International Journal of Research in Engineering and Science (IJRES)
ISSN (Online): 2320-9364, ISSN (Print): 2320-9356
www.ijres.org Volume 2 Issue 10 ǁ October 2014 ǁ PP.38-41

Information security and programming language C_s

Russin research center for artificult intellegence

Abstruct : *The secure language C_s and operating system created by using the language give the full protection from harmful programs and ciber-attacks. We give more detailed syntax of C_s. The syntax inherits the syntax of C++ but we remove operators that violet security of a computer and we remove all superfluous operators and functions. In particular, we remove pointers to protect codes of programs from changing by pointers. We add logic-mathematics language. So it is enough to formulate a mathematical problem by the language and a computer creates needed program. The compiler of C_s provides continuous part of memory to codes and continuous part of memory to data. The code area is protected from changing by any other code. The data area has access only to its program and is protected from wrong addressing of any part of the data. The designer of the system (C_s and its operating system) has a server that provides a variety of services including unlimited access to Internet. The server provides protection against cyber-attacks such as DDOS by disabling sources of cyber-attacks up to by disabling non-users of the system since software of users is protected from harmful programs. The server collects information from all sites on Internet. This information is provided to users through their operating systems. These systems process information coming from Internet in accordance with the standards of C_s.*

I. Introduction

Who owns information he owns the world. This sentence reflects the role of information in our civilization. The modern civilization can not exist without information. By the words of Norbert Wiener, "Information is not matter or energy, information is information." He stressed that the main thing is not matter or energy, the main things is information. The acquisition, overall prevalence and data protection are spent huge amounts of money. The acquisition of information is provided by science, prevalence of is performed by education. But information can not exist without protection.

The significant part of the crimes is committed in information. Only the cost of computer crime was almost a billion dollars in 2012. Theft of data and information from corporate networks and databases, substitution of information, forgery of documents in electronic form and industrial espionage become rampant. But spread of viruses and harmful programs acquires special scope. The huge amounts of money is spent to protect information.

Full protection of information does not exist now and will not exist in the future. But complete protection from viruses, harmful programs and cyber-attacks exists. The system realizing such protection is given by the secure programming language C_s. Preliminary communication on C_s was given in [1]..

II. Syntax of language

Syntax of C_s inherits the syntax of C++ [2]. But we remove from C++ operators that violate security of computers, we remove all superfluous in C++, and we add all achievements of contemporary languages. In particular, we remove pointers because they provide unlimited access to any memory location. But there are implicit pointers. They are names of variables and functions. These pointers are dereferenced automatically, pointers of pointers are dereferenced automatically, too. Pointers are deleted in the language Ruby [3]. Automatic multi-level dereference of implicit pointers was first implemented in Algol-60 [4]. Operators random allocation and release of memory manually are deleted too, because memory for data and programs is allocated automatically when a block of a program begins running and is cleared when the block is ended. The memory is not cleared if it is declared with of the class static or extern. So the modification "public" is superfluous and incorporation of data is superfluous, too. This allows to select continuous but isolated areas of memory for a program and data and allows to protect memory for programs from changing them by codes. Definitions of arrays are changed. At a declaration, both maximum length of an array and a variable for storing current length of the array are initialized. Changes in the length of the array are recorded into this variable automatically. A program is interrupted if the address is outside of an array. This ensures safety of programs and data, too. A

control over the correct addressing is carried out not only for arrays, but also for other data, in particular, for structures containing arrays and for all data memory. A control over the correct address is also carried out for list structures.

Here we give an example of an array declaration: int M [1000, m] [2] = {0,1 }; The array has a variable m of length in the first index. The maximum length of the first component equals 1000, a current length is 1 and the length is placed automatically into the variable m. The length in the second index is fixed. The program is interrupted if the length of array becomes more than declared. A processing of the interruption can be used for automatic increasing length and for continuing to execute the program.

Operators "while" and "do-while" are removed since their functions are performed by the operator "for". This operator is added by the fourth parameter setting the maximum number of cycles. By default, this value is equal to one million.

III. Other features of language

The language is also designed for easing use. The C++ language was created as functional. This simplifies a design of a compiler, but hampered the use of this language. In contrast, the language of C_s is designed to simplify its use by adding into a compiler many functions. The language is added by compatibility with logic-mathematical language. In particular, a universal quantifier is interpreted as indicating ranges of its variables. An existential quantifier is interpreted as finding solutions its formulas. For example, a programming of integral can be represented by mathematical notation. The precisely of integration is 6 digits of mantis by default but can be specified by user, too. Two steps of integration are calculated, one step is half the other. If the precisely is not enough then these steps are increased or decreased. If the integrand has a singularity then its limit is calculated as speed of increasing or decreasing. If the limit is not compatible with the integration then the calculation of the integral is interrupted. The logic-mathematical language is also used for data processing. If the language is not enough to get a problem then it must be added in accordance with the rules of logic. Such additions will enrich mathematics and delete existing ambiguities in it. Symbols of logic-mathematical language are displayed to simplify its use.These symbols can be added to the keyboard using additional registers.

There are other features of the language C_s. Everything new in modern programming languages is added to C_s.

IV. Compiler

The compiler of C_s must detect program errors as much as possible. The compiler must also determine size of memory allocated to program.

From mathematical point of view, each class C++ is an algebra since this class contains a set (data) and operations (functions) over this set. Software project is also an algebra, and its classes are subalgebras. Users can have multiple algebras but these algebras are isolated from each other, and exchange of information between them is impossible.

The compiler allocates memory to each algebra. This memory has a code area and a data area. The code area is protected from changing by any other code and occupies a continuous part of memory. The data area is enabled only by this algebra and also occupies a continuous part of memory. External memory is an extension of this area, and also is enable only by the algebra. This allows to ensure the security of data and codes with minimum time loss.

The compiler uses relative addresses of codes and data. The address 0 for codes is address of beginning of their area, the address 0 for data is address of beginning of data area. Code area is fixed and compiler sets the size of the area taking into account the dynamically loaded programs. Real addresses are provided by the operating system, it places the data area directly behind the area teams.

V. Operating system

The operating system (OS) is constructed by means of language C_s. OS is also an algebra and the compiler allocates not only relative but also a real memory to OS. In this case the compiler performs a simple function of the operating systems. OS of C_s has all basic functions of operating systems: sets an interface between devices of the computer system and application programs, operates by devices and computational processes, efficiently allocates computing resources between computing processes, but reliability of the calculations is provided by processor. So OS does not interfere in inner workings of programs and does not collect information about workings programs. OS blocks any interference in any work of programs, in particular, OS has no administrator of user computer. But OS increases a memory area for data on demands by algebras. If the increase of the data area is not possible then OS performs a garbage collection by moving

algebras in memory and puts the algebra to vacant memory.

OS allocates memory for the algebra of the system designer. Only the designer has access to this memory.

OS provides connection with Internet and serves as a browser by using information provided by the site of designer. OS provides protection from viruses and harmful programs. Information coming from Internet is processed in accordance with the standards of C_s.

By requirements of algebras, OS can delete codes coming from Internet, can permit audio and video information, can permit codes to read some memory of algebra or may allow codes to write in some data memory of the algebra. OS performs the same steps for information getting from external medium of informations. A processor must reduce the time spent on these operations by realizing functions of these operations.

VI. Closure of system

The system is closed. All programs of OS and users must be created only by the compiler of C_s. By own initiative or at requests of users, the designer acquires a software of other designers, processes the software in accordance with the standards of C_s and with systematic approach , and places it into a system library. The systematic approach requires that all operations are fulfilled by unique rules. Revised programs differ from originals so much that problems of copyright do not arise.

The system will become attractive to users by thanks to the closure since all the necessary software is concentrated in the system and it has no license restrictions. The system software is available for free but it gives advertisements to finance activity of the system.

VII. Program of remaking

A special program remakes programs of other designers in accordance with the standards of C_s. The program removes all superfluous, in particular, a protection of licensing programs. In addition to remaking of programs, the program generates a source code of C_s for these programs. This allows the designer to change programs in order to implement a systematic approach to the fulfillment of their functions. A systematic approach allows to work with new programs without studying documentation. In particular, the user interface should be implemented by the same rules. Access to reference information must also be carried out by the same rules,. All this provides ease of using.

VIII. Website of designer

The designer is a very large team of programmers since it is necessary to create a huge amount of software. The designer creates programs and provides support for them. All programs are tested for safety and are entered into a system library, which is available to copy by any users of the system. The website of designer is a server for all users of the system, too. It provides a variety of services including unlimited access to Internet. For this, the site provides protection against cyber-attacks such as DDOS by disabling sources of cyber-attacks. If it is necessary then all non-registered users are turned on by routers of the site. Users of the system are registered. They are not disabled since their software is protected from harmful programs. The site also serves as a browser to collect information from all sites on Internet. This information is provided to users through their OS.

IX. Security of corporate information

Full protection of corporate information does not exist.

One of the problems of the protection is to block unauthorized access to information. As a rule, such protection is provided by passwords but this is ineffective. The most effective security is given by biometric . Sales of biometric equipment only in the United States in 2012 exceeded $ 9 billion. In this year, sales are tens of billions of dollars. Fingerprint scanners form 44 % of this volume , facial recognition systems form 14%, devices of recognition by the shape of palms form 13%, voice devices form 10%, irises form 8%, and signature verifications form 2%.

The recognition by voice is most promising since it does not require special equipment. This recognition is implemented by software. But the voices can be imitated. So a recognition by keyboard handwriting style is more perspective. Keyboard handwriting style exists just as a radio operator's handwriting style. The recognition of the style does not require a special equipment, too. Imitation of keyboard writing style is almost impossible even by observing a work of a user. The style of a user is writable and readable in user computer only by OC. This has full protection by the system.

Existing systems of information security are uncompetitive capable with respect to the system with C_s since the system provides full protection from harmful programs and cyber-attacks and a significant protection against unauthorized access to corporate information. Almost all users will be included in the system.

X. Libraries

All programs are placed in libraries. There are an operating library, several system libraries and several user libraries.

The operating library contains programs of OS. The library is writable only by the designer and is readable by only OS. Every user has a copy of the operating library.

A system library contains programs that are needed to all users. The most complete system library is placed on the designer website. It is writable only by designer but is available to copy to any user of the system. In a local area network, system library is located in the host computer. Every user has its own system library. Each system library is writable as the designer as a user, but a user can write only copied programs of the designer system library. The library is readable only by its user. A system library of local network is placed on host computer and is writable as the designer as the administrator of the network. But the administrator can only copy programs of designer library. The library is readable by every user of the network.

User library is available for writing only by the compiler or by coping programs of the designer library. But in a local area of network, there is a readily available user library. Every user can read programs of the library and can write programs from his library. Additionally, the compiler of administrator of the network can write programs. There can be several readily available user libraries of network. Every library has the own administrator.

XI. Stages of development of system

The program of remaking the other programs must be constructed at first. The program is created by means of language C++. After creating, the program remakes itself in accordance with the standard of C_s. This program is used for remaking the compiler C++. The result of remake is used to generate the C_s compiler. Then Windows is remade and is added by components that enhance computer and corporate security. After this the kernel of the system is complied and the shell of the system begins to construct. This is the most time-consuming part of the construction. Almost all the existing software must be remade in accordance with the standards of C_s. Simultaneously with the construction of the shell, CPU, BIOS, and other components of motherboards must be improved in order to enhance security. In particular, the program of remaking other programs should be built into the motherboard. The processor is also complemented by codes speeding up computers and some functions of security. Then OS must be respectively corrected.

References

[1] M.A. Malkov, Secure programming language Cs, *International J. of Research in Engineering and Science*, **2**, issue 5, ver. 2, 42-44, 2014.

[2] B. Stroustrup, *The C++ programming language*, fourth edition, Addison-Wesley, 2013.

[3] David Thomas, Chad Fowler, Andrew Hunt, *Programming Ruby: The Pragmatic Programmer's Guide*, second edition, Addison-Wesley, 2004.

[4] J.W. Backus, F.L. Bauer, J.Green, C. Katz, J. McCarthy, P. Naur, A.J. Perlis, H. Rutishauser, K. Samuelson, B. Vauquois, J.H. Wegstein, A. van Wijngaarden, M. Woodger, *Revised Report on the Algorithmic Language Algol 60*, Copenhagen, 1960.