulated. Compared to the full described ETSI standard, we demonstrate in our simulator all key aspects impacting the physical layer performance. The time division multiple access technique with fade mitigation based on forward error correction codes is the access technique in the DVB-S2 system. Data frames of constant size are packaged and the coded-modulation used to adapt to the propagation conditions on each communication link. The system model that is utilized for the DVB-S2 system simulation. Fig. 1 shows the high-level system block diagram based on the specification [1].

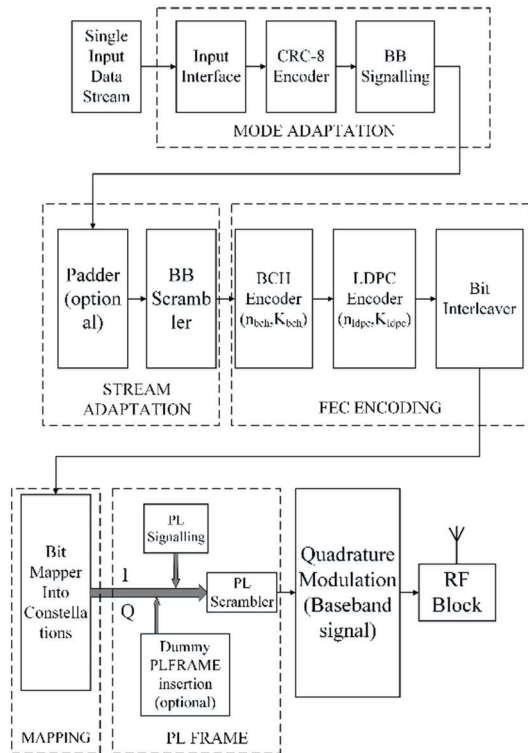


Fig. 1 Block diagrams of the DVB-S2 System
Source: [1].

Input data of the system is a single Moving Picture Experts Group (MPEG)-4 video stream. The output of the Forward Error Correction (FEC) encoding block is frames that have a fixed size of 64800 bits for long FECFRAME or 16200 bits for short FECFRAME. Modulation schemes with various code rates are QPSK, 8-PSK, 16-APSK and 32-APSK. In the article, there is an assumption that non-perfect of the Radio Frequency (RF) block has not been

mentioned. The first block in the chain of the signal is the Mode Adaptation block which is considered in detail in the next part of the article.

2.2 Mode and stream adaptation blocks

Unlike the standard DVB-S, Mode and Stream Adaptation blocks in the specification DVB-S2 process types of input such as MPEG-2 or MPEG-4. It provides a constant length base-band frame for the input of the next block. The output sequence of the Mode Adaptation block in the system is grouped into fix size frames, which is structured by base-band header (BBHEADER) and a DATA FIELD field. In the Stream Adaption block, the BBHEADER is randomized by a scrambler using the initial sequence is 100101010000000 for protecting data from unauthorized intruders.

2.3 FEC encoding block

In FEC encoding block, K_{bch} bits BBFRAME input stream is the input of Bose–Chaudhuri–Hocquenghem (BCH) encoder and Low-Density Parity-Check (LDPC) coders. Additional bits are appended after BBFRAME field with K_{bch} BCHFEC bits and K_{ldpc} LDPCFEC bits for interleaving. The Berlekamp algorithm used in the BCH encoder for correcting t -error. The generator polynomial of the BCH encoder is built by multiplying the first t -polynomials.

The system using LDPC codes over the satellite channel allowed to be set very close to the Shannon limitation. Ten different 1/4, 1/3, 1/2, 3/5, 2/3, 3/4, 4/5, 5/6, 8/9, 9/10 LDPC inner codes and BCH outer codes are designed for the range of bandwidth efficiency from 0.5 bits/symbol up to 4.5 bits/symbol. The long FECFRAME used for all rates. The output using 8-PSK, 16-APSK, and 32-APSK modulation will be interleaved to against burst errors by random allocation of the data stream in time. Hence, an error correction capability of the DVB-S2 system can be up to 12 bits.

2.4 Physical layer frame

In this block, the output of Physical Layer (PL) Frame is a payload of 64800 bits for long FECFRAME or 16200 bits short FECFRAME with a PLHEADER. In the broadcasting mode, the length of FECFRAME is 64800 coded bits regardless of code rates and modulation schemes. FECFRAME is packaged by a PLHEADER field and slots. This PLHEADER field plays a role at the synchronization and signaling information configuration at the receiver. It is modulated independently by $\pi/2$ -BPSK modulation schema for reducing the envelope fluctuation in comparison with the classical BPSK modulation. The PLHEADER field includes 26 bits Start of Frame (SOF) field on the left side of the

PLHEADER and 64 symbols linear binary code Physical Layer Signaling (PLS). PLS transports 7 bits of information with a minimum distance 32. 7 bits signaling information inform receivers the modulation scheme, code rates, pilot configuration, and the length of the LDPC coded data: 5 bits MODCOD field, one bit shows the length of frame is long or short and one bit expresses the status of pilot bits is in the frame or not. 64 bits PLSCODE is further scrambled by the scrambling sequence for improving the autocorrelation property and energy dispersal.

The standard DVB-S2 allows two operating modes: pilotless and pilot. Pilot signal consists of 36 symbols in every 16 slots for a priori knowledge of the satellite channel and carrier recovery. The number of slots in the frame is determined by the length of FECFRAME and the modulation type.

2.5 Modulation block

Unlike DVB-S that supports only QPSK modulation, Quadrature phase-shift keying (QPSK), 8-ary phase-shift keying, 16-ary amplitude and phase-shift keying (16-APSK) and 32-ary amplitude and phase-shift keying (32-APSK) modulations are supported in the DVB-S2 standard for the transmitted payload. In the broadcast applications, QPSK and 8PSK schemas are typically selected because of constant envelope modulations and satellite transponders may be operated near saturation. Baseband raised cosine filter using roll off factor 0.2; 0.25 or 0.35 is applied before the IQ modulation for reducing the inter-symbol interferences. QPSK modulation operates in the Signal-to-Noise Ratio (SNR) range below 6 dB while the 16-APSK and 32-APSK performs well in SNR levels between 8 and 16 dB. The following table demonstrates code rates and modulation schemas in the DVB-S2 system.

2.6 Satellite channel model under the rainy condition

In fact, in case of large variations of the atmospheric attenuation mainly due to the presence of rain, a tremendous waste of resources of the system for a great time percentage of the service can be strongly affected. Crane's assumption for an effective rain height and the convective raincell model to describe the horizontal variation of the rainfall structure is used. However, the variation of the elevation angle of broadband satellite systems is rather small and thus, the rain attenuation modeling assumptions that the main impact of rainfall on the channel is the amount of rainy water. Under the above assumptions, the definition of the Rician K-factor can be accordingly modified as the following formula that simply explains well how for rain attenuation parameter to be distributed in the satellite channel:

$$K_{\text{rician}} = 16.88 - 0.04R \quad (2)$$

where K_{rician} is the Rician K-factor (in decibels), and R is the rainfall rate (in mm/h). The equation shows that the K-factor with a negative sign is proportional to rain rate. This is expected because as the rain rate increases, the coherent power decreases and the incoherent scattered power increases. Therefore, pdf of the rain attenuation in the channel was derived on the rain attenuation random variable. This random variable is presented by Rician distribution variable. Rician processes were simulated by various simulators such as Sum of Sinusoids simulators [10] or Filtered Gaussian Noise (FGN) simulators [11], [12]. The channel model is approximated by using the sum of sinusoids method to generate the set of complex path gains a_k for the number of expected multipath components and these are uncorrelated with each other. At the receiver, we have the following received band-pass signal:

$$r(t) = h(t) * s(t) + w(t) \quad (3)$$

where $s(t)$ is the source signal, $w(t)$ is the band-pass noise, $h(t)$ is the band-pass channel impulse response which have the set $\{h_{Fi}\}$ of tap weights given by:

$$h_{Fi} = \sum_{k=1}^K a_k \sin c \left[\frac{\tau_k}{T_s} - n \right] \quad (4)$$

where T_s is the input sample period to the channel, $\{\tau_k\}$, where $1 \leq k \leq K$, is the set of path delays, K is the total number of paths in the Rician channel, $\{a_k\}$ with $1 \leq k \leq K$ is the set of complex path gains of the channel, N_1 and N_2 are chosen so that $|h_{Fi}|$ is small when n is less than $-N_1$ or greater than N_2 .

3 EXPERIMENTAL TEST

DVB-S2 simulation model in Matlab faces a lot of limitations, due to the ambition to raise significantly the data rate and non-perfect transmitter/receiver with the asynchronous carrier. In this part of the paper, the research presents and establishes a design of the test bed for evaluating DVB-S2 system.

The DVB-S2 channel consists of the rainy attenuation and noise in the path from a satellite to a user terminal is simulated over-the-air in the anechoic chamber room. Satellite communication system imperfections beside the noise, the transceiver imperfections are included in the form of carrier phase and frequency errors. The nonlinear distortion also occurring in radio frequency amplifiers is taken into account, but it has not yet been mentioned here.

An experiment test-bed was built to study the performance of the DVB-S2 system over the satellite channel under the rainy condition in the laboratory environment:

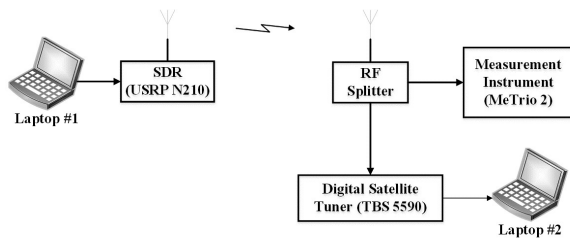


Fig. 2 General block of experimental test
Source: authors.

At the satellite receiver in Europe, the received broadcast signal is in Ku-band, while a low noise amplifier (LNB) converted signal from Ku-band to an Intermediate Frequency (IF) falling between 950 MHz and 2150 MHz. Therefore, a design of the satellite broadcast system operating at the IF frequency band is built from scratches for proving the validity of the proposed model. The experimental test uses software defined radio solutions that offer flexible software support for the specification and functions. We created an environment to prove the validity of the proposed model based on real-world radio propagation.

The RF environment was simulated in an anechoic chamber for isolating an undesirable noise. To test the proposed model, we use a testbed consisting of the first SDR is USRP N210 connected to a Linux laptop by the Ethernet interface as the transmitter. The transmitter on this first SDR sends the DVB-S2 signal with the fading channel to the received part. A program based on GNURadio application at the first laptop controls the signal on this SDR. Rayleigh channel takes in the Doppler frequency shift as a normalized value, non-line-of-sight parameters that is either true or false value. The Bit Error Rate (BER) performance of system was measured by the measurement instrument. The transport stream of the DVB-S2 system is observed by a digital satellite tuner as TBS 5590 with software applications. The measurement instrument furnishes general information at the RF level, such as IF polarization, SNR, symbol rate and modulation type as well as BER. SDR configured for using the sampling frequency is 10 Msps and the carrier frequency is 1.18 GHz. We assume that there are not non-linear effects of connectors, cable for concentrating the fading channel effect.

4 RESULTS

In the DVB-S2 standard, modulation schemas using various code rate correct the error for effective transmission. In this simulation, the article only focus on the analysis of modulation schemas using three

code rates: highest correction level using code rate 1/4, lowest correction level using code rate 9/10 and middle correction level using code rate 3/4. Fig. 3 demonstrates the BER performance of system using QPSK with code rates in the clear sky and rainy conditions.

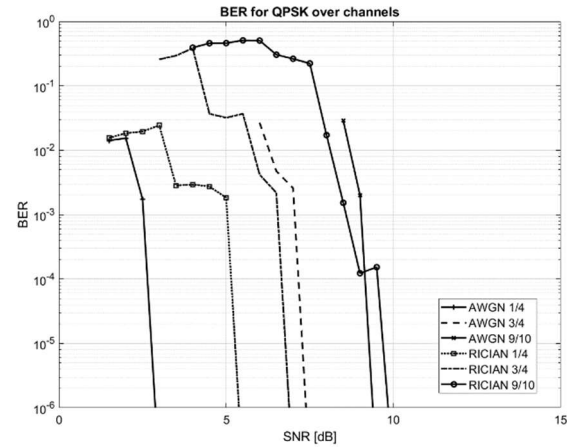


Fig. 3 QPSK
Source: authors.

SNR was changed in the range from 0 to 15 dB. The figure palpably noticed that QPSK modulation using the same rate code in the clear sky condition

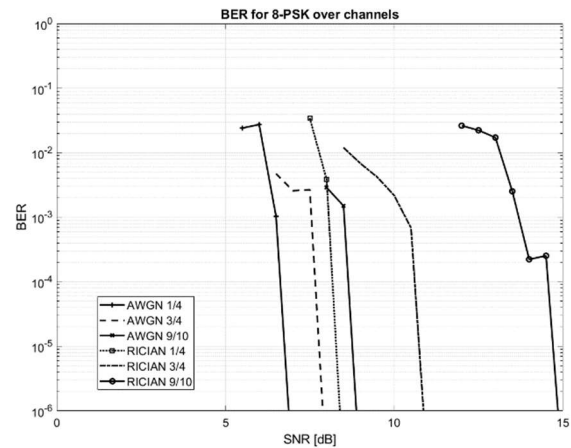


Fig. 4 8-PSK
Source: authors.

works well at BER values larger than 3 dB when system operates in the rainy environment. In the case 8-PSK, the system increases overall system performance by FEC codes at SNR higher than 6.5 dB.

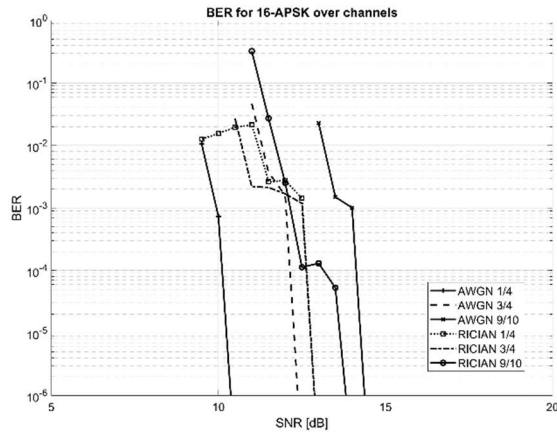


Fig. 5 16-APSK
Source: authors.

Fig. 5 shows the simulated BER performance for 16-APSK modulation with SNR in the range from 5 dB to 20 dB. In the rainy condition, the minimum desired BER of 10^{-2} is observed at SNR higher than approximate 10 dB of the system using the rate code 1/4.

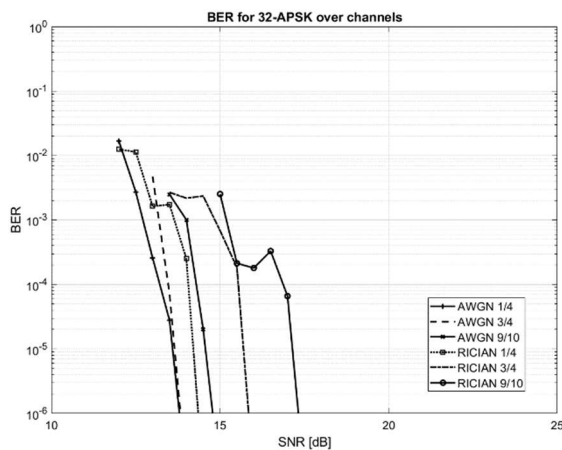


Fig. 6 32-APSK
Source: authors.

For the sake of comparison, the highest order modulation of DVB-S2 system with rate codes was introduced in the Fig. 8. SNR in the range from 10 to 25 dB was analyzed. It can be seen that the highest code rate saved the SNR ratio approximation 1 dB to 4 dB respectively compared to the same conditions. Moreover, comparing with simulation results [13] on Matlab, the result of the proposed model shows the quality of the signal is significantly decreased in the practical environment. It is logical that this model covers the basic aspects of the complex satellite system as well as some limitations of simple transceivers using SDR. However, the system is enough precision and reliability for the evaluation in the laboratory environment and testing satellite-to-Earth scenarios in the dynamic environment of the propagation path.

5 CONCLUSION

In this article, the basic technical features and principles of the DVB-S2 system were analyzed and the performance BER of system in the rainy condition was simulated over-the-air in the anechoic chamber room. As it can be seen from the results, BER performance of satellite signal was strongly influenced in the rain but reasonable code rate is the one of good solutions for against the environment condition. Results are analyzed and used to propose future research to discover optimal code rates and modulation schemas as well as its influence on the entire signal transmission.

References

- [1] ETSI EN 302 307-1 V1.4, 2014. Digital Video Broadcasting (DVB); Second generation framing structure, channel coding and modulation systems for Broadcasting, Interactive Services, News Gathering and other broadband satellite applications; Part 1: DVB-S2 [online]. October 2014. European Telecommunications Standards Institute. Available from: <<https://dvb.org/?standard=econd-generation-framing-structure-channel-coding-and-modulation-systems-for-broadcasting-interactive-services-news-gathering-and-other-broadband-satellite-applications-part-2-dvb-s2-extensions>>.
- [2] DOWNEY, Joseph A., MORTENSEN, Dale J., EVANS, Michael A., BRIONES, Janette C. and TOLLIS, N. 2016. Adaptive Coding and Modulation Experiment with NASA's Space Communication and Navigation Testbed. In: *34th AIAA International Communications Satellite Systems Conference*. 2016. p. 11.
- [3] GIRAULT, N., SMEYERS, O., DE GAUDENZI, R., MOREAU, C., ALBERTY, E., BIGRAS, A., DANE, G., SINGH, S. and MEGYERI, P. 2011. DVB-S2 Satellite Experiment Results. In: *29th AIAA International Communications Satellite Systems Conference (ICSSC-2011)* [online]. Nara, Japan: American Institute of Aeronautics and Astronautics. 28 November 2011. [Accessed 14 January 2020]. ISBN 978-1-60086-946-4. Available from: <<http://arc.aiaa.org/doi/10.2514/6.2011-8019>>
- [4] YOUSSEF, B., SALMI, K., HAJAR, Ch., AHMAD, B., ABDELHAMID, B. and DRISS, M. 2019. Measurement and test of a DVB-S2 satellite broadcast. In: *2019 7th Mediterranean Congress of Telecommunications (CMT)*. [online]. Fès, Morocco: IEEE. October 2019. p. 1–5. ISBN 978-1-72814-420-7.
- [5] HAO XU, RAPPAPORT, T. S., BOYLE, R. J. and SCHAFFNER, J. H. 2000. Measurements and models for 38-GHz point-to-multipoint radiowave propagation. In: *IEEE Journal*

- on *Selected Areas in Communications*. March 2000. Vol. 18, no. 3, p. 310–321. DOI 10.1109/49.840191.
- [6] CHEFFENA, M., BRATEN, L.E., TJELTA, T. and EKMAN, T. 2006. Space - time dynamic channel model for broadband fixed wireless access. In: *2006 First European Conference on Antennas and Propagation*. [online] Nice : IEEE, November 2006. p. 1–6. ISBN 978-92-9092-937-6.
- [7] LIOLIS, K. P., PANAGOPOULOS, A. D. and SCALISE, S. 2010. On the Combination of Tropospheric and Local Environment Propagation Effects for Mobile Satellite Systems Above 10 GHz. In: *IEEE Transactions on Vehicular Technology*. March 2010. Vol. 59, no. 3, p. 1109–1120. DOI 10.1109/TVT.2009.2036731.
- [8] CRANE, R. 1980. Prediction of Attenuation by Rain. In: *IEEE Transactions on Communications*. September 1980. Vol. 28, no. 9, p. 1717–1733. DOI 10.1109/TCOM.1980.1094844.
- [9] ADEYEMO, Zachaeus K., AJAYI, Olumide O. and OJO, Festus K. 2012. *Simulation Model for a Frequency-Selective Land Mobile Satellite Communication Channel*. 2012. Vol. 3, p. 14.
- [10] PÄTZOLD, M. 2004. On the Stationarity and Ergodicity of Fading Channel Simulators Based on Rice's Sum-of-Sinusoids. In: *International Journal of Wireless Information Networks*. April 2004. Vol. 11, no. 2, p. 63–69. DOI 10.1023/B:IJWI.0000034538.27413.60.
- [11] YOUNG, D. J. and BEAULIEU, N. C. 2000. The generation of correlated Rayleigh random variates by inverse discrete Fourier transform. In: *IEEE Transactions on Communications*. July 2000. Vol. 48, no. 7, p. 1114–1127. DOI 10.1109/26.855519.
- [12] FECHTEL, S. A. 1993. A novel approach to modeling and efficient simulation of frequency-selective fading radio channels. In: *IEEE Journal on Selected Areas in Communications*. April 1993. Vol. 11, no. 3, p. 422–431. DOI 10.1109/49.219555.
- [13] MANH, Hung N., RICHTEROVA, M. and VRSECKA, M. 2019. A Simulation of DVB-S2 System in Fading Channels. In: *2019 Communication and Information Technologies (KIT)* [online]. Vysoké Tatry, Slovakia : IEEE. October 2019. p.1–6. ISBN 978-80-8040-575-5.

Dipl. Eng. Hung NGUYEN MANH
PhD student at Department of Communication Technologies, Electronic Warfare and Radiolocation
Faculty of Military Technology
University of Defence
Kounicova 65
662 10 Brno

Czech Republic
E-mail: manhhung.nguyen@unob.cz

Assoc. Prof. Dipl. Eng. Marie **RICHTEROVÁ**, PhD.
Department of Communication Technologies,
Electronic Warfare and Radiolocation
Faculty of Military Technology
University of Defence
Kounicova 65
662 10 Brno
Czech Republic
E-mail: marie.richterova@unob.cz

Bc. Dipl. Eng. Tomáš **BŘINČIL**
PhD student at Department of Communication Technologies, Electronic Warfare and Radiolocation
Faculty of Military Technology
University of Defence
Kounicova 65
662 10 Brno
Czech Republic
E-mail: tomas.brincil@unob.cz

Hung Nguyen Manh was born in HaNoi, Vietnam. He received the M.Sc. degree in Electronic Engineering from PaiChai University, Korea, in 2012. He is PhD. candidate at the Department of Communication Technologies, Electronic Warfare and Radiolocation, Faculty of Military Technology, University of Defence since 2017. His research interests include sensor network, digital signal processing, signal analysis and modulation recognition.

Marie Richterová was born in Vyskov, Czech Republic, in 1965. In 1989, she graduated at the Faculty of Electro Engineering Technical University in Brno in branch Electrotechnology. Her scientific degree PhD. was obtained in the year 2002 at the Military Academy in Brno. Since 2009, she has been become the associate professor at the Department of Communication Technologies, Electronic Warfare and Radiolocation, Faculty of Military Technology, University of Defence. Her specialties are modulation recognition, special signal analysis and methods of pre-processing.

Tomáš Břinčil was born near Prague in 1991. In 2014 he got bachelor degree on Czech Technical University in Prague, Faculty of Electrical Engineering, Communications, in program Cybernetics and Robotics with focus on Sensors and Instrumentation. Since 2017 he studies his PhD. at the Department of Communication Technologies, Electronic Warfare and Radiolocation, Faculty of Military Technology, University of Defence in Brno. He is licensed radioamateur specialized on satellites communication.

OPTIMIZATION OF ULTRASONIC CUTTING TOOL GEOMETRY

Anton MYDLIAR

Abstract: This article solves the optimization of the ultrasonic cutting tool geometry. The parameterizable model of ultrasonic cutting tool was created in finite element (FE) software Ansys. The eigenfrequencies and modal shapes were extracted by modal analysis. Harmonic analysis showed the improvement of the amplitudes and the mechanical stresses of optimized ultrasonic tool. The results and measurements are summarized in the conclusion.

Keywords: Optimization; Eigenfrequency; APDL; Ultrasound; Cutting tool.

1 INTRODUCTION

The article is focused on the optimization of ultrasonic cutting tool which is a part of ultrasonic cutting product and performs the technological cutting operation. The generators of ultrasonic energy provide the longitudinal vibrational movement [1], [2], [7]. The magnetostriction and piezoelectric effect are often used like ultrasonic energy generators in a science and industry.

The geometric shapes of resonators are constructed with uniform or stepped change of a section. The well-known resonators' sections are

circle, conic and exponential. The optimization of ultrasonic tool geometry was realized by length and material changes in the past.

The FEA (FE analysis) and optimization methods bring advantage and cost-reduction of a tool construction [1]. The ultrasound is defined in the frequency range 20 kHz – 10 GHz [8], [9]. The particles of matter (e.g. solids) do a periodic vibrational movements in UZ field, which is generated by ultrasound generators. The important improvement brings the usage of a resonator which aim is to gain the amplitude of ultrasonic transducer or the ratio $A_{\xi k}/A_{\xi m}$ (Fig. 1).

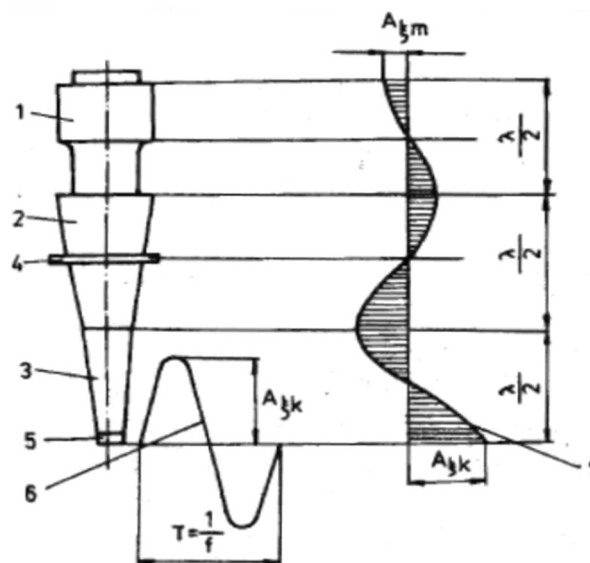


Fig. 1 The distribution of longitudinal vibration of ultrasonic vibrational product

1 – Ultrasonic transducer, 2 – primary resonator, 3 – secondary resonator, 4 – node, 5 – tool, 6 – time trend of displacement amplitude - oscillating movement, 7 – the wave trend along the vibrational product

Source: [7]

There is requested that ultrasonic tool which perform technological operation, vibrates with displacement amplitude 10-100 μm . This is possible to do with the resonators with variable change of a cross-section. They are called the mechanical transformers of displacement and velocity.

2 FE ANALYSIS AND MEASUREMENT OF ULTRASONIC CUTTING TOOL

To analyze ultrasonic cutting tool is necessary to create the parameterizable FE model in ANSYS. There were set the basic material constants (Young's modulus, density and Poisson's ratio), type of element and method of mesh with APDL macro.

The equations of motion for free vibration of undamped system with lumped parameters with n -degree of freedom are following

$$\mathbf{M}\ddot{\mathbf{x}} + \mathbf{K}\mathbf{x} = \mathbf{0} \quad (1)$$

where \mathbf{M} , \mathbf{K} , $\ddot{\mathbf{x}}$, \mathbf{x} are the mass matrix, stiffness matrix, the accelerations and displacements vectors, respectively.

The solution (2.1) we propose in the shape

$$\mathbf{x} = \mathbf{a}e^{i(\Omega t + \varphi)} \quad (2)$$

where i and φ are complex number and phase angle, Ω is unknown angular frequency and \mathbf{a} is unknown eigenshape or eigenvector [2], [8], [9].

The modal analysis shows the longitudinal vibrational shape is at frequency 28.64 Hz and there is the bending of the wedge (Fig 2).

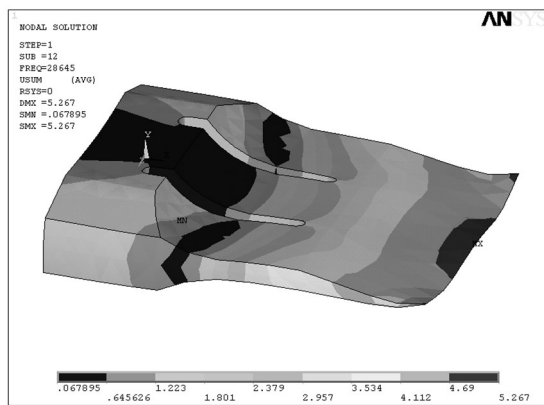


Fig. 2 Longitudinal eigenshape with the transversal wedge bending
Source: author.

2.1 Sensitivity analysis

There is importance to know the influence of parameter change on the vibrational modes. The ultrasonic cutting tool should have a uniform vibration of wedge and bending vibration should be minimized or canceled.

There are the parameters depicted on Fig. 3 for a sensitivity analysis - SA.

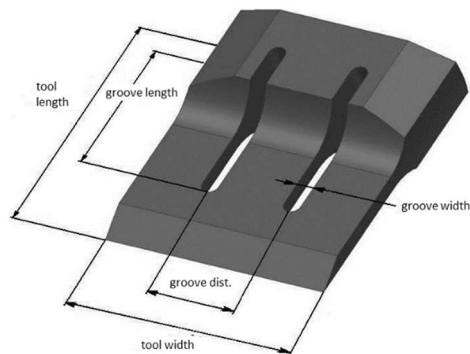
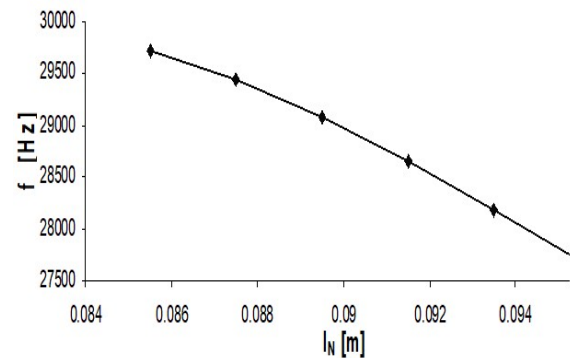


Fig. 3 Dimensions for SA
Source: author.

Graph 1 displays the evolution of the resultant eigenfrequencies for the different tool lengths.



Graph 1 Eigenfrequency versus tool length
Source: author.

There are depicted the eigenshapes for parameter tool length when it has minimum value – Fig. 4 and maximum value Fig. 5.

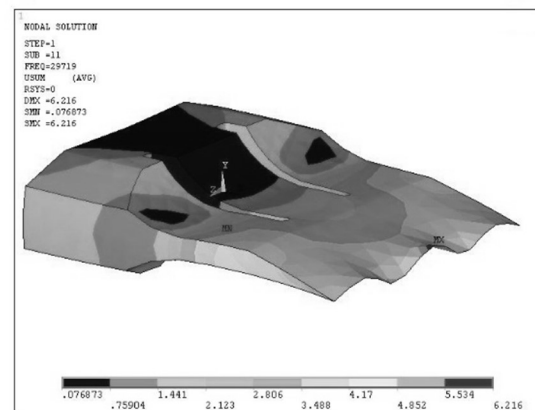


Fig. 4 Eigenshape at 29719 Hz - the minimum length of cutting tool
Source: author.

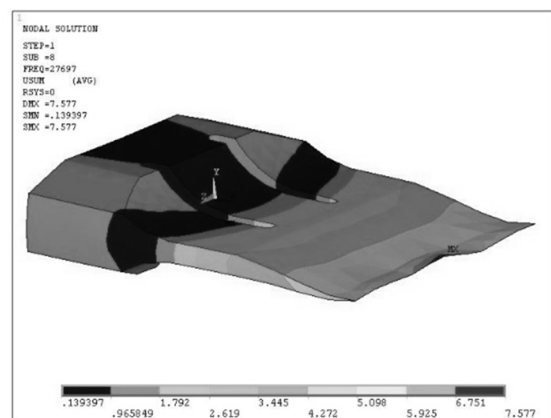


Fig. 5 Eigenshape at 27697 Hz - the maximum length of cutting tool
Source: author.

2.2 Optimization and results

In Ansys optimization toolbox is necessary to define:

1. Design Variables;
2. State variable;
3. Objective function.

The first order method was used in optimization process what is gradient method [1]. The trend of objective function for the prescribed restriction and design variables is depicted on Fig. 6.

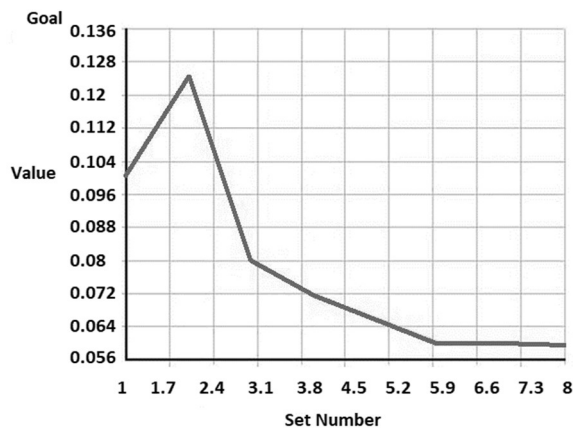


Fig. 6 Relation between trend of objective function and no. of iteration
Source: author.

The optimized eigenfrequency 30015 Hz and eigenshape is depicted on Fig. 7.

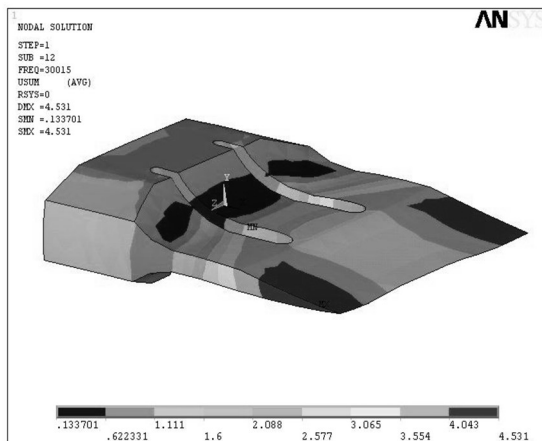


Fig. 7 Optimized eigenshape at 30015 Hz
Source: author.

2.3 Measurement

The measurement of electric quantities of piezoelectric transducer was performed on ultrasonic product. The piezoelectric transducer was connected to the source of alternating voltage (Fig. 8) [5], [6].

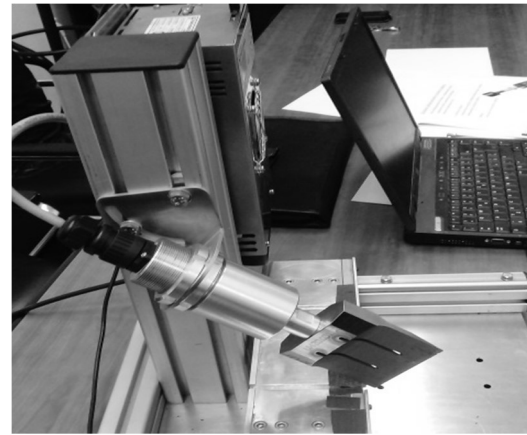


Fig. 8 Ultrasonic cutting tool
Source: author.

The target of the experiment was the measurement of the frequency dependent electric quantities in electric circuit of PZT transducer of ultrasonic product [3], [4]. The measurement no.1 found the frequency 30.52 kHz and electric current, by which the ultrasonic cutting product vibrated longitudinal and provided the technological operation cutting. The measurement no. 2 found frequency 30.76 kHz, by which tool or ultrasonic product had bending eigenshape and working mode was worse and the tool could fail. There was acoustic disturbance in the frequency range 30.6 – 31kHz. The measurement no. 3 found electric voltage by oscilloscope on PZT transducer by frequency which correspond longitudinal eigenshape of ultrasonic cutting tool. The measurements no. 1 and 3 show the same frequency 30.52 kHz. The measurement results are summarized in the Table 1.

Tab. 1 The measurement of frequency characteristic of ultrasonic transducer

Measurement no.	Frequency range [kHz]	f [kHz]
1	29-31	30.52
2	30.6-31	30.76
3	29-31	30.52

Source: author.

2.4 Harmonic analysis of ultrasonic product

Model of ultrasonic product was built in ANSYS. The model consists of ultrasonic transducer, resonator and cutting tool. Modal analysis found eigenfrequency and eigenshape for optimized tool. The electric voltage excited PZT transducer and for harmonic analysis of coupled field the distribution of displacement amplitudes in the direction of Z are depicted on Fig. 9. The cutting edge has the displacement amplitudes between 4 -11 μm .

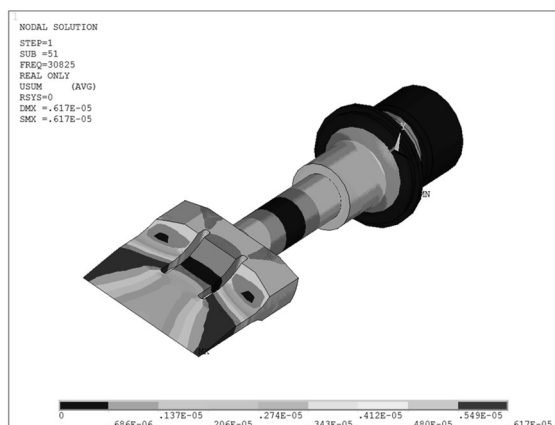


Fig. 9 Eigenfrequency 30.8 kHz and the amplitude distribution in the Z axis direction
Source: author.

The next model of ultrasonic cutting tool was analyzed and the displacements are depicted on fig. 10. The difference are e.g. longer holes and others construction changes.

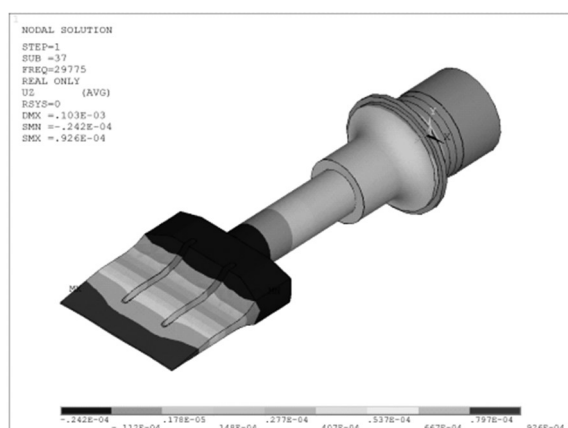


Fig. 10 Eigenfrequency 29.8 kHz the amplitude distribution in the Z axis direction
Source: author.

The displacement amplitude has the maximal values 88 μm .

3 CONCLUSION

The optimization of geometry gives the model which satisfies the range of design variables and achieves the objective function during optimization process. The vibrational optimizations of the resonators with the complicated geometries require the usage of FEA with the mathematical models for an optimization.

By optimization of the ultrasonic cutting tool geometry was achieved the variants where the conditions were satisfied for eigenfrequency and

eigenshape. The optimization tuned the longitudinal eigenshapes at frequency 30 kHz. Dynamic harmonic analysis confirmed that optimized tool vibrates longitudinal.

By the measurement frequency dependent electric quantities (electric current and power) of PZT transducer was found the working frequency of ultrasonic product. The measurements no. 1 and 2 showed the trends of electric current and power in dependency on frequency. The measurement 3 was conducted by oscilloscope direct on PZT transducer in working mode. The measurements no. 1 and 3 found the same vibrational eigenshape of ultrasonic product at 30.52 kHz. The measurement no. 2 found the bending eigenshape where was acoustic disturbance.

The amplitude-frequency characteristic determined the displacement amplitudes of ultrasonic cutting product wedge in range 4-11 μm at 30.8 kHz. The next model was showed the uniform displacement amplitude of wedge (in cutting direction) about 88 μm at 29.8 kHz.

References

- [1] Ansys, Inc.: Ansys Documentation.
- [2] BENČA, Š.: *Výpočtové postupy MKP pri riešení lineárnych úloh mechaniky*. Bratislava : STU, 2004. 150 p. ISBN 80-227-2031-1.
- [3] GUTTEN, M., KORENČIAK, D.: *Meranie a meracie systémy II*. Laboratórne metódy. Žilina : EDIS v Žiline, 2012. 156 p. ISBN 978-80-554-0584-1.
- [4] KARRIS, T. S.: *Circuit Analysis II*. Fremont : Orchard publications, 2009. ISBN 978-1-934404-19-5.
- [5] ONWUBOLU, G. C.: *Mechatronics principles and applications*. Elsevier, 2005. 645 p. ISBN 0 7506 6379 0.
- [6] SHERRIT, S., LEARY, S. P., DOLGIN, B. P., BAR-COHEN, Y.: Comparison of the Mason and KLM equivalent circuits for piezoelectric resonator in the thickness mode, IEEE. In: *Ultrasonic symposium*, 2, vol. 2, 1999. pp. 921-926.
- [7] ŠVEHLA, Š., FIGURA, Z.: *Ultrazvuk v technológii*. Bratislava : Alfa, 1984.
- [8] ŽIARAN, S.: *Technická diagnostika*. Bratislava : STU, 2013. 332 p. ISBN 978-80-227-4051.
- [9] ŽIARAN, S.: *Znižovanie kmitania a hluku v priemysle*. Bratislava : STU, 2006. 339 p., ISBN 80-227-2366-5.

Dipl. Eng. Anton **MYDLIAR**, PhD.
Armed Forces Academy of General M. R. Štefánik
Demänová 393
031 01 Liptovský Mikuláš
Slovak Republic
E-mail: anton.mydliar@aos.sk

Anton Mydliar - was born in Ružomberok, Slovakia in 1984. He graduated from Slovak University of Technology in Bratislava in 2018 where he received Ph.D. on Department of Applied Mechanics. He is the assistant professor at the Department of Mechanical Engineering, Armed Forces Academy of General M. R. Štefánik in Liptovský Mikuláš.



New Trends in Signal Processing 2020

October 14-16 2020

in Hotel Chopok, Demänovská dolina, Slovakia

(<http://ntsp2020.aos.sk>)

The conference covers main topics:

- Signal Processing
- Applied Electronics
- Information and Communication Engineering
- Microwave Engineering
- Signal Processing in Military Applications

All papers for the NTSP 2020 will be reviewed and published in electronic form on DVD with ISBN 978-1-7281-6154-9 and ISSN 1339-1445 and will be submitted to *IEEE Xplore* database.

Contact:

Address: **NTSP 2020**

Department of Electronics
Armed Forces Academy of General M. R. Štefánik
Demanova 393, 031 01 Liptovsky Mikuláš, Slovak Republic

E-mail: ntsp2020@aos.sk

Administrators: Ing. Roman BEREŠÍK, PhD. Phone: +421 960 423140 Mobile: +421 905 934287
Ing. Marián BABJAK, PhD. Phone: +421 960 423876 Mobile: +421 915 829405

FORMAL MODEL OF DECOMPOSITION AND MAPPING IN ACCELERATED CLUSTER ARCHITECTURE

Miloš OČKAY, Ľubomír DEDERA

Abstract: Accelerated Cluster is currently the core architecture that drives computing performance in HPC applications. Graphic accelerators push the boundaries of the established parallel cluster architecture in a significant way, and in the near future, it will enable the achievement of exaflops milestone in HPC systems. Presented paper outlines basic elements of accelerated cluster architecture. It also explains decomposition and mapping in multistage architecture, using the data and task parallelism. Presented formal model describes the decomposition on all stages, allowing more efficient mapping and achieving an accelerated solution in a complex problem.

Keywords: Computer cluster; GPU; Accelerator; Parallel; Decomposition; Mapping.

1 INTRODUCTION

A computer cluster is a well-established architecture in High Performance Computing (HPC). The cluster consists of a number of computational nodes connected by high-speed networks, allowing hundreds of processors to be involved in the calculation. The computer cluster is not a new architecture and appeared in the 1960s. Its primary purpose was to overcome computational and memory constraints. Incorporating accelerators into a cluster architecture makes it an ideal computational tool for complex computational problems [1]. Nowadays, the most commonly used accelerators are based on GPUs and allow computing power to rise up to several PetaFLOPS [2]. The downside of this approach is more complex programming, often unique to each task. The accelerated cluster architecture is a key computational architecture, which will allow an entry into the exascale HPC era in the near future [3].

2 ACCELERATED CLUSTER ARCHITECTURE

An accelerated cluster is a specific extension of the cluster architecture. A graphics accelerator can be included in the cluster node to enrich the node's computing capabilities with a massively parallel processor. If an accelerator is included in a cluster node, the memory system, communication model, decomposition and mapping of the problem to the architecture are modified. To simplify the architecture design and decomposition process, we have divided the accelerated cluster into three stages (Fig. 1):

- cluster node stage,
- basic cluster stage,
- accelerator stage.

The cluster node stage is represented by a single CPU based computing system. The basic cluster stage consists of interconnected cluster nodes forming the compute cluster [4]. The accelerator stage consists of graphical accelerators located within the cluster nodes. The cluster may be built on a heterogeneous basis and may contain accelerated and non-accelerated nodes. A cluster node that does

not include a graphics accelerator is referred to as a non-accelerated node. Accelerated nodes may contain one or more accelerators.

Compute and memory resources can be clearly identified at individual stages. It is also possible to identify the cost of data transfers.

Due to the arrangement of the computational resources, it is possible to identify the parallel and serial portions of the problem at individual stages. At the basic cluster stage, a cluster nodes form the scalable cluster. Several cluster nodes can be included in the calculation and the algorithm can be implemented in parallel. CPU is a main computational resource at the cluster node stage. Although a single CPU may contain multiple cores or a cluster node may contain several CPUs, this single element is considered serial computational resource. The cluster node can use an accelerator stage for parallel calculations. At this stage, GPU is the basic computational element and it allows to implement massive parallelism. The algorithm should respect this distribution and use the parallel computational resources to implement parallel tasks of the problem and map the serial tasks to serial computational resources. Identifying parallel and serial tasks within the problem and map them correctly to individual stages of the accelerated cluster is an important, but not the only factor for successful acceleration. The accelerated cluster has two parallel computational stages. The first consists of a group of cluster nodes at the basic cluster stage. Graphical accelerators represent the second. While the basic stage creates parallelism by clustering the nodes, the graphics adapter has a separate massively parallel processor composed of multiple stream processors. Clustering of graphics adapters is possible but limited in terms of scalability. Grouping graphics adapters at an inter-node level is very inefficient in terms of communication cost. This implies that basic stage parallelism differs from accelerator stage parallelism. While it appears to be the most efficient task parallelism at the basic cluster stage, data parallelism is more efficient at the accelerator stage.

As we will see in the process of decomposition, the use of data parallelism at the basic cluster level is not excluded, sometimes it is even necessary.

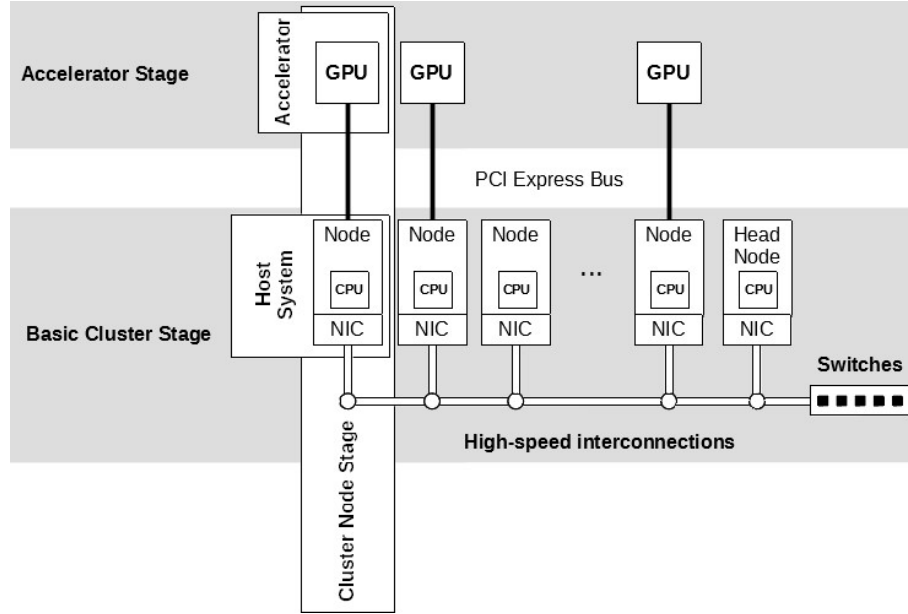


Fig. 1 Accelerated cluster architecture
Source: authors.

3 DECOMPOSITION AT THE BASIC STAGE USING TASK PARALLELISM

The P problem can be expressed as a set of its partial tasks T_1, \dots, T_n , ie. $P = \{T_1, T_2, \dots, T_n\}$, $n \in N$ (Fig. 2).

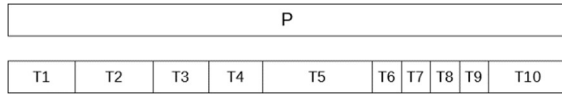


Fig. 2 Problem decomposition
Source: authors.

Dependencies of the particular partial tasks T_i can be represented by means of an oriented graph of dependencies of partial tasks (GDPT) (Fig. 3). GDPT is the set of vertices which represent partial tasks (together with a specific partial task $Result$ and oriented edges which express dependencies between individual tasks. We use the $Result$ as a specific task to indicate the completion of the entire calculation.

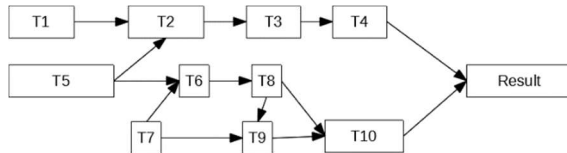


Fig. 3 An example of Graph of Dependencies of Partial Tasks
Source: authors.

Definition 1.1.: Let the $P = \{T_1, T_2, \dots, T_n\}$, $n \in N$ be a problem. Then the graph of dependencies of partial tasks GDPT is called oriented graph (V, E) , whose set of vertices $V = P \cup \{Result\}$ and set of edges

$E \subset P \times (P \cup \{Result\})$. E contains oriented edge (T_i, T_j) just when the output of the partial task T_i is connected to the input of the partial task T_j .

Independent tasks are not dependent on the output of other partial tasks. Therefore, they can be used as primary elements of task sequences. These tasks do not create downtime and allow creating as many task sequences as the number of independent partial tasks included in the problem.

Definition 1.2.: The task T is called an independent task if there is no oriented edge (V, T) , $V \in P$ in the set of oriented edges.

Using GDPT, we can divide problem P into disjunctive subsets of tasks P_1, \dots, P_k so that

$$P = \bigcup_{i=1}^k P_i, \quad (1)$$

$$P_i \cap P_j = \emptyset \text{ for } i \neq j,$$

if two partial tasks belong to the same subset of tasks P_i , then there must be (oriented) path between the vertices in the GDPT.

Since the GDPT may contain loops between tasks and the tasks of the individual P_i subsets must be arranged based on their mutual dependencies, for each subset of P_i described above, we define the sequence of tasks PT_i :

$$PT_i: N_{k_i} \rightarrow P_i, \quad N_{k_i} = \{1, \dots, k_i\}, \quad (2)$$

which will reflect the order of execution of individual tasks within a subset P_i . k_i represents the number of elements in a given subset of tasks, and since there may generally be a cyclic dependency between the tasks, the above representation may not be bijective.

The elements of the individual sequence of tasks are arranged on the basis of the GDPT, whose arrangement determines in which order the individual partial tasks within the individual sequence of tasks must be performed.

Definition 1.3.: The calculation plan CP will be understood as a set of task sequences.

$$CP = \{PT_i, i = 1, \dots, k\}, P_i \in P \quad (3)$$

where each sequence of tasks contains only partial tasks pertaining to one particular problem P .

Definition 1.4.: Initial calculation plan is a calculation plan in which each sequence of tasks contains exactly one independent task.

Independent tasks allow us to create several task sequences that can be processed in parallel. The initial calculation plan will contain as many sequences as the number of independent tasks in the problem.

Definition 1.5.: An optimized calculation plan is a calculation plan on which the optimization steps O1 and O2 were performed.

Definition 1.6.: Let $G = (V, E)$ be GDPT and

$$t_V: V \rightarrow \langle 0, \infty \rangle \quad (4)$$

$$t_E: E \rightarrow \langle 0, \infty \rangle \quad (5)$$

in which $t_V(T_i)$ represents the execution time of the task T_i , and $t_E(T_i, T_j)$ indicates the communication complexity between the output of the task T_i and the input of the task T_j . Then the oriented rated graph, which is obtained from the GDPT by evaluating its vertices using the t_V mapping and the edges using the t_E mapping, is called the rated graph of dependencies of partial tasks RGDPT (Fig. 4).

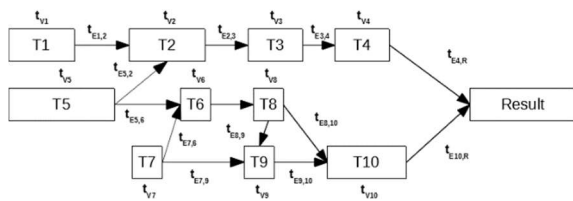


Fig. 4 An example of Rated Graph of Dependencies of Partial Tasks
Source: authors.

RGDPT unlike GDPT is an oriented graph whose vertices and edges are rated. Time values can be obtained by realizing the problem using the initial calculation plan, or in another way, e.g. testing and/or approximation.

Definition 1.7.: The time period of the calculation $\langle t_1, t_2 \rangle$ is called the serial period of the problem, if there is exactly one task active at each time $t \in \langle t_1, t_2 \rangle$.

The time period of calculation $\langle t_1, t_2 \rangle$ is called the parallel period of the problem if at each time $t \in \langle t_1, t_2 \rangle$ the number of the active tasks k is equal to the number of sequences of tasks.

The period of the calculation $\langle t_1, t_2 \rangle$ is called a partially parallel period of the problem if at each time $t \in \langle t_1, t_2 \rangle$, $1 < \text{number of active tasks} < k$.

Definition 1.8.: The time interval $\langle t_1, t_2 \rangle$ for the PT_i task sequence is called idle if the time period $\langle t_1, t_2 \rangle$ is a serial or partially parallel period and no task is active in the interval $\langle t_1, t_2 \rangle$ in the given task sequence PT_i (Fig. 5).

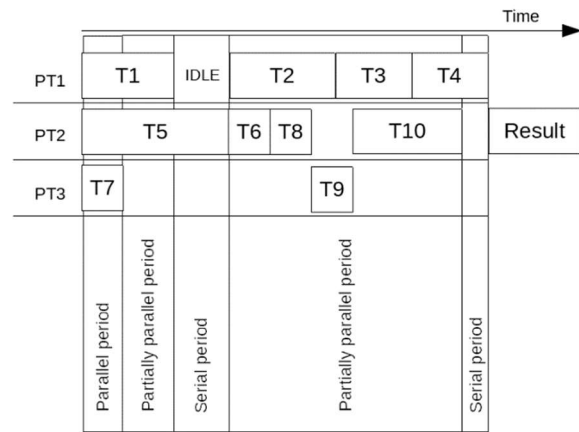


Fig. 5 An example of parallel, partially parallel and serial time periods of problem
Source: authors.

The objective of the optimization step O1 is to maximize the use of hardware computing resources. The transfer of tasks to replace idles is possible if $t_V(T_i) \leq t_{idle}$. Such move does not increase the problem processing time, but it reduces inter-nod communication.

The objective of the optimization step O2 is to reduce communication overhead. It is necessary to consider the possibility of moving the tasks which communicate with the tasks included in other task sequences to these sequences.

Let $T \in PT_i$ be a task included in the PT_i task sequence. Then its input communication complexity (ICC) will be defined as follows

$$ICC_{PT_i}(T) = \sum_{(V,T) \in E, V \notin PT_i} t_E(V, T) \quad (6)$$

and output communication complexity (OCC) will be defined with the following formula

$$OCC_{PT_i}(T) = \sum_{(T,V) \in E, V \notin PT_i} t_E(T, V) \quad (7)$$

Formulas (6) and (7) take into account the fact that communication complexity is neglected for the tasks assigned to the same sequence.

$$\max(T) = \max_{V \in P} \{t_E(T, V)\} \text{ and } \quad (8)$$

$V_{\max}(T)$ is a task to which it applies
 $t_E(T, V_{\max}(T)) = \max(T)$

Thus $V_{\max}(T)$ is the task to which from T leads the edge with the maximum rating (ie. with the maximum communication complexity).

Further, $PTV_{\max}(T)$ is the sequence of tasks to which $V_{\max}(T)$ belongs. Then in the optimization step O2 we move the partial task T to the sequence of tasks $PTV_{\max}(T)$ if

$$\left(OCC_{PTV_{\max}(T)}(T) + ICC_{PTV_{\max}(T)}(T) \right) < \left(OCC_{PT_i}(T) + ICC_{PT_i}(T) \right) \quad (9)$$

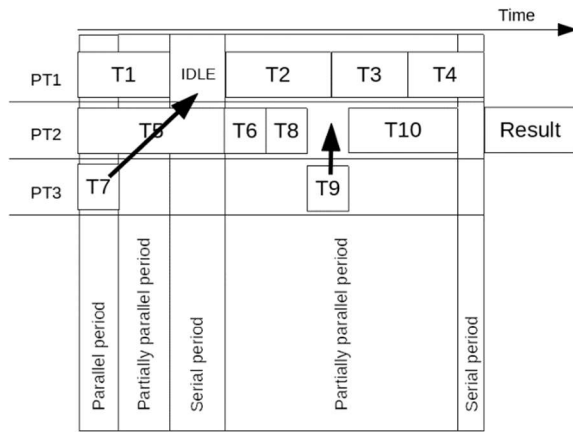


Fig. 6 An example of optimization
 Source: authors.

During the O2 optimization step it is necessary to determine between which tasks the maximum communication complexity arises. Based on formula 9, the input and output communication complexity is compared before and after the transfer of task T to $PTV_{\max}(T)$. If the formula 9 is true, the task T is moved to $PTV_{\max}(T)$ and the result is the optimization of the communication costs for the task. If the formula 9 does not apply, the task remains in the original sequence of tasks (Fig. 6).

If the partial tasks of the problem are completely independent, these problems can be mapped to separate computational nodes.

4 DECOMPOSITION AT THE BASIC STAGE USING DATA PARALLELISM

Definition 1.9.: If the problem $P = \{T_1, T_2, \dots, T_n\}$, $n \in N$ contains exactly one independent task T_1 and at the same time GDPT contains just edges (T_1, T_2) , (T_2, T_3) , (T_3, T_4) , ..., (T_{n-1}, T_n) , then we call the P problem a serial problem.

It is not possible to effectively use task parallelism for a serial problem. If the processed data can be decomposed into smaller units (subsets) with the same data structure that do not directly depend on each other (the task uses just one subset of the

processed data during the calculation), then it is possible to parallelize the serial problem using data parallelism. If data parallelism is used, several copies of the problem are created and they process different parts of the data on separate computational nodes (Fig. 7).

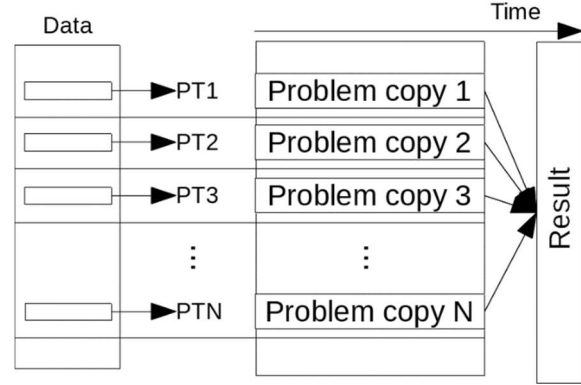


Fig. 7 Data parallelism
 Source: authors.

5 DECOMPOSITION AT THE NODE STAGE

In order to use the acceleration stage, we need to express the task T as a sequence of its portions (u_1, u_2, \dots, u_k). Projection $ACCT_i: u \rightarrow \{S, P\}$ assigns the way the portions (u_1, u_2, \dots, u_k) are processed. The portion labeled S (serial) is processed by the CPU and the portions labeled P (parallel) are processed using the GPU [5] (Fig. 8).

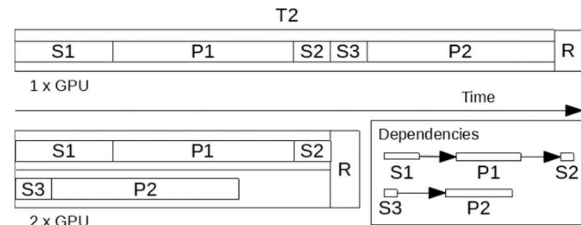


Fig. 8 Task portions S and P
 Source: authors.

6 MAPPING PROBLEM TO ARCHITECTURE

The accelerated cluster AC can be expressed as a set of cluster nodes K containing GPU accelerators $AC = \{K_i, i = 1, \dots, N\}$.

The mapping of the problem to computational resources can be expressed at cluster node level as projection of CP to AC with the mapping of serial and parallel portions of tasks described in the previous section. Problem to architecture mappings use the following scheme:

- problem $P \rightarrow$ accelerated cluster AC ,
- task sequence $PT \rightarrow$ cluster node $K_i, i \in 1 \dots N$,
- serial portion $S \rightarrow$ CPU,
- parallel portion $P \rightarrow$ GPU.

7 CHARACTERISTICS OF THE ADAPTABLE PROBLEM GROUP

The decomposition process allows us to tailor selected problems so that they can be implemented on an accelerated cluster architecture, and the overall result is available in a shorter time in contrast with a serial system. Furthermore, decomposition makes it possible to identify problems whose processing is not effective on accelerated parallel architecture and does not speed up the overall solution of the problem.

With accelerated cluster architecture, the decomposition process is divided into several stages. At the basic cluster stage, the problem is divided into the tasks. The distribution should be coarse-grained and the tasks should not be elementary. If there is a high degree of dependency between the tasks, or if the tasks have serial characteristics, it is necessary to reconsider the structure of the processed data package. If the data package contains repetitive data or data that can be processed independently, the task can be effectively parallelized using the above methods. Problems that do not have these characteristics are not suitable for processing using accelerated cluster architecture.

At the cluster node stage, tasks are divided into serial and parallel portions. The CPU handles serial portions while parallel portions are processed by the GPU. The quality of the processing of parallel portions at the acceleration stage are highly dependent on the level of code optimization and hardware utilization [6].

Decomposition is looking for the appropriate arrangement of the individual parts of the problem and the data so that the problem can be mapped to accelerated cluster resources in order to achieve the best problem-solving performance.

From the perspective of decomposition, problems with low level of dependency among the tasks or/and low level of dependencies within the processed data will be the best candidates for the mapping to the accelerated cluster architecture [7].

References

- [1] KOLLÁR, J.: *Metódy a prostriedky pre výkonné paralelné výpočty*. Košice : Elfa, 2003. 106 p. ISBN 80-89066-70-4.
- [2] MEUER, H., DONGARRA, J., STROHMAIER, E.: *TOP500 Supercomputing sites*. TOP500, 2020, [cit. 2020]. Available at: <<http://www.top500.org/>>
- [3] INTEL: *Driving Exascale Computing and HPC with Intel*. 2019, [cit. 2019] Available at: <<https://www.intel.com/content/www/us/en/high-performance-computing-fabrics/omni-path-driving-exascale-computing.html>>
- [4] BOOKMAN, CH.: *Linux Clustering: Building and Maintaining Linux Cluster*. Indianapolis : Sams Publishing, 2002. 300 p. ISBN 1-57870-274-7.
- [5] SHOWERMAN, M., ENOS, J., PANT, J. A., KINDRATENKO, V., STEFFEN, C., PENNINGTON, R., HWU, W.: QP: A Heterogeneous Multi-Accelerator Cluster. In *10th LCI International Conference on High-Performance Clustered Computing*, 2009. Available at: <http://web-test.ncsa.illinois.edu/~kindr/papers/lci09_paper.pdf>
- [6] RYOO, S., RODRIGUES, Ch., BAGHSORKHI, S., STONE, S., KIRK, D., HWU, W.: Optimization principles and application performance evaluation of a multithreaded GPU using CUDA. In *Proc. of 13th ACM SIGPLAN Symposium*, New York : ACM, 2008. ISBN 978-1-59593-795-7.
- [7] OČKAY, M., DROPPA, M.: Embarrassingly parallel problem processed on accelerated multi-level parallel architecture. 2011. Informatics 2011 : proceedings of the eleventh international conference on Informatics, November 16 - 18, 2011 Rožňava, Slovakia. Košice: Technical University of Košice, 2011. ISBN 978-80-89284-94-8. s. 29-32 p.

Dipl. Eng. Miloš OČKAY, PhD.

Armed Forces Academy of General M. R. Štefánik
Department of Informatics
031 01 Liptovský Mikuláš
Slovak Republic
E-mail: milos.ockay@aos.sk

Assoc. Prof. RNDr. Ľubomír DEDERA, PhD.

Armed Forces Academy of General M. R. Štefánik
Department of Informatics
031 01 Liptovský Mikuláš
Slovak Republic
E-mail: lubomir.dedera@aos.sk

Miloš Očkay is an assistant professor at the Department of Informatics at the Armed Forces Academy in Liptovský Mikuláš. In 2003 he graduated (MSc.) at Military Academy in Liptovský Mikuláš as a civil student. He holds PhD. degree in the field of Informatics, received in 2012 from the Technical University of Košice. His scientific research is focuses on parallel computing, computer graphics and steganography.

Ľubomír Dedera works as an Associate Professor at the Department of Informatics, Armed Forces Academy in Liptovský Mikuláš. He graduated (RNDr.) from the Faculty of Mathematics and Physics, Comenius University in Bratislava in 1990. He received a PhD. degree in Artificial Intelligence from the Military Academy in Liptovský Mikuláš in 1997. His research interests include computer languages, computer security and artificial intelligence.

CYBER SECURITY STATE IN REAL ENVIRONMENT

Martin DROPPA, Marcel HARAKAL

Abstract: The purpose of this document is to give an insight into the wide area of world of cyber security and state of the cyber security in AFA (Armed Forces Academy in Liptovský Mikuláš) environment. All IT studies confirm that cyber attacks are increasing annually. Few years ago, as many as 70 % of companies worldwide were affected and loss probability of economic value was € 190 billion. Preventing attacks that are trying to penetrate the internal network through firewalls, and the lack of user awareness, is not a matter of choice but of necessity. Are the companies of 21st century ready for cybercrime? In the end, the findings from the state of cyber security at AFA are described. Based on a simple analysis of the situation in the AFA environment, the proposed recommendations, measures, methods of eliminating cyber attacks are briefly described.

Keywords: Detection; Threat; Assessment; Malware; Attack; Network; Vulnerability; Exploits.

1 INTRODUCTION

The answer is more than worrying. The global EY survey (a professional consultancy company) claims that only 36 % of companies could detect cyber attack. Many companies don't even know that they have been targeted by cyber-attackers and how many of their corporate data are already for sale. By comparison, in 2017 there was an increase of up to 255 % of cyber attacks compared to the previous year [1].

1.1 Why address the biggest SOC (security operations center) problems?

While more and more companies are responding to attacks by introducing a variety of non-conceptual repair systems, security teams do not have enough insight and automated technologies to detect and respond to these advanced threats. They lack automation to ensure maximum use of resources. The ability to detect today's cyber threats is not enough. Security teams use different so-called interconnected disparate systems that tend to increase complexity instead of providing solutions. Working with these isolated and inconsistent systems unnecessarily complicates the problem because they have to work in different environments, on different screens, which takes too much time. This in turn leads to overloading of alarms, slower investigation of incidents and eventually delayed response. The attackers thus have more time to operate in the invaded network. The data are at increased risk if the investigation, response, and consequently the attacker's stay in your system takes longer than allowed.

Only an antivirus is not enough

Many companies consider the IT security problem to be resolved when antivirus software is installed, which is an absolute gamble for the company with data and sensitive data. Cyber threats are steadily rising to infrequent company cannot be sure it is safe. Much of the company has already invested a significant amount of resources in an important prevention that is effective in routine attacks, but does

not provide adequate protection for the current dynamic environment of targeted attacks. Prevention alone is no longer sufficient.

Failure under the complexity and size of security infrastructures

Conscious organizations have invested in implementing security products that initiate or trigger an alarm whenever they detect any suspicious event. Some companies have even employed and trained security personnel who are trying to respond to alarms. Theoretically, it is a sensible concept, but it does not work in practice. Companies fail under the complexity and size of the security infrastructures they have created, and are losing many generated alarms. Very often, they encounter the problem of missing employees who would sort, validate, investigate, prevent further dissemination or provide remedies based on generated alarms or events.

The problem is the budget

Organizations usually do not have a budget for security teams, and they cannot get enough of the necessary qualified security specialists to handle such huge amounts of alarms. As a result, security teams do not manage to respond to generated alarms, and in many cases, serious attacks escape their attention. However, most companies will only realize this when something has happened. It is therefore essential to calculate the risk first. Organizations must be able to detect attacks that penetrate existing perimeters, respond to them, and stop them before they cause damage.

Logs are not enough to detect attacks

To achieve this, it is necessary to realize that the log events are not sufficient information about the attacks. Fragmented data is difficult to correlate and evaluate. It is necessary to introduce solutions / systems that contain information about infiltration or attacks, including their context, not only to obtain additional isolated data. This will help companies avoid losses or respond in a timely manner. In order for security teams to be able to respond effectively to

a greater number of security threats in the short term, perfect visibility is essential to enable quick decision making. In contrast to the traditional solution, need is to find one that can, along with visibility, also provide automated detection of end point scans, which saves time and costs and loss. So what should be the key questions of the corresponding system?

- Has the attack reached its target?
- Has the attack been activated?
- What happened before and after the attack?
- Has this activity occurred in any other system in our environment?

There is no zero risk on IT security. It is imperative that security teams continue to advance, acquire new technology solutions, and always be ahead of potential attackers. Despite state-of-the-art technology, security teams face challenges they need to fight 24 hours a day. However, the challenges are a better future, and the future includes better IT security.

2 FOR CHALLENGES THAT SECURITY TEAMS HAVE TO COMBAT

Many companies are losing in the flood of alerts and security events, they are interested in looking into the history of where it originated from and where a particular security incident was spread. They are having trouble finding qualified security experts who would have relevantly assessed each incident and what's more internal employees may pose a greater security threat than outside hackers.

Prioritization of alerts

Although virtually all security systems try to assign a "criticality level" to each alarm, they are in fact unable to distinguish between attacks that are just an attempt at penetration and attacks that have actually been successful. Then the security analysts don't know where to start and which alarms really require attention, as most of them are false positive. This leads to unnecessary overloading of alarms and the inability to properly assess which ones are priority. The alarms generated by older infrastructures do not contain enough context to allow security analysts to build a comprehensive knowledge of the attack from a flood of information from isolated products, each reporting a different aspect of the attack.

The possibility of a retrospective view is missing

Non-selective retrospective visibility in one unified system is invaluable. It mimics the functioning of the human brain - before making a decision, a person considers what he or she knows based on his or her previous experience. Modern systems should work the same way, helping analysts to effectively detect certain types of threats that can be only detected by a certain pattern that is created

after some time. Security analysts need a context and an overview of the entire life cycle of the threat, where they are able to find a patient zero, clear endpoints, create forensic analysis, and analyze attack vector.

IT addiction and missing security analysts

Although security systems can assign a criticality level to each alarm, they cannot really distinguish between intrusion attempts and real attacks. Generated alarms do not contain enough context and do not allow a large number of alarms from isolated systems to build a complete attack image. All security teams are dependent on other IT teams to correctly collect information and events to evaluate properly. In addition, it is nothing unusual to see security analysts surrounded by eight to twelve monitors, between which they copy and transmit information from one system to another, which may represent a potential weakness at the point of attack.

Insufficient security awareness

They are not always just hackers who can pose a danger to the company. "Evil" often drills even inside the company itself in the form of employees who often do not even know that they have caused anything wrong. An employee to whom an attack has occurred does not have to have unfair intentions. It is enough if it is insufficiently educated in terms of security and, for example, opens a file, or inserts a USB key that it found on the street, and the problem is in the world.

Spams

IBM X-Force reported that the coronavirus spam emails were disguised as official notifications sent by a disability welfare provider and public health centers. The email content warns recipients about the rapid spread of the virus, and instructs them to download an attached notice that allegedly contains preventive measures. This spam emails had word document attachments. The text in the document contained instructions to click on the enable content button to be able to view the document. Clicking on the button installs the Emotet payload using a PowerShell command [2].

3 ADVANCED THREATS ARE IMPROVING EVERY MOMENT

According to the latest Forrester agency report, 53 % of business security leaders said that the biggest challenge their companies face is the rapid progress of cyber threats [3].

Cases, when it is not clear how intrusions come in, seems to appear more and more in Slovakia too.

Small businesses usually have a low level of security compared to larger businesses, especially due to high cost and complexity. However, cyber threats

are increasingly affecting smaller organizations. According to Verizon's security attack analysis [4], smaller businesses represent up to 58 % of data theft and over 48 % of data breaches in 2018 were due to direct attacks against Web applications. Web applications are an attractive target for hackers, as they are often public applications that require openness to the Internet to stimulate e-commerce and business. Back-end databases connected to web applications are the main repository for sensitive information such as credit card numbers, personally identifiable information (PII) and confidential information covered by intellectual property.

Firewall, VPN, antivirus, IPS, antispam, web filtering etc. - these traditional, stand-alone security solutions that any more advanced company should have, work in an isolated environment with a separate management interface without a meaningful way to gather or share threat information with other devices in network. They were not designed to protect today's distributed architectures across all attack capabilities. Insufficient integration slows the response to threats. Working with these isolated and inconsistent security systems is overloading SOC's.

How to make security teams more efficient?

In order to avoid detection and to infect systems, modern attacks use new distribution channels, such as social engineering, and deploy new techniques such as multi-tier and file-less attacks. Disruption can occur in minutes or seconds. To effectively detect and defend against these attacks, it is necessary to coordinate, automate security and ensure automatic response to potential threats so that attacks and incidents can be identified and isolated in real time. Companies invest in various non-conceptual systems. They lack automation to maximize resource utilization and the ability to quickly detect and respond to today's cyber threats. Security officers need a unified system designed solely for detection and response to provide them with:

- Deep visibility of network events and endpoints in real time.
- The depth visibility of events on the network and endpoints retrospectively.
- Various coordinated threat detection techniques at each stage of the attack (Cyber Kill Chain).
- Automatic correlation and evaluation of network-based threats that actually attacked endpoints.
- Grouping of redundant and related security alarms into a conclusion that can be further processed as a single alarm.
- All information, including context, tools needed to investigate and block an attack in a single screen.

What will ADR bring you?

In order to stand up in times of cyber attacks, Automated Detection and Response (ADR) systems

need to be deployed as a major security tool for security teams. Quickly identify suspicious actions, validate threats by a number of criteria, automate corrective action, and step-by-step analysis or proactive threat detection will bring you lower response costs, human resources, less threat to company reputation, less and shorter business disruption, tangible results after a few days and increasing the efficiency of SOC.

Deep and wide visibility

The ADR system must provide us with independent visibility of what is happening in networks, but also at endpoints - in both virtual and physical environments. Deep visibility means keeping track of packets that pass through the network, network connections and content, including content hidden deep in files that is not visible in packets. Broad visibility means having an overview of all network protocols and ports, including non-standard and unknown proxy passes, both inbound and outbound. For endpoints, the term "depth" means having an overview of processes, memory, storage media, file manipulation, network activity, registry changes.

Real-time visibility and retrospective

In order to detect security incidents effectively, it is necessary to provide visibility not only of what is happening now but also of what has happened in the past. The system must non-selectively retain detailed information describing what happened on the network, endpoints, regardless of whether it was considered suspicious or not. This will allow security teams to go back in time to detect threats based on a behavioral pattern and take advantage of sequence analysis.

Correlation, validation and consolidation

Effective action by security teams requires a single platform to provide information and acceleration. However, providing information does not mean giving as many alarms as possible and hoping that the correct information will be hidden and revealed somewhere. Providing information means maximally filtering out noise, useless information and false alarms while combining content, context, telemetry and valid findings.

Effective threat detection

In many cases, companies do not even know that they are victims of cyber attacks. Detecting past and future attacks, detecting and preventing all phases of the attack cycle, or continuously mapping the network with its activities are key elements for effective threat detection. It can be done by automatically analyzing or comparing threats with new signatures and rules.

Identification of "patient zero"

By identifying the first "victim" along with the size and extent of the attack, you will be able to detect the depth and severity of the incidents and start taking corrective action immediately.

Ability to determine what happened

Confirming that the attack had a real impact on society is the most important step before the reaction. Confirming detected attacks is time-consuming and knowledge-intensive. Usually, the cooperation of several organizational units within the company is necessary. For the security teams to be effective, it is necessary to automate validation steps and automatically monitor and evaluate attacks, suspicious and malicious behavior from detection points to their manifestations on endpoint systems.

Immediate reaction

Be able to respond to the incident yourself. This can reduce the investigation time from the original few weeks to a few hours. Automatic validation, correlation, and threat data collection make it possible

to respond significantly faster. This will increase the effectiveness of security specialists through immediate response.

4 PREDICTION IN THE IT WORLD

The estimation of developments in the IT world is very fast. It is hard to guess what to expect and what to prepare for. What do the closest forecasts look like?

Fast market growth in the cloud space

With the introduction of cloud computing infrastructure, a rapid growth in the cloud security market is expected. IDC Research predicts that global public cloud services spending reached \$ 210 billion in 2019 and that by 2020 more than 90 % of businesses will use multiple cloud services and platforms. Software as a Service (SaaS) is the largest category of cloud computing, capturing more than half of all public cloud spending in 2019. Infrastructure as a Service (IaaS) is the second largest category of public cloud spending in 2019, followed by Platform as a Service (PaaS) [5].

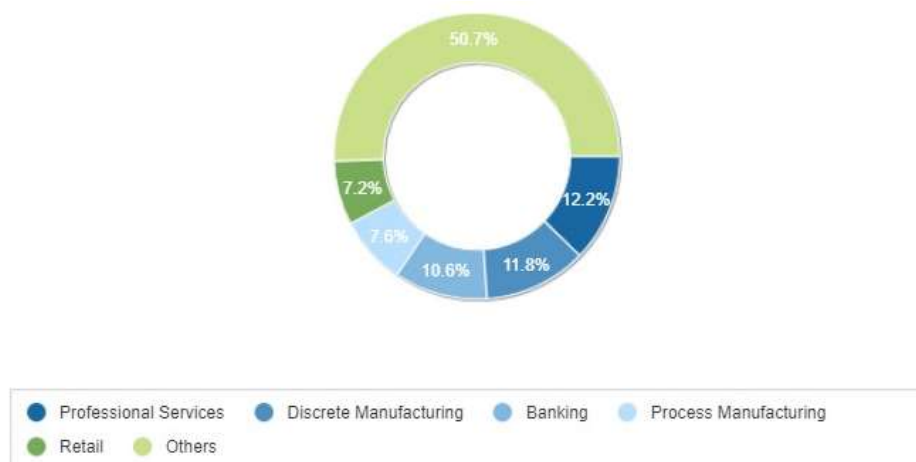


Fig. 1 The top industry based on market share
Source: authors.

Increased traffic inside networks

As local environments continue to evolve and become private clouds, network traffic flows will gradually shift from traditional north-south to east-west. By 2021, up to 86 % of total traffic from data centers will be east-west instead of north-south. Proper control of all east - west operations to ensure proper load segmentation will easily expand the current market for data center security in the north - south direction by up to four times.

The boom of IoT devices

IoT devices are expected to increase. With its steep rise in corporate networks, the answer to what

is on the network has never been more difficult or important. IoT devices are naturally insecure, so their identification and segmentation are required to prevent east-west infection.

The rapidly evolving threat environment penetrates vulnerable endpoints

According to Gartner, by 2021, the trend will continue that 99 % of exploited vulnerabilities have been known to security and IT professionals for at least 1 year. Maintaining endpoint hygiene is becoming increasingly demanding. As a result of the separation of functions, network security teams usually do not have much control over "patching" and

have little control over the freedom of action in end use. At the same time, however, IT has to deal with leaks from broken endpoints [6].

Lack of qualified cyber security personnel

Organizations are exposed to the growing need for experienced cyber security personnel in the future, but are faced with an ever-decreasing global supply of resources. Many organizations either are seeking to outsource NOCs and SOCs, or are looking for better solutions that can keep pace with a constantly evolving threat environment.

In 2019, damage caused by ransomware will increase

Global damage from ransomware was forecast to exceed \$ 5 billion in 2017 - compare to year 2015 it was only \$ 325 million. This is a 15-fold increase in just two years and the future will be even worse. Damage is expected to rise to \$ 11.5 billion by 2019, with one enterprise expected to fall victim to ransomware every 14 seconds [7].

Myths and facts

Myth:

Community, unsupported versions of technology or products are free.

The fact:

The result is significantly more costly to implement than technology from a trusted or reliable partner, where in the pre-production stage everything is ready for deployment, including available technical and educational support from the partner.

A reason:

- Community versions are changing fast.
- The need to frequently operate and update the environment.
- Lack of expertise.
- More work with content.

For reflection:

It is worth considering how the platform under consideration is compatible with other technologies. Choice and possible changes in the tools used and the benefit of everything that is available today and in the future is essential

Surveys Say Organizational Inertia Weakens Cyber Security Defenses

Inertia, by definition, indicates resistance to speed, direction or motion. It can creep in over a period of time and become an established behavior in an organization. There were found specific examples of cyber security inertia which, if not addressed, could hinder an organization's ability to detect and contain threats that break through the perimeter [8].

A key finding of published reports is that nearly half (46 percent) of organizations rarely make substantial changes to security strategy—even after being hit by a cyber attack. This represents a failure to learn from past incidents that puts sensitive data, infrastructure and assets at risk; a consequence that respondents recognized, with the same proportion—46 percent—saying their organization can't always prevent attackers from breaching internal networks.

Another worrying discovery is that more than a third (36 percent) of organizations store usernames and passwords for privileged user accounts in word or excel documents on company PCs. These privileged credentials deliver fast-track access to networks and systems across the enterprise, making them a tempting target for attackers.

Speaking of raising the odds of an attack succeeding, it is surprise to find a growing number of organizations grant users administrative rights on their endpoint devices. It was found that, on average, 87 percent of users are allowed these rights. With advanced malware attacks over the past year, such as WannaCry and NotPetya, greater prioritization around blocking credential theft is necessary to prevent attackers from gaining access to the network and initiating lateral movement.

Removing inertia requires businesses to build and sustain a pervasive culture of cyber security that is driven by executives and the board. This should be a top-down initiative supported by clearly defined and communicated security strategies and actively executed with participation by employees company-wide.

5 MITIGATING CLOUD VULNERABILITIES

The Organizations must consider cyber risks to cloud resources, just as they would in an on-premises environment – including how they approach privileged access management. This message reinforces our long-held view that no matter where they “live” – on-premises or in the cloud – privileged accounts must be protected.

A cloud adoption introduces a host of new risks that must be understood and addressed. It places strong emphasis on the shared responsibility between organizations and cloud providers in protecting applications, data and other sensitive information in the cloud.

The shared responsibility model illustrates that, while cloud providers are responsible for the cloud infrastructure, organizations are still accountable for the security of certain services and sensitive data stored in public clouds, such as configuration, applications, data and environments.

The cloud vulnerabilities were categorized into four main groups– misconfiguration, process access control, shared tenancy vulnerabilities and supply chain vulnerabilities. These groupings take into

account both how often these vulnerabilities occur and how sophisticated a cyber attacker has to be to take advantage of them.

- *Misconfiguration*: one of the most prevalent vulnerabilities in cloud environments today, misconfigurations offer attackers the path of least resistance and so require very little in terms of sophistication from the attacker. These misconfigurations often arise from either policy mistakes or misunderstanding of the security responsibilities on the organizations side. These misconfigurations can result in several issues from denial of service to account compromise. If an attacker can abuse a misconfiguration to compromise a single privileged user, for example, they will use these credentials to compromise a cloud management console or – worse — ultimately take over control of the organizations cloud environment.
- *Poor Access Control*: these attacks almost always involve privileged access. The prevalence of this attack is widespread and it requires only moderate sophistication from the attacker. They look for opportunities to exploit weak authentication and authorization methods. Once they gain a foothold, attackers will start to escalate privileges, move laterally through the environment and, ultimately, compromise as many cloud resources as possible. For example, an attacker can bypass multi-factor authentication (MFA) by evoking a password reset where only single factor authentication is required to reset credentials.
- *Shared Tenancy Vulnerabilities*: adversaries who are able to determine which software and hardware components are used in a public cloud hypervisor could take advantage of vulnerabilities to elevate privileges in the cloud. The reports note that while there have been no reported compromises in any major cloud computing platform, security researchers have demonstrated both hypervisor and container breakouts. A recent example of this was CVE-2019-1372, where the attacker could remotely execute code, bypassing the sandbox, and CVE-2019-1234 where attackers could make requests to the internal Azure Stack resources. Both examples here have since been addressed.
- *Supply Chain Vulnerabilities*: supply chain vulnerabilities in the cloud include attackers inside the supply chain and backdoors intentionally installed in hardware and software. While infiltrating the supply chain is not usually an attacker's ultimate goal, if the attacker can get the cloud provider to install hardware with a backdoor, it makes all other controls useless.

However, as it was noted previously, cyber attackers will almost always seek the path of least resistance to carry out their mission. That usually involves abusing misconfigurations or privileged

access instead of turning to highly sophisticated methods such as inserting an agent into the cloud supply chain.

The securing cloud instances is an ongoing challenge. The guidance offered by a security organizations is helping to demystify some of this and – maybe even more importantly – provide prioritization to the most susceptible areas so organizations know where to optimize their security resources [9].

Data Privacy Day: Data Protection Lessons from the 2010s

Data protection standards have come a long way since 1981, especially in the last couple of years with GDPR and CCPA – two regulations that extend the rights of individuals to better control and protect the use of their personal data in the evolving digital landscape. It's generally believed that GDPR and CCPA are laying the foundation for further groundbreaking regulations.

And it makes sense. According to Business Insider, “of the 15 largest data breaches in history, 10 took place in the past decade.” These breaches collectively resulted in the loss of nearly 4 billion records [10].

- *2016: Uber Breach* – while it was disclosed in 2017, Uber suffered a breach in 2016 that exposed personal information belonging to 57 million drivers and customers. Attackers stole names, email addresses and phone numbers and demanded a \$100,000 ransom. To add insult to injury, Uber also was fined nearly \$150 million for not disclosing the breach earlier. Lesson learned - don't store code in a publicly accessible database. Uber data was exposed because the AWS access keys were embedded in code that was stored in an enterprise code repository by a third-party contractor.
- *2017: Equifax Breach* – several tech failures in tandem – including a misconfigured device scanning encrypted traffic and an automatic scan that failed to identify a vulnerable version of Apache Struts – ultimately led to a breach that impacted 145 million customers in the US and 10 million UK citizens. Lesson learned - get security basics right. Despite cyber attacks becoming more targeted and damaging, organizations are frequently still ignoring the security basics. Patches need to be applied promptly and security certificates need to be maintained. This breach also inspired elected officials to push for legislation to tighten regulations on what protections are required for consumer data and influenced an increase in executive accountability.
- *2018: facebook's Cambridge Analytica Breach* – Cambridge Analytica, a British political consulting firm, harvested the personal data from millions of peoples' Facebook profiles without

their consent and used it for political advertising purposes. The scandal finally erupted in March 2018 when a whistle blower brought this to light and Facebook was fined £500,000 (US\$663,000), which was the maximum fine allowed at the time of the breach.

Lesson learned - Protect user data (or pay up). Lawmakers claim Facebook “contravened the law by failing to safeguard people’s information” – and suffered the consequences. Now there is placing additional pressure on Facebook to stop the spread of fake news, foreign interference in elections and hate speech (or risk additional, larger fines).

- 2019: *Ecuador Breach* – Data on approximately 17 million Ecuadorian citizens was exposed due to a vulnerability on an unsecured AWS Elasticsearch server where Ecuador stores some of its data. While the sheer scale of this breach made it headlines news, the breadth of exposed information really made everyone stand up and take notice. Exposed files included official government ID numbers, phone numbers, family records, marriage dates, education histories and work records. In addition, a similar Elastic search server exposed the voter records of approximately 14.3 million people in Chile, around 80% of its population.

Lesson learned - Adhere to the shared responsibility model. Most cloud providers operate under a shared responsibility model, where the provider handles security up to a point and, beyond that, it becomes the responsibility of the customer. As more and more government agencies look to the cloud to help them become more agile and better serve their citizens, it’s vital they continue to evolve their cloud security strategies to proactively protect against emerging threats – and reinforce trust among the citizens who rely on their services.

These incidents are just a small sample of the numerous data breaches that occurred in the 2010s. Any organization that collects or stores customer information can learn from these incidents and the many more like them. Not prioritizing data protection or simply doing the bare minimum can lead to regulatory non-compliance fines, or worse – the destruction of customer confidence and brand damage. Listening to the lessons of the past can help us prepare for a more secure future.

How Would You Detect and Impede Ransomware on an Endpoint?

Malware that encrypts and holds files at ransom can have devastating effects on personal documents, customer data and business operations.

One of the ways of detecting the *Presence of Ransomware* should consider how the behavior of file-encrypting ransomware might be different from

the operation of normal programs. Ransomware probably reads and writes more files than typical processes on the system. In addition, the encrypted files it creates have patterns that are different from the majority of non-encrypted files. That means that anti-ransomware tool should:

- flag processes that read or write too many files too quickly, define a threshold of normal file access activities, and react when it is exceeded.
- flag processes that change files’ entropy values, encrypted files tend to have a more uniform distribution of byte values than other files. Their contents are more uniform. The tool could compare the file’s entropy before and after the change. If a modified file has higher entropy, it might have gotten encrypted, indicating that the responsible process might be ransomware. However, the entropy of encrypted files is similar to that of compressed files; to keep down the rate of false positives.

When a tool spots the likely presence of ransomware, it could react to suspected ransomware by alerting the user and then, if configured to do so, kill the offending process. Terminating the process could be risky, though, if ransomware operates by injecting its code into legitimate processes. This highlights the importance of coupling anti-ransomware functionality with techniques that look for other malicious behavior, such as code injection, to spot malware even before it exhibits ransomware characteristics.

If terminating the suspicious process (or thread) is impractical or undesirable, a tool could interfere with the potentially-malicious code by slowing down its interactions with the file system. This would give the endpoint’s user time to decide whether to terminate suspected ransomware, turn off the system or respond in some other way.

Another way of impeding ransomware was proposed on the Free Forensics blog. This method involves setting up infinitely-recursive directories by taking advantage of certain features of the Windows file system. Ransomware that attempts to traverse the file system to locate the files it needs to encrypt will get stuck in such a sinkhole, giving the victim the opportunity to react to the infection [11].

Anti-Ransomware in the Real World

Any methods for detecting and impeding ransomware cannot be foolproof, as is the case with any anti-malware technology. For instance, ransomware authors could adjust their code to stay under the threshold anomalous file access. Many aspects of information security have such action-reaction dynamics, which might be the reasons why the majority of anti-malware vendors rarely reveal details about their inner-workings.

While the techniques outlined above might sound reasonable and achievable in the context of a single

endpoint, creating a product that can be deployed on many personal or enterprise systems drastically increases the endeavor's complexity. For example, the tool would need to have a practical way of dealing with false alarms where legitimate processes exhibit ransomware-like behavior or compressed files resemble encrypted ones.

The expectation is that commercial anti-malware vendors are creating or have already developed more sophisticated methods for dealing with ransomware, and have techniques that are more elaborate than what I described in this article. Those who remember the early days of spyware might recall standalone anti-spyware tools that were later merged into mainstream antivirus products. Similarly, anti-ransomware capabilities are becoming an essential feature of modern Internet security suites and anti-malware products.

6 CYBER SECURITY IN ARMED FORCES ACADEMY ENVIRONMENT

Educational institutions depend on computer networks and technologies to provide their students with university news, activities, emails, courses, academic year calendar, academic staff, student's marks and other personal information stored on their computer systems. Therefore, these systems need to be protected against a number of threats. It could be used by an adversary not only to affect the organization's assets by stealing their sensitive information, but to also affect the organization's financial side. Therefore, every organization using information systems must take information security seriously. The last decade has virtually perceived a dramatic increase in the number of cyber-security bugs and cybercrime incidents as well.

Information security awareness is particularly important for sensitive academic information which must at all-time be protected from cyber-attacks. This section briefly presents real examples of cyber security, as well as information security awareness and training, within the educational sector.

The period from 01.11.2019 to 29.02.2020 was chosen for the study and analysis of cyber security in the AFA environment. The reference databases and resources of the manufacturer of the implemented security technology were used for the subject of evaluation. The CTA (Cyber Threat Alliance) outputs were used as an additional source as well. The picture Fig.2 shows the overview of global application usage. The largest bandwidth is used by ssl communication, which is encrypted. The risk scale is given by the manufacturer's technology and is calculated as the sum of risk calculation weights. Each communication characteristic is assigned a factor. Picture Fig. 3 shows risk calculation weights and picture Fig. 4 shows risk assignment.

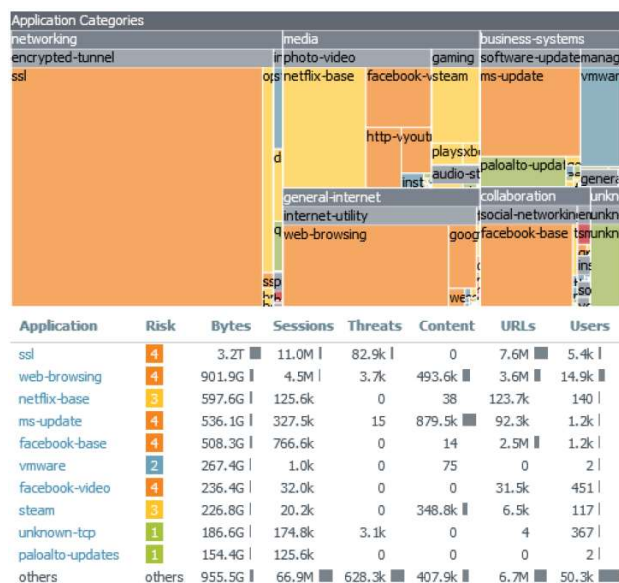


Fig. 2 The overview of global application usage
Source: authors.

Characteristic	Factor
Evasive	3
Excessive Bandwidth Use	1
Used by Malware	4
Capable of File Transfer	3
Known Vulnerabilities	3
Tunnels Other Apps	2
Prone to Misuse	2
Pervasive	1
Total	19

Fig. 3 Risk calculation weights
Source: authors.

Risk	Range
1	0 – 3
2	4 – 6
3	7 – 9
4	10 – 13
5	14 +

Fig. 4 Risk assignment
Source: author.

The picture Fig. 5 shows threat activity in selected period of time. As it is shown, the most common threats are spyware and vulnerabilities. The picture Fig. 6 shows application using non-standard port, where worth mentioning that more than 172,6 G bytes was transferred in 2.0 k sessions with web-browsing in the port 5000

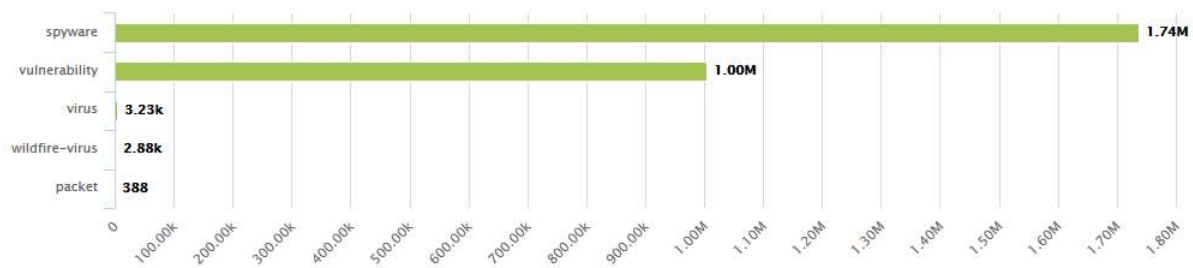


Fig. 5 The threat activity in selected period of time
Source: authors.

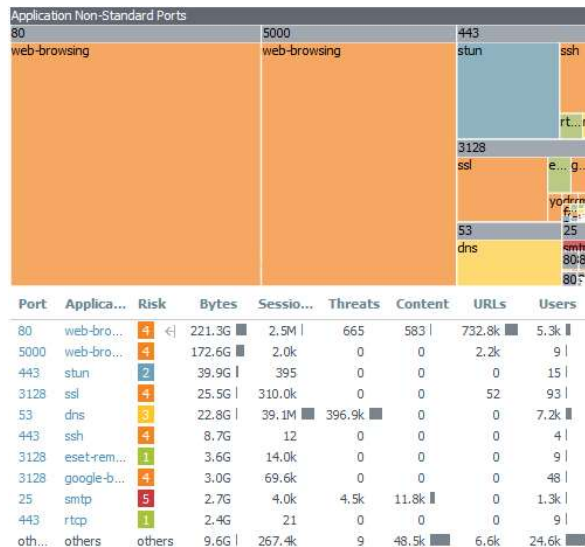


Fig. 6 The application using non-standard port
Source: authors.

The FW logs show, that the threat in category „dns-security“ occurs very often with very high risk severity. The threat „DNS Tunneling Domain“ occurrence will mean paying more attention to this type of communication. According to its definition, it is prone to a command injection vulnerability while parsing certain crafted HTTP requests. The vulnerability is due to the lack of proper checks on HTTP requests, leading to an exploitable command injection vulnerability. An attacker could exploit the vulnerability by sending a crafted HTTP request. A successful attack could lead to remote code execution with the privileges of the server [20].

DNS tunneling exploits the DNS protocol to tunnel malware and other data through a client-server model. This tunnel can be used to exfiltrate data or for other malicious purposes. Because there is no direct connection between the attacker and victim, it is more difficult to trace the attacker's computer.

An important security threat to this environment is also e-mail communication. Pictures Fig. 7-10 show executive summary of emails flow.

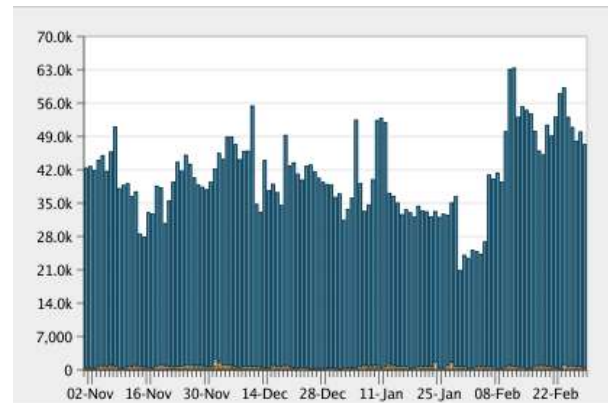


Fig. 7 The overview – Incoming mail
Source: authors.

Message Category	%	Messages
Stopped by Reputation Filtering	98.3%	4.8M
Stopped as Invalid Recipients	0.2%	8,758
Spam Detected	0.5%	22.9k
Virus Detected	0.0%	49
Detected by Advanced Malware Protection	0.0%	0
Messages with Malicious URLs	0.0%	645
Stopped by Content Filter	0.0%	16
Stopped by DMARC	0.0%	0
S/MIME Verification/Decryption Failed	0.0%	0
Total Threat Messages:	98.9%	4.9M
Marketing Messages	0.1%	4,437
Social Networking Messages	0.0%	1,172
Bulk Messages	0.3%	12.9k
Total Graymails:	0.4%	18.6k
S/MIME Verification/Decryption Successful	0.0%	0
Clean Messages	0.7%	35.0k
Total Attempted Messages:		4.9M

Fig. 8 The overview – Incoming mail summary
Source: authors.

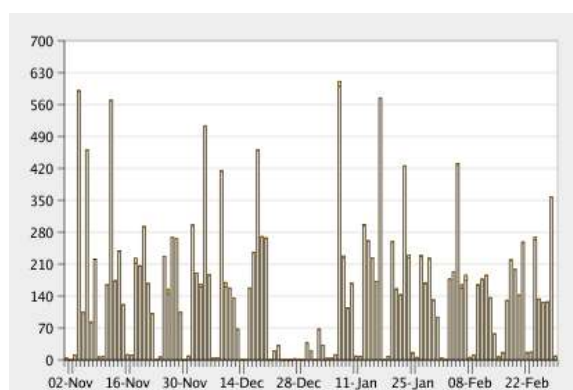


Fig. 9 The overview – Outgoing mail
Source: authors.

Overview > Outgoing Mail Summary		
Message Processing	%	Messages
Spam Detected	0.6%	100
Virus Detected	0.0%	0
Detected by Advanced Malware Protection	0.0%	0
Messages with Malicious URLs	0.0%	1
Stopped by Content Filter	0.0%	0
Clean Messages	99.4%	17.3k
Total Messages Processed:		17.4k
Message Delivery	%	Messages
Hard Bounces	1.8%	316
Delivered	98.2%	17.1k
Total Messages Delivered:		17.4k

Fig. 10 The overview – Outgoing mail summary
Source: authors.

Planned countermeasures of the AFA

The intent of the AFA, as one of the measures, is planning to use traffic decryption technologies for the purpose of full visibility and threat inspection. The best practice security policy dictates to decrypt all traffic except sensitive categories, which include health, finance, government, and traffic that is not decrypted for business, legal, or regulatory reasons, using decryption exceptions only where required. To ensure that certificates presented during SSL decryption are valid, the firewall will be configured to perform CRL/OCSP (certificate revocation) checks.

The best practice decryption policy rules include a strict decryption profile to attach to the organization's decryption policy rules, creation of the separate decryption policies and a separate decryption profile for sites that use TLSv1.1 so that only the sites that are legitimately needed for business purposes can access the network using TLSv1.1. The same is true about the SHA1 authentication algorithm—only for a few sites that are needed for business purposes the SHA1 will be used, separate decryption policies and a separate decryption profile will be created for them.

For traffic that won't be decrypted, the no decryption settings will be configured to block

encrypted sessions to sites with expired certificates or untrusted issuers.

Already fulfilled measure from the point of view of increasing email security is the deployment of other control mechanisms in the form of mail gateways.

The Academic institutions are also vulnerable to cyber-attacks; such examples include theft of private and sensitive information for students and faculty staff. Thus, in addition to a proper awareness programs, training, education and policies, the educational institutions must include all the safety measures such as, security, privacy, trust, identity management, audit and digital forensics to satisfy the legal and social requirements.

Awareness against cyber security incidents and their consequences should not to be ignored because it is important to all information security aspects and also their outcomes could affect academic institutions sensitive personal information as well.

7 CONCLUSION

Although a considerable amount of research effort has gone into malware analysis and detection, malicious code still remains an important threat on the Internet today. Unfortunately, the existing malware detection techniques have serious shortcomings as they are based on ineffective detection models. Signature-based techniques that are commonly used by anti-virus software can easily be bypassed using obfuscation or polymorphism, and system call-based approaches can often be evaded by system call reordering attacks. Furthermore, detection techniques that rely on dynamic analysis are often strong, but too slow and hence, inefficient to be used as real-time detectors on end user machines.

Based on the cases, the cyber crime is difficult to trace, and convict. Although countless resource has been spent on cyber security, the outcome is unpromising. The problem with current cyber security is that it is so passive – it is impossible to win this war only by defending. Fighting crime should be a common goal for all countries around the world. The collaboration is essential to mitigating current and future risks.

There can be defined “The five pillars of cybersecurity readiness”:

1. Education and Awareness.
2. Planning and Preparation.
3. Detection and Recovery.
4. Sharing and Collaboration.
5. Ethics and Certification.

Malicious software remains a major threat to today's information systems. Detecting and analyzing dangerous programs is a costly and often inaccurate endeavor. The difficulty of this task is underscored by a recent contest challenging participants to figure out

the nefarious behavior of a particular program that has already been determined to be malicious in nature [21]. Often identifying a program (or portion thereof) as malicious is half the battle.

Apart from providing the necessary protective and security mechanisms, a good awareness program must undergo proper designing to provide compulsory awareness and active training and education programs for users and employees as well. The program should be instrumental in developing and spreading security awareness between them, employing proper physical access controls, obeying the security policies and rules as laid down by the organization to achieve the best security.

The security of information systems is becoming a leading priority nowadays ever more, where the number of cyber security incidents rapidly rises and becomes more and more effective and aggressive than before. For that, each organization using information systems must take information security seriously as their top priority.

References

- [1] Available at: <https://www.ey.com/en_gl/giss>
- [2] Available at: <<https://www.trendmicro.com/vinfo/us/security/news/cybercrime-and-digital-threats/emotet-uses-coronavirus-scare-in-latest-campaign-targets-japan>>
- [3] Available at: <<https://go.forrester.com/blogs/predictions-2019-cybersecurity/>>
- [4] Available at: <<https://enterprise.verizon.com/resources/reports/dbir/>>
- [5] Available at: <<https://www.idc.com/getdoc.jsp?containerId=prUS44891519>>
- [6] Available at: <<https://www.gartner.com/smarterwithgartner/focus-on-the-biggest-security-threats-not-the-most-publicized/>>
- [7] Available at: <[https://www.darkreading.com/attacks-breaches/ransomware-damage-hit-\\$115b-in-2019/d/d-id/1337103](https://www.darkreading.com/attacks-breaches/ransomware-damage-hit-$115b-in-2019/d/d-id/1337103)>
- [8] Available at: <<https://www.cyberark.com/blog/survey-says-organizational-inertia-weakens-cyber-security-defenses/>>
- [9] Available at: <<https://www.cyberark.com/blog/nsa-offers-guidance-for-improving-cloud-security/>>
- [10] Available at: <<https://www.businessinsider.com/biggest-hacks-2010s-facebook-equifax-adobe-marriott-2019-10>>
- [11] Available at: <<https://zeltser.com/detect-impede-ransomware/>>
- [12] Available at: <<https://cybersecurityventures.com/cybersecurity-almanac-2019/>>
- [13] Phishing Attacks: A Guide for it Pross. TechRepublic.
- [14] YUE BA: *Understanding Cybercrime and developing a monitoring device*, Turku University of Applied Sciences, Cybersecurity Threats Challenges Opportunities.
- [15] HARPER, A., NESS, J., LENKEY, G., HARRIS, S., EAGLE, CH., WILLIAMS, T.: *Gray Hat Hacking – The Ethical Hacker's Handbook*, third edition.
- [16] KOLBITSCH, C., COMPARETTI, P., MILANI, KRUEGEL, CH., KIRDA, E., ZHOU, X. and WANG, X.: Effective and Efficient Malware Detection at the End Host. In: *18 th Usenix Security Symposium*. pp. 351-366.
- [17] MIRZA, M. B., ARSLAN, M., BOKHARI, SEYDA T. F., ZAFAR, R.: Malicious Software Detection, Protection & Recovery Methods: A Survey. In: *BRIS Journal of ADV S&T*, Vol. 2(5). pp. 14-23. ISSN 0971-1563.
- [18] Available at: <<https://www.zerodayinitiative.com/>>
- [19] Available at: <<https://talosintelligence.com/zeus-trojan-analysis>>
- [20] Available at: <<https://kb.cert.org/vuls/id/498544/>>
- [21] WEBER, M., SCHMID, M., GEYER, D. & SCHATZ, M.: *A Toolkit for Detecting and Analyzing Malicious Software*, Cigital, Inc. 21351 Ridgeway Circle Dulles.
- [22] SAMAHER AL-JANABI, IBRAHIM ALSHOURBAJI.: A Study of Cyber Security Awareness in Educational Environment in the Middle East. In: *Journal of Information & Knowledge Management*. Vol. 15, No. 1, 2016.
- [23] Available at: <<https://knowledgebase.paloaltonetworks.com/KCSArticleDetail?id=kA10g000000Cm12CAC>>
- [24] Available at: <<https://knowledgebase.paloaltonetworks.com/KCSArticleDetail?id=kA10g000000CImQCAS>>

Dipl. Eng. Martin **DROPPA** (PhD. student)
Armed Forces Academy of General M. R. Štefánik
Demänová 393
031 01 Liptovský Mikuláš
Slovak Republic
E-mail: martin.droppa@aos.sk

Col.(ret.) Prof. Dipl. Eng. Marcel **HARAKAL**, PhD.
Armed Forces Academy of General M. R. Štefánik
Demänová 393
031 01 Liptovský Mikuláš
Slovak Republic
E-mail: marcel.harakal@aos.sk

Martin Droppa - was born in Liptovský Mikuláš, Slovakia in 1980. He received his Engineer degree in 2003 in Communication and Information systems from the Military Academy in Liptovský Mikuláš. He

is currently the head of the communication and information systems department at AFA. His research is aimed to computer networks, information systems, security in wireless communication.

Marcel Harakal' - He received the MSc. degree in electrical engineering from the Faculty of Electrical Engineering, Slovak Technical University in Bratislava in 1983. In 1997 he successfully finished his PhD. studies in artificial intelligence. From 1983 to 1989 he worked as a research engineer at the Military Research Institute in Liptovský Mikuláš. In 1989 he joined the Armed Forces and since then he

has worked in various teaching and managerial positions at the Department of Informatics. In 2002 he habilitated as an Associate Professor in the Electronics and telecommunication field. In 2018 he became a Professor in Military Communication and Information systems. During his university career from 2004 to 2012 he led the Department of Informatics. Currently he is in the position of Vice Rector for Science of the Armed Forces Academy of General Milan Rastislav Štefánik, Liptovský Mikuláš, Slovakia.

His research interests include computer engineering, image processing, cyber security, and network operations.



ARMED FORCES ACADEMY OF GENERAL MILAN RASTISLAV ŠTEFÁNIK

Liptovský Mikuláš

Security and Defence Department

11th International scientific conference

„National and International Security 2020“

22nd – 23rd October 2020

Liptovský Ján

Aim of the Conference:

Analysis of the current state of development of national and international security and new aspects of its direction.

Topics:

- international relations and the formation of a global security environment (global threats and problems of humanity, asymmetric armed conflict, global problem solving by the world's largest players, participation of foreign and non-military actors in crisis management, observance and protection of human rights, development of democracy).
- national security as the object of research, formation and implementation of state security policy (state security and crisis management, internal state security, preparation of legislation and security strategic documents of the state, participation of the armed forces of the democratic state in ensuring internal security)
- current security science and other directions of its development the role and tasks of the armed forces in preventing and resolving conflicts (requirements for the capabilities of current and future forces, experience of the International Crisis Management)

Contact:

Chairman of the Organizing Committee
Assoc. Prof. Dipl. Eng. Ivan **MAJCHÚT**, PhD.
Tel.: 421 (0)960 422620
E-mail: ivan.majchut@aos.sk

Information about conference can be found on the web-site:
http://www.aos.sk/struktura/katedry/kbo/NMB2020/index_en.php

FUNDAMENTALS OF STATIC MALWARE ANALYSIS: PRINCIPLES, METHODS AND TOOLS

Andrej FEDÁK, Jozef ŠTULRAJTER

Abstract: Nowadays, the security of all systems connected to the public network is severely tested. Most users try to protect themselves against many abusive practices by using many security tools to keep their privacy safe. Information technology security involves many branches that address the prevention and protection against malicious software. One of those branches is the analysis of malicious files, specifically we will focus on the static analysis of malware. In static analysis, a suspicious sample is not executed and observed as in dynamic analysis, but many tools and methods are used to extract meaningful character strings from sample, data from the header of executable file format, information about the type of compression, the type of compiler used to create the file, and last but not least the application code. This work provides an initial insight into the complex subject of static analysis.

Keywords: Forensic analysis; Static analysis; Malware; Portable executable; String; PE header; Extractor; Obfuscation; Compression.

1 INTRODUCTION

The aim of this work is to describe the basic tools and methods used in the analysis of malware. Malware analysis is a large part of Information Technology (abbr. IT) security that is aimed at preventing the spread of malicious software. It analyses individual components of malware as well as the behaviour of malware in the infected computer. The main task of the analysis is to find out what functionality a given malware has, i.e. what it does and can do under what conditions. It is able to prevent on-coming computer attacks by detecting the way how malicious code get in your computer. For this purpose, analysts are offered a large number of methods and tools to analyse samples of malicious software. An important part of the analysis is also obtaining a sample of specific malware, which is usually in the form of an executable file (in the Windows operating system we can talk about files in Portable Executable format). Finally, a report of complete analysis should contain all the relevant information about the malware that was collected during the analysis [4].

2 MALWARE ANALYSIS

Methods of analysing malware can be divided into two main branches - static and dynamic methods. While both types of analysis have the same goal of finding out how a given malware works, the tools, time, and experience needed to perform the analysis are different. The basic difference between these methods is that in a static analysis, a given sample is not executed, whereas in a dynamic analysis this is necessary. Detailed static analysis of the program involves the use of a disassembler to allow subsequent analysis of the internal logic of the software using the exposed code. Dynamic analysis executes malicious code in a controlled environment that closely monitors its behaviour. When performing analysis of malicious software, pieces of information from static and dynamic analysis complement each

other and help to get a complete picture of the malware.

Generally, static analysis is the analysis of computer software that is performed without the need of executing programs. This analysis describes the data structure of the program or the process of analysing the code. Thanks to this, it is possible to determine some functions of the analysed software. Some of the static analysis methods are considered to be the primary analysis of malware. Basic static analysis provides information on whether a file is considered to be harmful, processes data from the file header (e.g. date of its creation) or provides a list of strings used in the code (from libraries to Internet Protocol addresses), but of course only if those parts of the code are not obfuscated. This analytical method is very fast and straightforward and quickly helps us to get familiar with the basic functionality of the file [4].

After the basic analysis, the acquired base of knowledge can be further expand using methods and tools of advanced analysis. The most detailed, complex and time consuming method is the analysis of the program code itself. This method consists of decompilation the machine code of application into the lowest-level programming language - assembly language, or in some cases a higher programming language or pseudocode, but there is a major problem with the code reconstruction, because the high level of abstraction sometimes makes the code unintelligible. Because of their nature, languages using intermediate representation (Java, C#) allow decompilation to a much simpler form. There are also some simplifications for other languages (e.g. IDA with HexRays decompiler). This is followed by an analysis of the program code, estimation of its functionality and extraction of additional data from the program code. Even though the analyst has an executable program or any part of it, he does not have the source code, so he sees only what is going to be executed at the processor level, but not the high-level concepts that author of the code actually used [3], [4].

One of the main differences between static and dynamic analysis is that static analysis is somewhat safer than dynamic one because it does not directly execute malicious code. Therefore, we do not need to worry too much about becoming a victim of dangerous malware techniques. The risk of accidental execution of malware can be further reduced by using a virtual machine (VMware, VirtualBox, etc.), by analysing malware on an operating system for which it was not made or by increasing the level of User Account Control (confirmation is required to run the program). Another advantage of the static analysis is the possibility to detect potential functions of malicious software that may not be found during dynamic analysis. Although the static analysis is more thorough, it is also more time-consuming. Many methods used in static analysis increase the time required to analyse code. Nowadays, almost every malware is obfuscated, which means that parts of the program are replaced by another functionally equivalent parts that are encoded, compressed or intentionally extended with random or confusing code. Because of this, security teams do not use such detailed analysis when dealing with a large number of incidents. Limited capabilities, resources and time do not allow each incident to be resolved by slow methods and therefore security teams tend to use automatic, partially less informative methods. And even after a comprehensive analysis of the code, they may not be able to identify all the functions that the software could potentially perform (for example external communication with websites, servers or receiving encryption keys from the environment) [4], [5].

3 ONLINE ANTIVIRUS SCAN

The first step in analysing files is to make sure that sample is perceived as malicious code using available antivirus (abbr. AV) tools. Online multi-AV scanners provide a quick and clear picture of an unknown file that can be potentially dangerous for us. In many cases the use of these services is very easy because of the intuitive and user-friendly interface. Some online scanners allow their services to be used with their

own tools and scripts that allow the user to automate and speed up repetitive tasks.

Before we begin, the risks associated with using these services should be understood. False positives and false negatives will always be a problem. Even if 100 % of antivirus products indicate that a file is safe, that doesn't necessarily mean the file is safe. This can also be applied the other way around. In addition, if a private instance of the service does not start, files that have been uploaded to public websites may be automatically shared with other resellers and third parties. This is generally good because the vendors need samples to build new signatures. However, targeted malware may contain hard-coded usernames, passwords, domain names, or Internet Protocol addresses (abbreviated IP) of internal systems that should not be distributed to suppliers and possibly to the public. [1]

Probably the best known online tool for analysing malware is VirusTotal. This tool allows you to upload a dangerous file, check a suspicious Uniform Resource Locator (abbr. URL), search for an already uploaded file using a hash and so on. Then it can perform automatic forensic analysis on the uploaded file using more than 60 antivirus engines as shown in Fig. 1. The result of such scan are simple pieces of information quickly obtained by many methods of static and dynamic analysis. On the other hand, the disadvantage of this tool is its closed source code. Similar features are provided by other online scanners such as VirSCAN and Jotti [3], [6].

As previously mentioned, the use of multi-AV online service is quite simple. All you need to know is the URL of a specific online tool (eg www.virustotal.com, www.virscan.org or virusscan.jotti.org) and after opening the website, the suspicious file only needs to be dragged to the website or the full path to the file which will be recorded and scanned. These online scanners provide sophisticated scripts and custom applications for their faster use and automation of certain tasks. The script called `virt.py` created by Xiaokui Shu was used to illustrate the use of VirusTotal service. By modifying the registry in Windows, this auxiliary batch script has been added to the right-click context menu:

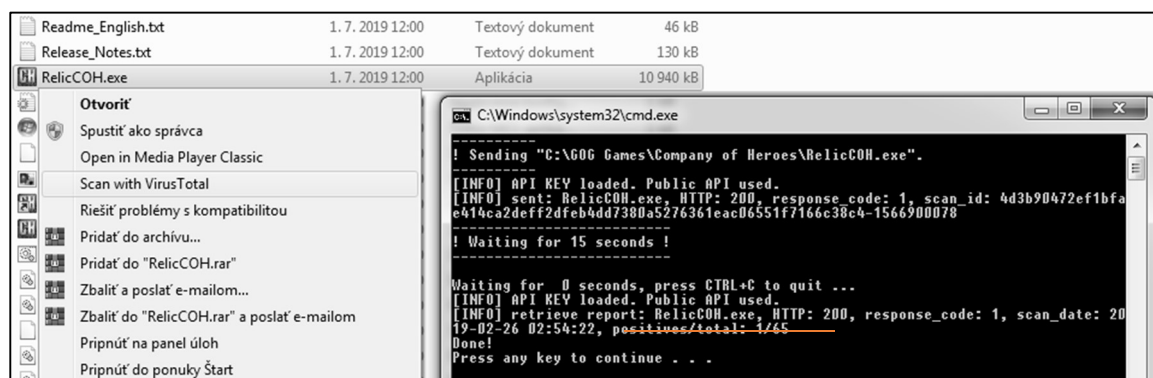


Fig. 1 Using the online VirusTotal service with the script
Source: [3].

```
REG ADD "HKEY_CLASSES_ROOT\*\shell\Scan
with VirusTotal"
REG ADD "HKEY_CLASSES_ROOT\*\shell\Scan
with VirusTotal\command" /t REG_SZ /d
"\""%CD%\script_check_file.bat" \"%*1\""
```

The code of the auxiliary batch script looks like this:

```
REM Enter the directory which contains our scripts
cd /d "%~dp0"
REM Execute the script with the parameter -s (send
file) and the input data
python.exe virt.py -s %1
REM Wait for the online scanner to process the file
timeout /T 15 /NOBREAK
REM Execute the script with the parameter -r
(retrieve report) and the input data
python.exe virt.py -r %1
```

A community that uses online multi-AV scanner services is raising its global level of IT security by sharing results of scanned malicious files and URLs. However, such openness to the community is also a major disadvantage of the online scanner, what makes it useless in some cases. Specifically, the biggest problem is the fact that all users may retrieve a report of any sample at any time. Authors usually modify their malware to have a unique hash fingerprint (no sample with that fingerprint has yet been analysed by VirusTotal). And when the analyst uploads the sample to VirusTotal, the author of malware immediately learns that his malware was found and is being analysed by a forensic analyst. Because of this, an attacker may change the behavioural strategy, turn off the sample and so on. Although the tool provides helpful features and integrates many analytical methods and tools, it is not advisable to use VirusTotal during an analysis conducted by a security team such as Computer Security Incident Response Team (abbr. CSIRT) [1], [3].

4 EXTRACTION OF STRINGS

Extraction of strings (a sequence of Unicode and ASCII characters - American Standard Code for Information Interchange) from the suspicious software is another method used by analysts when analysing malicious files. This extraction is probably the simplest method by which it is possible to reveal some features of the program. This method tries to find meaningful text strings in binary files that create a sequence of bytes with values in the range of printable characters ending with the byte of zero value. It is basically a trivial data mining from the binary files that can often be quite effective. It is a source of a huge number of artefacts, some of which may be crucial for forensic analysis. Such crucial artefacts include various strings such as IP addresses and URLs with which the malware is able to communicate, registry keys with values, commands that malware uses for communication over the

Internet (for example the Internet Relay Chat protocol is easily recognized by its text commands), file names and file paths that the malware works with, or decryption keys for the encrypted parts of the code. Although strings do not give a clear picture of the purpose and capability of a file, they can give a hint about what malware is capable of doing [4].

This approach will not work with encrypted strings and the output may additionally contain a significant amount of strings that do not represent any meaningful information. Malware authors often use tools and methods to prevent reverse engineering and encoding or compression to make the analysis and detection more complicated. A software without malicious code almost always contains a large number of strings, while compressed malware has only a few. Therefore we know that if we encounter a software containing a small number of strings, it is probably compressed and may contain a malicious code. Then the extraction of strings can be used again after the hidden part of the code is unpacked.

4.1 Tools Strings, HexDive or BinText

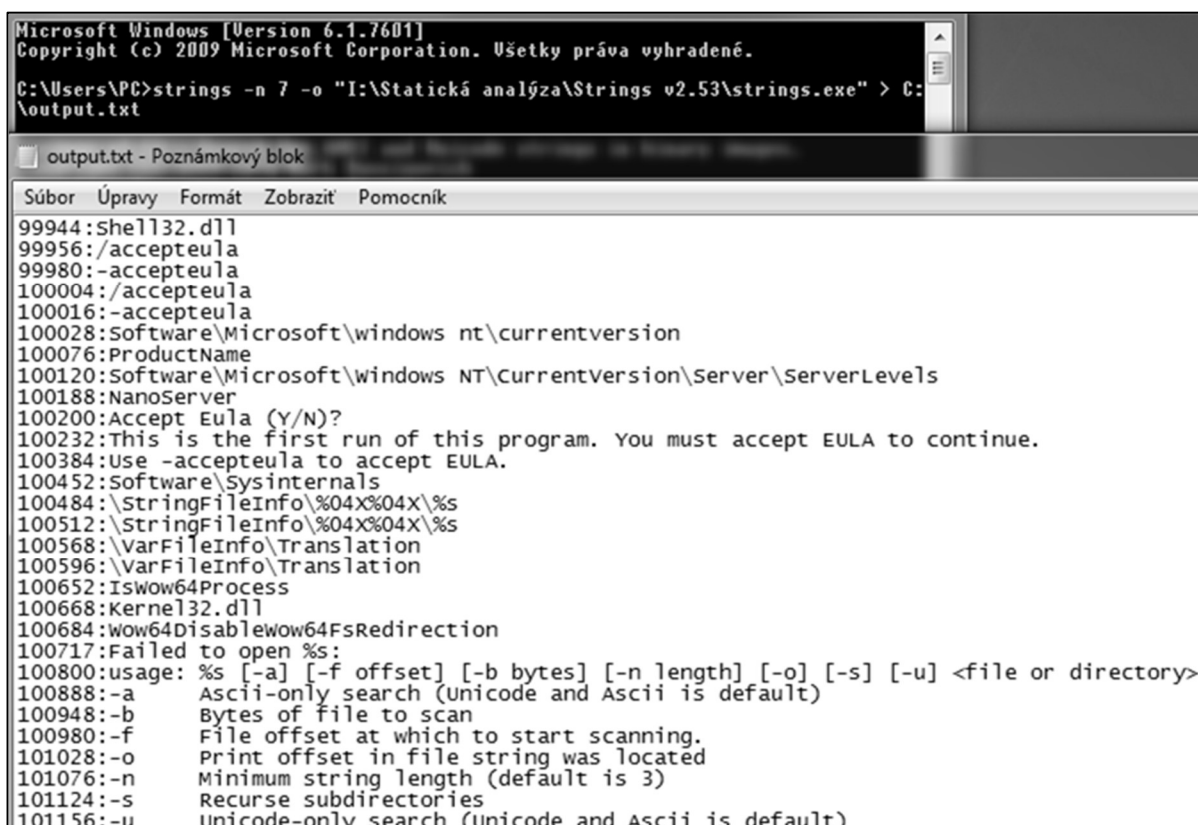
Specialized software such as Strings, HexDive or BinText can be used to search for strings stored in the program. All of these programs search for Unicode or ASCII characters and list all strings with a pre-set length. Strings from Windows Sysinternals is a basic tool that implements string extraction and its main advantage is a great compatibility. Once downloaded, it is a good idea to copy this tool to a directory which is included in the environment variable named Path (the content of the variable can be displayed by executing the command "set" or "echo %Path%") or add the path to the variable in order to run Strings from the command line [3], [7].

If you want to list strings of seven or more characters from a suspicious file, use the following command (Fig. 2):

```
strings -n 7 -o input_file > C:\output.txt .
```

HexDive is an intelligent extractor that speeds up the analysis of strings obtained from executable files. This is achieved by displaying only the relevant strings for malware analysis (its output is about two-thirds smaller than the output of Strings) [5], [8].

Finally, BinText provides useful information about strings in an intuitive graphical interface with the options to search, filter and store the output data in the table as depicted in Fig. 3. In the Windows operating system the shortcut to this application or the application itself may be copied to the folder C:\Users\<username>\AppData\Roaming\Microsoft\Windows\SendTo (in Explorer also accessible via the address shell:sendto), that way, it'll be always available for quick use [9].



```

Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. Všetky práva vyhradené.

C:\Users\PC>strings -n 7 -o "I:\Statická analýza\Strings v2.53\strings.exe" > C:\
\output.txt

output.txt - Poznámkový blok

Súbor  Úpravy  Formát  Zobrazit  Pomocník

99944:shell32.dll
99956:/accepteula
99980:-accepteula
100004:/accepteula
100016:-accepteula
100028:Software\Microsoft\windows nt\currentversion
100076:ProductName
100120:Software\Microsoft\windows NT\CurrentVersion\Server\ServerLevels
100188:NanoServer
100200:Accept Eula (Y/N)?
100232:This is the first run of this program. You must accept EULA to continue.
100384:Use -accepteula to accept EULA.
100452:Software\Sysinternals
100484:\StringFileInfo\%04X%04X\%s
100512:\StringFileInfo\%04X%04X\%s
100568:\VarFileInfo\Translation
100596:\VarFileInfo\Translation
100652:IsWow64Process
100668:Kernel32.dll
100684:wow64DisableWow64FsRedirection
100717:Failed to open %s:
100800:usage: %s [-a] [-f offset] [-b bytes] [-n length] [-o] [-s] [-u] <file or directory>
100888:-a      Ascii-only search (unicode and Ascii is default)
100948:-b      Bytes of file to scan
100980:-f      File offset at which to start scanning.
101028:-o      Print offset in file string was located
101076:-n      Minimum string length (default is 3)
101124:-s      Recurse subdirectories
101156:-u      Unicode-only search (Unicode and Ascii is default)

```

Fig. 2 Using the tool called Strings

Source: [5].

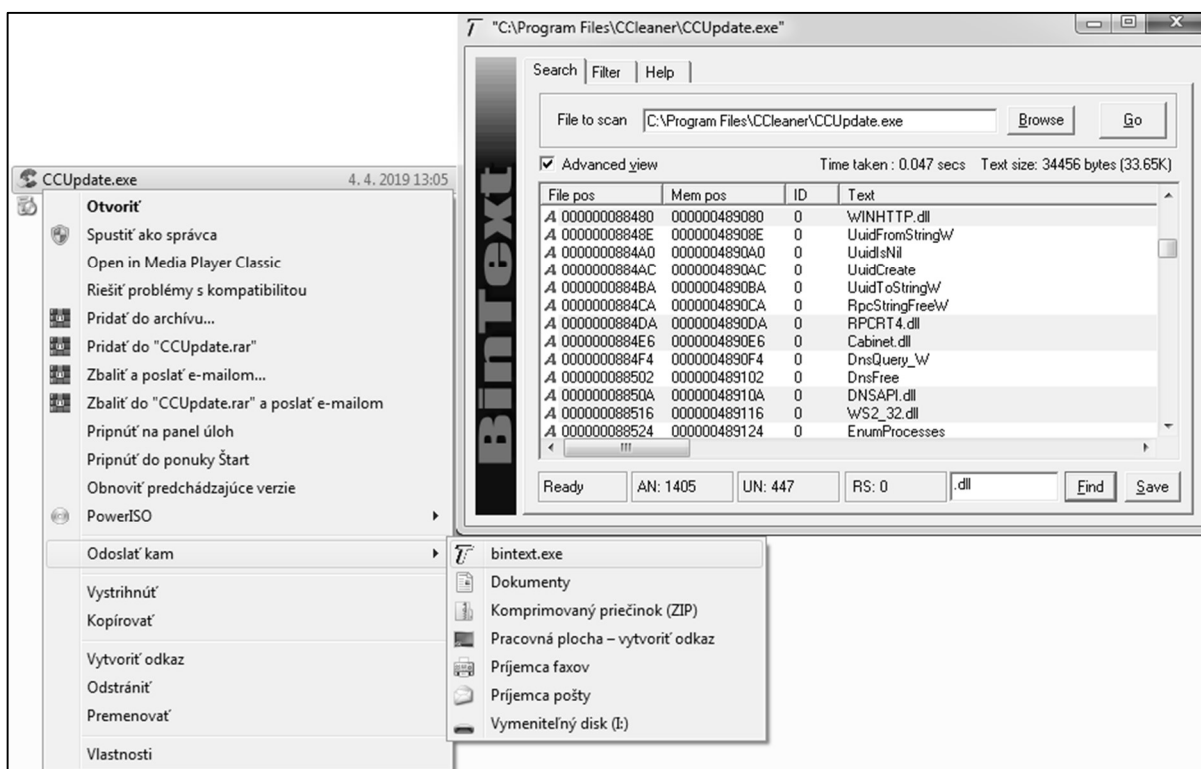


Fig. 3 Using the tool called BinText

Source: [3].

5 PORTABLE EXECUTABLE FILE FORMAT

In static analysis other very useful pieces of information can be obtained from the headers and sections of the Portable Executable (abbr. PE) file format such as the list of all Dynamic-link Libraries (abbr. DLL) and functions that the file imports. Binary executable files (usually with extensions like exe, dll, sys, acm, mui and others) used in all versions of Windows operating system (abbr. OS) are nowadays mostly in PE file format (rarely some legacy file formats are used) which is defined by the exact data structure. Data structure of PE file format contains the information necessary for the Windows OS loader to manage the wrapped executable code. As the name implies, the Portable Executable file format is portable between all versions of Windows OS regardless of the way the processor carries out the instructions of a computer program. Therefore the PE file can be executed on 32-bit systems as well as 64-bit systems [3].

The data structure of the PE file format apart from the actual application code and application data also defines the header where you can find detailed information about that program. Excluding the program code itself, the file header is one of the main sources of information in the static analysis, mainly because the header is available immediately at the start of the analysis and it can provide a first insight into the parameters and features of the analysed malware. Fig. 4 shows the structure of PE files which

begins with a header containing information about the code, the type of application, the required library functions, the required disk space, the creation date and many more. Just the list of used libraries and function calls can reveal many features of the program [5].

A PE file consists of a number of headers and sections. To maintain compatibility with the old Microsoft Disk Operating System (abbr. MS-DOS), each PE file begins with a header programmed for that system. This header is known as IMAGE_DOS_HEADER. In most cases it only contains the message "This program cannot be run in DOS mode." Especially the first e_magic field is interesting from an analyst's point of view because it is always at the beginning of each executable file and it has the fixed value of two characters (MZ). Therefore, if the analyst knows that this is an executable file, but there are no MZ characters at the beginning of the file, it is possible that the file is encrypted. Additionally these characters, together with that MS-DOS message, can help us find out the encryption key and decrypt the program because the values of both these fields of data are known [3], [4].

The IMAGE_FILE_HEADER (PE Header) structure contains basic information about the file, such as the date and time the file was created (TimeStamp), the number of sections which immediately follow the headers (NumberOfSections), the processor architecture (Machine) for which the program is intended etc. Such pieces of information are very important in the

Offset (h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	Layout		
0x0000	0x5045 = "MZ"		e_cblp		e_cp		e_cric		e_cparhdr		e_minalloc		e_maxalloc		e_ss		DOS Header		
0x0010	e_sp		e_csum		e_ip		e_cs		e_lfarc		e_ovno		e_res[4]		e_res[4]				
0x0020	e_res[4]		e_res[4]		e_oemid		e_oeminfo		e_res2[10]		e_res2[10]		e_res2[10]		e_res2[10]				
0x0030	e_res2[10]		e_res2[10]		e_res2[10]		e_res2[10]		e_res2[10]		e_res2[10]		0x0088 - Pointer to PE header						
0x0040	"This program cannot be run in DOS mode"																DOS Stub		
0x0050																			
0x0060																			
0x0070																			
0x0080	Undocumented								0x4550 = "PE"				Machine (CPU architecture)		NumberOfSections		PE Header		
0x0090	TimeStamp (file creation date and time)				PointerToSymbolTable				NumberOfSymbols				SizeOfOptional Header		Characteristics (specific property of an executable file)				
0x00A0	Magic (identifies the file e.g. EXE, ROM, PEE4)		MajorLinker Version		MinorLinker Version		SizeOfCode				SizeOfInitializedData				SizeOfUninitializedData				Optional PE Header
0x00B0	AddressOfEntryPoint (where the loader starts code execution)						BaseOfCode				BaseOfData				ImageBase (the address of image when loaded into memory)				
0x00C0	SectionAlignment				FileAlignment				MajorOperating SystemVersion		MinorOperating SystemVersion		MajorImage Version		MinorImage Version				
0x00E0	MajorSubsystem Version		MinorSubsystem Version		Win32VersionValue				SizeOfImage				SizeOfHeaders						
0x00F0	Checksum				Subsystem (CLI, GUI, EFI, boot app, driver etc.)		DllCharacteristics		SizeOfStackReserve				SizeOfStackCommit						
0x0100	SizeOfHeapReserve				SizeOfHeapCommit				LoaderFlags				NumberOfRvaAndSizes						
0x0110	DD_VirtualAddress		DataDirectory_Size		DD_VirtualAddress		DataDirectory_Size		DD_VirtualAddress		DataDirectory_Size		Other DataDirectories ...						
...	Name								VirtualSize (size of the section when loaded into memory)				VirtualAddress				Section Header (.text, .data, .idata, .reloc, .rsrc, ...)		
...	SizeOfRawData (size of the section or initialized data on disk)				PointerToRawData				PointerToRelocations				PointerToLinenumbers						
...	NumberOf Relocations		NumberOf Linenumbers		Characteristics				Other Sections ...										

Fig. 4 Structure of a portable executable file format

Source: [2].

static analysis. For example, the creation date of the file will determine whether it is an old sample or a new one that has not yet been scanned by an antivirus technology. Also a value stored in TimeDateStamp could not make any sense at all (referring to the future or the distant past). This artefact usually deepens our suspicions that the file may be malicious [4].

Moreover the header of PE file includes a structure called IMAGE_OPTIONAL_HEADER (Optional PE Header), which contains additional pieces of information for static analysis. There is an important field called AddressOfEntryPoint that contains the address of the entry point at which the program execution starts. The ImageBase field is also essential. It determines at which address in the memory the image of the program should be placed. Its default value is always 0x00400000 (for the DLL it is 0x10000000) and, as with TimeDateStamp, another value can be a sign of something potentially malicious.

The headers are followed by a table of sections and sections themselves which are an excellent source of information for forensic analysis. Here we will be interested in the sizes of individual sections. The virtual size (VirtualSize) specifies how much space should be reserved for the section when loaded into memory. The field named SizeOfRawData contains the size of the section or the size of the initialized data on disk. These sizes should be with small variations approximately the same. If the virtual size is much larger than the size of raw data, it might indicate that the file has been compressed [4].

One of the most useful pieces of information that we can gather about an executable is the list of functions that it imports. Imports are functions used by one program that are actually stored in a different program, such as code libraries that contain functionality common to many programs. Code libraries can be connected to the main executable by linking. Programmers link imports to their programs so that they don't need to re-implement certain functionality in multiple programs. The information we can find in the PE file header depends on how the library code has been linked. Code libraries can be linked statically, at runtime, or dynamically [2].

Static linking is the process of copying the entire code of imported functions directly into the body of the program what may result in a huge increase of the file size. Because of this impractical fact, static linking is not very used nowadays. In the field of malware analysis, dynamic and runtime linking is crucial [4].

When dynamic linking is used, program imports functions during its compilation. The code of the function is not stored directly in the program but it is stored only as a reference in the header of PE file. The .idata section contains the import directory table which includes a list of entries for every DLL which is loaded by the executable. In the first stage of the analysis, thanks to the import table the analyst can

figure out some of the application functions such as the feature to connect to the Internet or work with other files or resources. Based on this we can search for other artefacts such as IP addresses, domain names, file or application paths and so on [3].

The specialty of malware developers is the runtime linking. In the runtime linking, the functions are called during the program execution when a specific function is directly requested. Functions are neither imported at the time of compilation nor embedded directly into the program code. These functions are often called using the system functions (known as system API) such as LoadLibrary, GetProcAddress, LdrLoadLibrary, LdrGetProcAddr or using a serial number (each function has an assigned number). Then those system functions can be found in the import directory table. Runtime linking is usually used in the programs that are encoded, compressed or encrypted, and their code is used as a malware loader that extracts or decrypts the code of the application itself, which then loads the required libraries at runtime. Malware authors take advantage of the compression or the encryption to hide program functionality but it can be sometimes found in legitimate applications as well [2], [4].

5.1 Tools CEF Explorer, GT2

The tool called CFF Explorer allows you to extract the metadata from the PE file header as can be seen in Fig. 5. Additionally it offers the basic translation of machine language into assembly language, but in practice it is preferred to use specialized tools. The main advantage of this tool is that it presents the results completely and precisely, including the offset values, hexadecimal values with their meaning and other values a field might contain. On the other hand, the program expects that the user will be professionally experienced and able to correctly interpret the listed values. Therefore it does not inform the user about any anomalies and does not present any special results by their significance in forensic analysis, but only completely presents the results in the order in which they were found out [10].

GT2 is a command line program which is able to identify most of the executable files and archives by their binary signatures. So it is different from standard Windows filetype detection since it does not consider the file's extension by default. In addition, it can also read and analyse the metadata obtained from the file header [5], [11].

The frequently used tool called Dependency Walker can list all DLL libraries used in executable programs (this feature is also included in the tools mentioned above). Dependency Walker also displays a recursive tree of all the dependencies of the executable file (all the files it requires to run) which is evident in Fig. 6 [12].

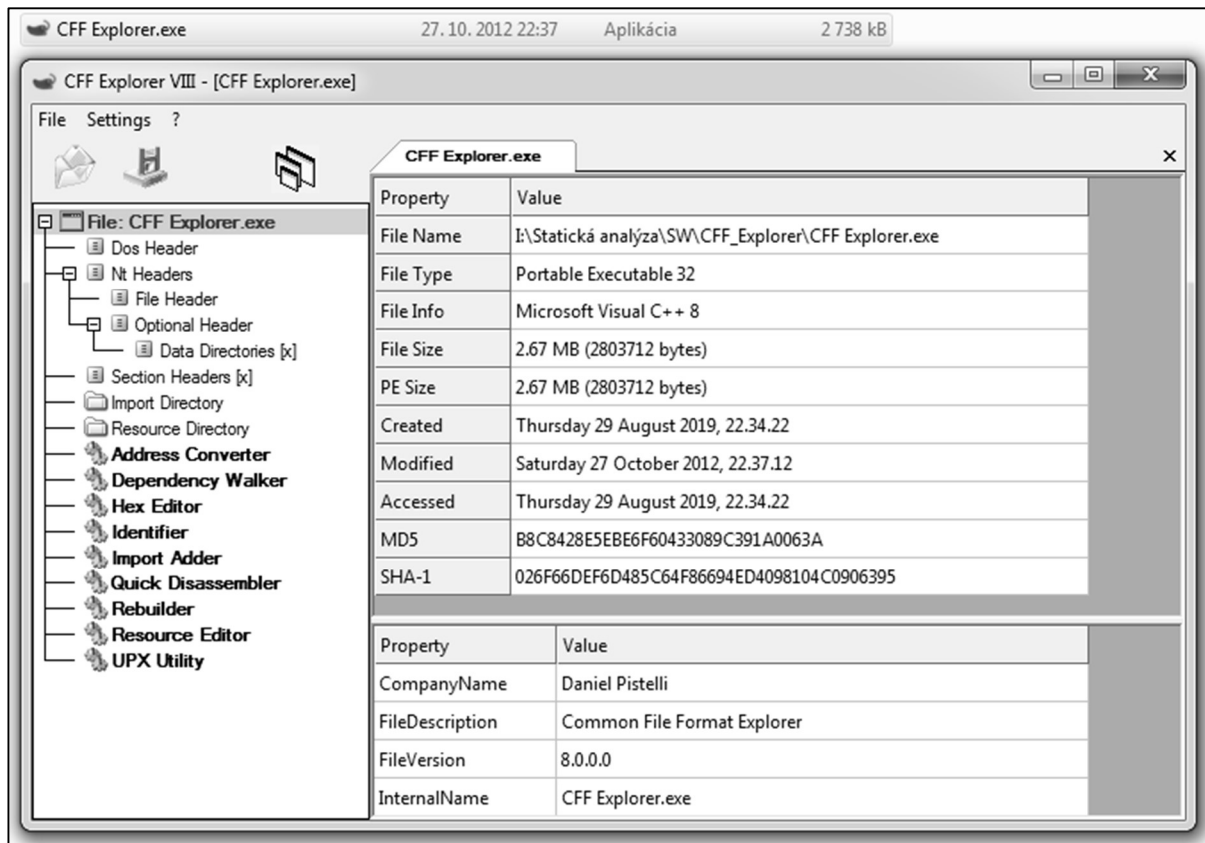


Fig. 5 Using the tool called CFF Explorer
Source: [5].

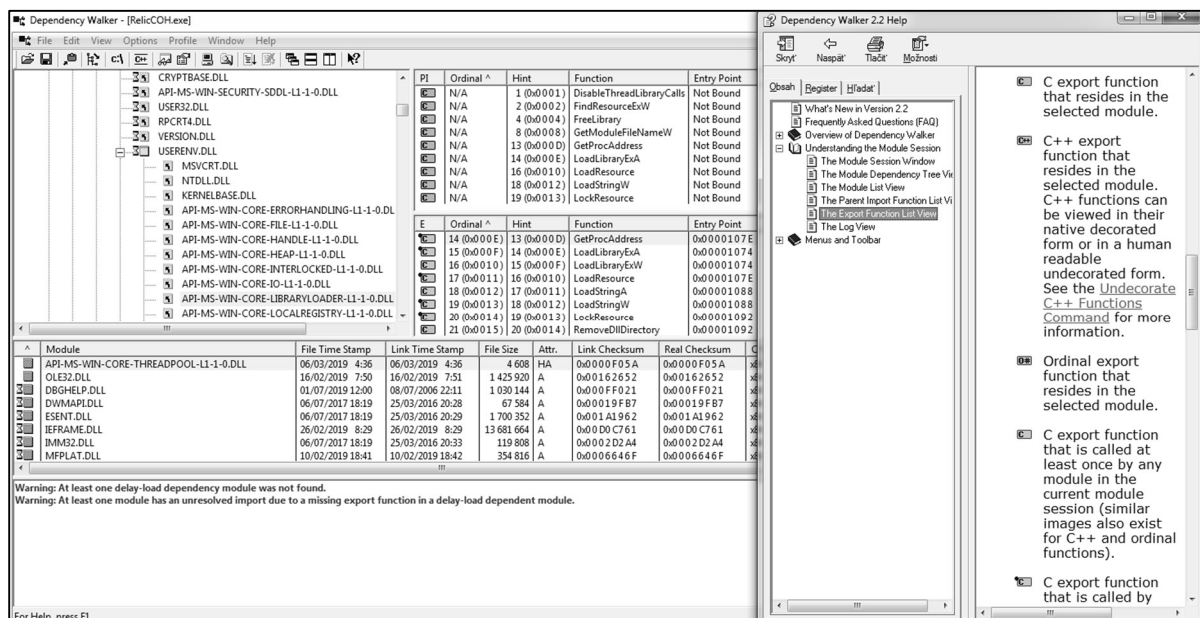


Fig. 6 Using the tool called Dependency Walker
Source: [12].

6 COMPRESSION OF MALWARE

De-obfuscation is the process of turning unintelligible information into something that you can understand. De-obfuscation is an undeniable requirement for malware analysis. Decoding, decryption, and packing are classified as forms of obfuscation. Although these terms differ slightly in a technical sense, they're all methods that attackers use to keep eyes off certain information. Without de-obfuscation techniques, your understanding of malware and its capabilities will be limited [1].

Malware compression is a very popular method for encrypting malicious programs because there are a lot of free and easy-to-use utilities that can do it. Compressed malware is smaller in size, difficult to detect by antivirus programs and difficult to analyse. The principle of the compression is to transform the binary code of the executable file into another form. As a result of this change, malware can escape the attention of antivirus programs when detecting signatures, because after each use of the compression tool, a new, unique sample is created that the antivirus databases do not recognize. Even several compression tools can be used on a single sample what may reduce the chance of successful detection [4].

When trying to statically analyze packaged malware, an extreme lack of information is evident. No interesting strings were found, the list of imported functions will be minimal (usually LoadLibrary and GetProcAddress) and all program instructions will be encrypted. The purpose of unpacking is to remove the layer of confusion applied to the program when it was packaged. There are many different methods for unpacking programs, most of which can be classified as manual or automated methods. Automated unpackers can definitely save you time, but they don't always work [2].

There are many special tools that automate and greatly simplify the detection of the packer (software for compression of malware). One of the most used are the veteran PEiD and frequently updated Exeinfo PE. Both applications provide the user with an intuitive graphical interface.

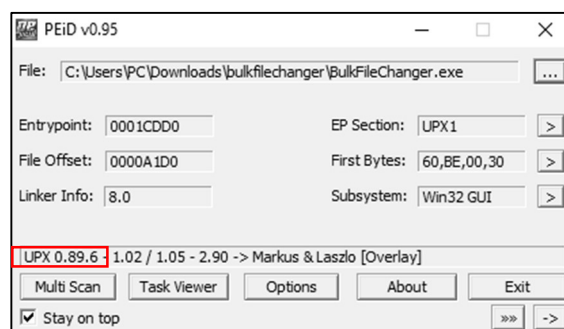


Fig. 7 Using the tool called PEiD
Source: [13].

6.1 PEiD and Exeinfo Pe

The most popular and most widely used program for basic analysis of malware is PEiD which can extract essential information from the file header and identify the type of compression or a compiler used to create the malware as shown in Fig. 7. It can detect over 470 different signatures in PE files. Official support and development of this tool has ended, but it is still often used in the analysis, mainly because it is still possible to add a new signature to the database based on which the compression methods are identified [13].

Exeinfo PE is another free portable application for extracting pieces of information about compression tools from executable files (Fig. 8). The latest version of the application (v0.0.5.6) is capable of detecting more than 1040 specific signatures of PE files. An external database containing approximately 4500 additional signatures (may not be reliable) is included with the application. Furthermore, the user may use other 490 signatures of files that are not executable. This application also provides a lot of information which is able to extract from the PE file header. Its functionality can be further extended by downloadable add-ons [14].

After successfully identifying the packer used in the creation of malware we can proceed further with its decompressing and disassembling (translating machine language into assembly language - assembler) with a number of specific tools. Then the analyst is free to closely analyse the malicious code.

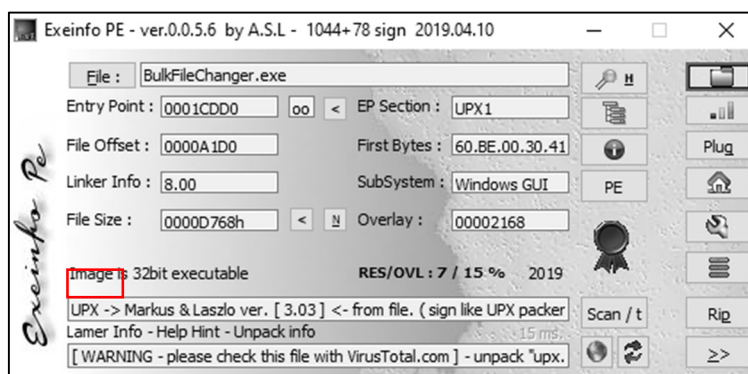


Fig. 8 Using the tool called Exeinfo PE
Source: [14].

7 CONCLUSION

One of the goals of this work is to get yourself familiar with the malware analysis, specifically with the complex subject such as static analysis. This includes clarifying what the analysis is and what it is used for. This paper presents a brief overview of basic methods and tools used in static analysis. Another contribution of this paper is further description of these analytical tools and methods. Finally, after testing the analytical tools, it is advised to combine them into a single package and automate their functions to reduce the time required to perform a static analysis, while ensuring that the resulting malicious file report contains all the necessary information.

References

- [1] LIGH, M. H., ADAIR, S., HARTSTEIN, B., RICHARD, M.: *Malware Analyst's Cookbook*. Indianapolis : Wiley Publishing, Inc., 2011. s. 746. ISBN 978-0-470-61303-0.
- [2] SIKORSKI, M., HONIG, A.: *Practical Malware Analysis*. San Francisco : No Starch Press, Inc., 2012. s. 802. ISBN-10: 1-59327-290-1.
- [3] KRÁL, B.: *Forenzní analýza malware*. Brno : Vysoké učení technické v Brně, 2018, s. 63.
- [4] DANILOV, M.: *Metody a nástroje malwarové analýzy*. Praha : Vysoká škola ekonomická v Praze, 2016. s. 85.
- [5] FUJTIK, O.: *Zjišťování podobnosti malware*. Brno : Masarykova univerzita, 2014. s. 72.
- [6] VirusTotal. [Online]. [accessed 20. July 2019]. Retrieved from: <<https://www.virustotal.com/gui>>
- [7] Strings – Windows Sysinternals. [Online]. [accessed 20. July 2019]. Retrieved from: <<https://docs.microsoft.com/en-us/sysinternals/downloads/strings>>
- [8] HexDive 0.6. [Online]. [accessed 20. July 2019]. Retrieved from: <<http://www.hexacorn.com/blog/category/software-releases/hexdive>>
- [9] BinText. [Online]. [accessed 20. July 2019]. Retrieved from: <<https://www.aldeid.com/wiki/BinText>>
- [10] Explorer Suite. [Online]. [accessed 25. July 2019]. Retrieved from: https://ntcore.com/?page_id=388
- [11] GT2 0.34. [Online]. [accessed 25. July 2019]. Retrieved from: <<http://www.helger.com/gt/gt2.htm>>
- [12] Dependency Walker 2.2. [Online]. [accessed 25. July 2019]. Retrieved from: <<http://www.dependencywalker.com>>
- [13] PEiD. [Online]. [accessed 1. February 2020]. Retrieved from: <<https://www.aldeid.com/wiki/PEiD>>
- [14] Exeinfo PE. [Online]. [accessed 1. February 2020]. Retrieved from: <<https://www.softpedia.com/get/Programming/Packers-Crypters-Protectors/ExEinfo-PE.shtml>>

Lt. Dipl. Eng. Andrej **FEDÁK** (PhD. student)
 Department of Computer Science
 Armed Forces Academy of General M. R. Štefánik
 Demänová 393
 031 01 Liptovský Mikuláš
 Slovak Republic
 E-mail: andrej.fedak@gmail.com

Prof. Dipl. Eng. Jozef **ŠTULRAJTER**, CSc.
 Armed Forces Academy of General M. R. Štefánik
 Department of Computer Science
 Demänová 393
 031 01 Liptovský Mikuláš
 Slovak Republic
 E-mail: jozef.stulrajter@aos.sk

Andrej Fedák - was born in Žiar nad Hronom in 1994. He received his engineering degree from the Armed Forces Academy of General M. R. in the field of Military Communication and Information Systems. Nowadays, he is an officer of aeronautical ground information systems - Air Force Headquarters. His research is focuses on computer networks, information systems, information and cyber security.

Jozef Štulrajter works as a professor at the Department of Informatics, Armed Forces Academy of General M. R. Štefánik in Liptovský Mikuláš. He graduated (Ing.) at the Military Technical College in 1974. He obtained the degree of CSc. diploma in Theoretical Electrical Engineering - Theory of Circuits and Systems of the Military Academy in Liptovský Mikuláš in 1992. His research interests include Information and Communication Technology (ICTs), computer architectures, image coding, computer security.