CONTEMPORARY TRENDS IN THE DEVELOPMENT OF INFORMATION SECURITY AND COMPUTER VIROLOGY

Eugene NICKOLOV

1. Introduction

The information and its security were subject of special attention throughout the ages. All achievements in this area were highly appreciated and were finding immediate application. During the past years, the development of the contemporary society is correlated with the continuously growing information activity. The information accumulation and its movement through the Internet environment are the next challenge to not only to information security, but also to broader aspects of societal security. In these conditions the notion of *information security* becomes a basic instrument for evaluating the risks for an information unit. One of the main effects on the information security of a given information object stem from computer viruses and their derivatives. The science of *Computer Virology* emerges as a response to these challenges. It deals with analysis and synthesis of virus signatures used by the anti virus programs for detection, blocking and removal of the computer viruses. In the contemporary information society the computer virology turns into one of the most important agents for mastering of the "malicious thinking" and for guaranteeing the necessary information security.

2. Information Security

Let us consider the notion of information security as containing some components connected by specific relationships. The number of these components has to be a reasonable approximation to the exact image of the real phenomena. The number and the functional names of the components can be changed in the wanted direction if needed. The principle of the multiple decomposition can be applied in the analysis of each separate component, during which new components can appear or some of the existing can drop out. The relation between the components can be bound in specific

mathematical relationships with functional character influenced by the changes in a number of given arguments. Lately, the information security function can be controlled by relevant parameters representing the projection of the separate components and their relative weight in the general pattern, when some assumptions and reasonable simplifications are made. Along with that, the information security can be interpreted as the possibility for a definite information unit with a definite composition and structure to be moved from the point A to the point B with guaranteed *constancy of the integrity* of the information unit and *constancy of the movement* of the source and receptor points. In this case, *constancy* means a lack of possibility for the "malicious thinking" to change the preliminary defined behavior.

2.1 Components

Data Security (S_{DATA}). In the gradation of the components this is the offset. The possibilities for the security evaluation on a data level are generalized in this component. This means an analysis of the possible critical points for the specific forms in which data emerge, exist and disappear, such as bit, byte, word, block, and package.

Computer Security ($S_{COMPUTER}$). The next component generalizes the possibilities for a security evaluation on a computer level. An analysis is performed here of the possible critical points for the separate modules as processor, memory, peripherals.

Communication Security ($S_{COMMUNICATION}$). In this component the possibilities for the security evaluation on a communication level are generalized. The analysis of the possible critical points includes the communication environment, as well as the processes of input/output, transformation, and compression.

Network Security ($S_{NETWORK}$). Here the possibilities for a security evaluation on a network level are generalized. The analysis of the possible critical points includes the modules network drive, network protocol, and network architecture.

Mobile Security (S_{MOBILE}). This component is a generalization of the possibilities for a security evaluation on an information object mobility level. The analysis of the possible critical points includes the object coordinates, access authorization, transfer by satellite.

Manipulations Security ($S_{MANIPULATIONS}$). This is a component in which the possibilities for a security evaluation are generalized on operator manipulations level. The analysis of the possible critical points includes manipulations on the objects key, mouse, pen, screen, etc., as well as the types of manipulations press, click, and touch.

Biometric security ($S_{BIOMETRIC}$). In this component the possibilities for a security evaluation on the operator's biological characteristics are generalized. The analysis of

the possible critical points includes fingers, hands, palms, face, eyes, voice, scent, blood, and DNA.

Steganometric security ($S_{STEGANOMETRIC}$). The last component in this arrangement generalizes the possibilities for a security evaluation on the information object steganography level. The analysis of the possible critical points includes stegochannel, stegoobjects, and host signal.

2.2 Relations

The dynamics of the real processes require the formal definitions of the separate parts of a whole to be generalized by defining their relationships. A formal record of the above examinations for the information security components and their possible critical points, made by applying the multiple decompositions, is as follows:

$$S_{DATA} = F\ (S_{bit}, S_{byte}, S_{word}, S_{block}, S_{package}) \quad (1.1)$$
$$S_{COMPUTER} = F\ (S_{memory}, S_{processor}, S_{peripheral}) \quad (1.2)$$
$$S_{COMMUNICATION} = F\ (S_{input/output}, S_{transformation}, S_{compression}, S_{media}) \quad (1.3)$$
$$S_{NETWORK} = F\ (S_{device}, S_{protocol}, S_{architecture}) \quad (1.4)$$
$$S_{MOBILE} = F\ (S_{co\text{-}ordinates}, S_{authorization}, S_{satellite}) \quad (1.5)$$
$$S_{MANIPULATIONS} = F\ (S_{key}, S_{mouse}, S_{pen}, S_{screen}, S_{press}, S_{click}, S_{touch}) \quad (1.6)$$
$$S_{BIOMETRIC} = F\ (S_{fingers}, S_{hands}, S_{palms}, S_{face}, S_{eyes}, S_{voice}, S_{scent}, S_{blood}, S_{DNA}) \quad (1.7)$$
$$S_{STEGANOMETRIC} = F\ (S_{stegochannel}, S_{stegoobjects}, S_{hostsignal}) \quad (1.8)$$

2.3 Function and Arguments

The formalization can be prolonged and transformed as follows:

$$S_{INFORMATION} = F\ (\text{Data, Computer, Communication, Network, Mobile, Operator, Biometric, Stegometric}) \quad (1.9)$$

The general expression for the information security function and its arguments can be obtained by substitution:

$$S_{INFORMATION} = F\ [\ (\underline{\text{bit, byte, word, block, package}}),\ (\underline{\text{memory, processor, peripheral}}),\ (\underline{\text{input/output, transformation, compression, media}}),\ (\underline{\text{device, protocol, architecture}}),\ (\underline{\text{co-ordinates, authorization, satellite}}),\ (\underline{\text{key, mouse, pen, screen, press, click, touch}}),\ (\underline{\text{fingers, hands, palms, face, eyes, voice, scent, blood, DNA}}),\ (\underline{\text{stegochannel, stegoobjects, host signal}})\ ] \quad (1.10)$$

Depending on the kind and the depth of the investigations, the arguments can be defined as parameters with specific limits and relevant numeric values. In this way, a specific strategy for information security control can be realized along with guaranteeing and finding a reasonable balance among the requirements for security, speed, performance and price.

2.4 Procedures

The basic procedures for realizing the necessary information security level include *monitoring*, *blocking*, *removing*, *protection* and *verification*.

The procedures *monitoring*, *blocking* and *removing* have to be considered as a sequence of actions of the anti-virus programs on a single computer virus OR a single family computer viruses.

The procedure *protection* represents a functional unification of the above three procedures but on other objects. It should be considered as a sequence of actions of the anti-virus programs on some family of viruses OR all currently known families of viruses OR some family of virus derivatives OR all currently known families of virus derivatives.

The procedure *verification* contains two separate parts: 'identification' and 'access', which are very closely connected (this is the reason they are examined in a unified procedure). These parts in most cases accomplish an integrity check for specific information objects using control sums, which can be mathematical, steganometric and biometric.

The procedures discussed above represent a generalization of the contemporary views for information control in modern computer systems. They are the most important instruments of the information security, helping its control by relevant parameters, as well as the fast and effective evaluations of the multiple factors exerting influence on the operation of the separate workplace.

3. Computer Virology

3.1 Working principles

The first basic principle that computer viruses accede to is *reproduction*. The programs realizing the virus idea and its derivatives always accomplish the operation *self-reproduction*. At first, it was an ordinary copying with identical original and copy and this was making the work of the anti virus programs much easier. As a consequence, the virus writers very soon proceeded to a reproduction with a considerable difference between the copy and the original. Exceptional efforts are exerted to realize this idea and the achievements are considerable. Besides, nowadays

the number of the generations is an astronomical figure. This makes the reproduction the hottest point in the rivalry between the viruses and the anti-virus programs. Secondary results of this perpetual competition are not always visible and widely known, but every new model of a computer or computer system, every new version of an operating system or application package contains the latest results of this competition.

Transportation is the second basic principle which computer viruses abide. It realizes the quantitative accumulation of the virus idea and its moving in space. Initially, the movement of the virus was limited to a single computer but later a suitable carrier (changeable carrier with enabled/disabled writing) was used. The network development provoked gradually the use of a communication type carrier. During the past few years the Internet became the most global and handy carrier of the virus idea and imposed the need for relevant and constantly updated virus filters. Main tools for realizing these filters are the Internet protocols. They are subjected to continuous improvements but, unfortunately, the number of security holes does not decrease. Another solution is the increase of the security level of each work place.

The malicious thinking is the third main characteristic of the computer viruses. It marks the limit between the scientific problems in the field of informatics and the computer viruses. Initially, the *reproduction* and the *transportation* existed only in the research laboratories and were used for scientific purposes. But starting with the early 80s—along with the development of the personal computers—the *malicious thinking* began also to use *reproduction* and *transportation*.

Depending on its goals, the malicious thinking could be divided into the following basic types:

- **Destruction.** This is one of the first goals set and successfully realized by the malicious thinking. Main variants of destruction are: immediate, awaiting, single, and basic. Nowadays, a certain decrease of the virus' destructive action is observed, but its disappearance is highly unlikely.
- **Modification** is the next goal of the malicious thinking. Everything is subject to modification - operation systems, application packages, local programs. The public and the personal information are modified in order to be attacked or to be made ready for future action. This trend, which is very dangerous, dominates malicious thinking in the Internet environment. The main cause is the contemporary infrastructure of the computer systems, which is not adjusted for a global access to their resources. This determines the serious lapses in the systems controlling the access and in the whole computer security. This is true to some extent even for the most modern Internet systems. Their designers are often not able to study them carefully

in laboratory conditions. This is the reason security tests are performed in real conditions, which imply many risks. Unfortunately, this aspect is not secret for the malicious thinking.

- **Misappropriation**. This goal is comparatively new and its propagation is limited. Until recently, information in its various forms was often appropriated, but money was rarely appropriated. However, the e-commerce development in Internet environment menaces to become a strong motivation for the development of the malicious thinking in this direction. The classical forms of attack are related to hooking of personal information and accomplishing of transactions causing damages. Unfortunately, despite the widely advertised multi-digit identifiers and passwords, the malicious thinking achieves remarkable success though nobody speaks about it - neither those who gain nor those who lose.

Good-natured thinking is the fourth fundamental principle to which computer viruses may accede. It is an alternative of the third one – the malicious thinking. Its use for the past few years has been constant and probably will remain unchanged in the future. Its main characteristic is the absence of planned losses of resources and information. Usually, it is manifested through various strange effects - sound, music, speech, inscription, image, multimedia, Internet activity, etc. They cam be categorized in the following basic types:

- **Joke** - one of the first manifestations of the virus idea. Usually used by friends or acquaintances. Serious troubles are very rare and are caused by unconscious errors. The jokes are a herald of positive ideas and it may safely predicted that they will never disappear.

- **Advertising.** This kind had very modest presence in the past but the exclusive possibilities for its development, especially in Internet environment, reserve a great future for it. However, if the reasonable proportion between usefulness and boredom is upset then, figuratively speaking, it crosses the line and may be classified as malicious thinking.

- **Experiment**. This kind may be named “useful viruses” because they allow the investigation of complex systems that cannot be studied in a different way. If the experiments are made systematically by the appropriate organizations, they eliminate the risks and allow a reasonable management of the resources. But if experiments are made by curious programmers, they could also bring serious trouble.

3.2 The Investigation of computer viruses

The isolation of a working virus sample is the first step in virus investigation. In the past, computer virus designers did not take any precautions and every infected file provided such a sample. But later the viruses became encrypted and scattered in the file (files) body, which transformed the isolation in a highly qualified activity almost impossible for the common user.

The second step in the virus investigation is the program code *decomposition*. This is almost always non-trivial task. Very often the program code is repeatedly encrypted and compressed; non-standard identifiers, masks, and addressing are used; the virus writers rely on non-documented and badly known system functions, interruptions and procedures. Certain particularities of certain processor chips are used to a maximal extent. The implementation of a new virus idea is the most interesting for the anti-virus researchers. The family characteristics are unusable and the significance of every bit from the program code must be clarified. This process is one of the most difficult in the investigation.

The formal description of the program code is the next step aiming to prepare certain automated operations connected to the creation of a working model of the investigated computer virus. Most often the model is created on a matrix and vector basis and, sometimes, through the implementation of more complex mathematical instruments. The goal is to achieve object and class description in an appropriate hierarchy looking for some control of the events. But sometimes the examined sequence of actions cannot be realized, therefore one must rely on experience and intuition. Often the formal description implies complicated formulas calculating the needed coefficients. Sometimes it is impossible to use standard computing programs.

The computer virus *modeling* is an extension of the previous step. The basic problem is to verify all the known information and to create a working virus model. Now the researchers look for the process dynamics and the relevant narrow points of the created model's structure. Another important task is to create the so-called "temporal magnifier" allowing acceleration or delay of the local virus time. Thus, the full life cycle of a computer virus can be described and its future behavior can be forecasted. The factors, which are critical with regard to its reproduction and transportation schemes, are searched for in order to limit its spreading. The needed input data are loaded in the model and then a series of experiments are made during which preliminary planned statistical data are collected. Various modeling techniques are used, mainly analytical, simulative, or mixed. The choice is made generally according to the needed computing and non-computing processing and the needed experiments and statistics.

The decision-making is the step that requires good forecasting skills because the decision is made on the basis of a few virus samples, at best. It must be valid not only for the current generation of the investigated computer virus but also for the whole family. During the decision making process it is very important to take in consideration the program solutions that are realized already. The new solution must be incorporated in the existing ones without affecting the speed and the performance of the existing anti virus program. When principally new solutions are needed, long-term application and finalization of a separate programming module for the new class of computer viruses has to be created, bearing in mind the general performance and speed.

The program realization is the last step before getting the product ready for use. It is important here to choose such a programming language, or languages, that allow the creation of programs running simultaneously on various platforms, while taking into account the processor or the operating system. The correlation between the high-level languages and the assembler modules is also of a great significance, because the performance, the speed and the flexibility of the anti-virus program depend on it. All non-standard and specific operations of the anti-virus program, related to non-standard and specific functions and interruptions, are very often realized by the assembler modules. The screen interface also has a great importance for the final success of an anti-virus program. The expenditure needed to obtain a suitable solution is very often comparable with the rest of the expenses.

3.3 Detection of computer viruses

Signature analysis is one of the first methods applied for detection of computer viruses and their mutations. It consists of a continuous filtering of the information flows in certain spots of the computer configurations. The aim is to obtain a coincidence on an information unit level, most often byte, with the so-called virus signature. If such a coincidence occurs, this means that there is a virus code in the information flow. The creation of the virus signature of a separate virus or virus family is not a quickly-solvable problem. A non-repeating (unique), non-changing and the shortest possible sequence of hexadecimal or binary symbols must be found assuring a secure recognition of the viral code without false alarms. To find such a virus signature requires always some time, and this is seen as a serious disadvantage of this method. Another disadvantage is the consumption of system resources, which does not always go without penalty. A constant update of the virus signature base is needed, although, if Internet connection exists, this may be made automatically without disturbing the user.

The creation of self-mutating computer viruses is a serious challenge for this method, as each new reproduction requires some time for creating a new virus signature. Of

course, sometimes it is possible to create a virus signature which is valid for all possible virus copies, but this requires an investigation of the full virus life cycle. Regardless of these disadvantages, the signature analysis is the most used method. It provides a reasonable balance between price, flexibility and applicability. It is easy for use, highly reliable, particularly if its application is combined with other methods and means. The future of this method is in the creation of means for shortened search. This means to replace, according to certain principles, few bytes from the virus signature with a unique byte. This byte is grouped along with other similar bytes to form a representative excerpt valid for a big group of virus signatures. The investigations in this direction show that, if the rates of increasing the performance of the processors and the other system resources decrease, this will immediately lead to the creation of new means for management of the virus signature base. It is necessary to obtain a unified standard virus signature base for all commercial realizations of anti virus programs. This would be an exceptional success for the computer security of the contemporary computer systems. The Internet, which is a manifestation of the idea of globalization and unification, brings closer the integration of the different bases.

Integrity check is the next basic method for detecting computer viruses. It is based on the use of the so called *control sums*. Special procedures are started in a virus-free environment to process all or some file objects and system points by special mathematical methods. As a result, hexadecimal sequences are obtained and they are constant if the chosen objects and points remain unchanged. This means that every virus attack or an attempt for accidental or deliberate change of an object or a point will change the control sums and therefore will be immediately detected. The application of this method requires calculation of the control sum every time when the system is started or the user gets access to a file object or point. This method provides high security but is time consuming and requires a lot of resources that can be used differently. Another disadvantage is the *post factum* reaction, which is inadmissible in some applications. For this reason, when possible, the object integrity is verified some time before the objects are used in real actions and, if it is necessary, backup copies can be found and started. Another possibility is to start a self-restoring procedure for the change localization and removal by mathematical means. The existing applications of this method include sophisticated encoding and decoding procedures. They are applied after the control sums calculation in order to preserve them from manipulation. In some procedures the control sum base, after being encoded several times with accidental component codes, is divided into several parts that are stored separately. These parts on the other hand change their locations continuously. The particularities of this method limit its application to systems with relatively constant composition and structure, because in a dynamic environment a number of serious problems arise. The integrity check can be integrated with the use

of hardware security points using special integrated circuits with factory-made access control means. Such a protection offers almost ideal security but the prices are so high that very few real systems can afford it.

The combination of the integrity check with the signature analysis is a very good solution and can be often found in real systems.

Monitoring is the third method applied for detection of computer viruses and their derivatives. It requires creation of a relevant environment for monitoring the whole activity of a computer system. This includes:

Runtime evaluation of certain events and processes. Fully automated methods or an empirical human appraisal can be used. In the first case a database is created with standard runtimes, which are compared with the current values during the next starts and runs of the computer system. If a discrepancy bigger than a preliminary defined interval is detected, a warning message is generated. In the second case, the possible high skills and reflexes of the user can help the timely change detection in the computer system operation, but this method cannot be applied on a mass scale.

Evaluation of the non-standard screen interface reactions and behavior - automated methods with standard screens or other interface components also can be used for comparison, but the expenses are very high and their use is limited. It is easier to rely on the human factor but the requirements for high qualification are obligatory because the probability for false alarm is very high. Serious knowledge of the system and application resources is needed to obtain a high level of precision.

Evaluation of the input/output peripheral operations - the automated methods have the highest success in this case. Control sums including the run times for the relevant transactions are used relatively often. This is a good guarantee for the transactions' security and this solution has great prospects. The main problem is the accuracy of the measurements in the starting and ending points of the transaction, because thousandths and ten-thousandths of a second must be measured. In Internet the use of uniform time is necessary, but the possibilities for errors and false alarms are significant.

Restriction is the fourth method used in the computer virus detection. It uses two clearly defined areas of allowed and not allowed events and processes. Each frontier violation generates a warning message.

The permitted area represents a multitude of addresses, fields, operations, drives, etc., which are examined for reliability and possible regulations. The information flow movement (input/output points, routes, data volumes) is strongly regulated and the user is given a warning for each deviation. At present, these regulations can be applied only in relatively static systems, but the use of new algorithmic means for

self-verification of the information flow content will decrease the importance of this kind of restrictions and increase the importance of the access and identification control for each object that is in contact with the information flow. The biometric user information combined with some cryptographic methods can assure a personal biometric tracing of events and processes and, thus, realize restrictions in the permitted area. Each contact for which biometric information does not exist will be thrown out of the permitted area.

The forbidden area usually is the multitude of addresses, fields, operations, drives, etc., for which the examination of the reliability and possible regulations is negative. They are risk points needing a strong protection. The requirement for the static character of the systems is strict because in dynamic systems the formulation of clear criteria is very difficult. The availability of some information for past behavior of the risk points is very important and requires the presence of automatic registration systems. The increased resource consumption is compensated by the possibility for analyzing the accumulation of incidents in the risk points.

3.4 Removal of computer viruses

The removal of the computer viruses and their derivatives from the attacked objects and information flows is a risky process. Despite of the precautions and the collected experience about 5 % of all virus attacks and incidents cause non-recoverable destruction. As a rule, they are a consequence of the malicious thinking, which has planned such a development of the virus attack. In this case the only possibility is to find uninfected objects and to copy them on the destructed objects. If system resources and objects are damaged, the system has to be re-installed. If data is damaged, then eventual backup copies have to be used.

If the computer virus removal is possible and necessary, it goes on in the following order:

Localization. This is the initial step when the location/s/ of the computer virus body has to be pointed out exactly. The existence of a mechanism connecting the separate virus parts and of control sums assuring the identification of these parts ought to be clarified. Viruses have to be localized in time too, because some of them exist in continuously closed locations, which open in a couple of ticks to allow an information exchange. This opening occurs if certain conditions are realized, and that is not always possible. If the localization is impossible, it is very important to obtain information about the causes and to modify the next phases. The presence of registering mechanisms is essential because they allow to track the history and the steps of the infection.

Identification is the next step in the computer virus removal. It has two modifications. The fast identification aims to point at a specific group, family or class of computer viruses emphasizing on the speed. The exact identification is a prolongation of the fast one and its goal is to identify exactly the virus and to specify if a mutation, a new variant or a dead virus body is found. This is not always possible but it is important to achieve an identification of the future variants if the virus makes use of a known working principle.

Removal is the third step in the computer virus removal. The virus body is deleted most often and the separate parts of the infected objects are jointed. It is very important to find out the exact boundaries of the virus to make possible the conglutination of the different parts. If the virus is in the beginning or in the end of a given object, the procedure is trivial, but nevertheless risky. If the virus body is located in specific system areas, for example the initial sectors of the hard disks, there are two possibilities. The first one is to find the original image of the object, which is often stored somewhere and to restore the object. The second one is to generate again this system object if sufficient information about it is kept. A special attention has to be paid to the group of viruses, which encrypt the whole content of some media, for example hard disks. In this case it is very important to decrypt the media before the virus removal. If this requirement is violated the data decryption after the virus removal may be impossible.

Deactivation is the next step in the virus removal. It represents a variety of the *cleaning* and is applied when the prognosis for the virus cleaning is negative, i.e., when the virus cleaning would destroy the object, whereas after the deactivation the object keeps on functioning normally in spite of the dead virus body in it. The basic task is to find the starting point of the virus and to write specific information on it. Thus, the virus cannot reproduce and transport itself anymore. This solution is not always possible but, if it is, the results are very good especially when the change of one byte kills the virus body and the system continues to work.

Verification of existent, known and accessible control sums related to the restored object is the next and often the last step in the computer virus removal. It has to provide a warranty for the restored object integrity and working capacity. If possible, a new control sum is generated and its coincidence with the database is verified.

Conclusion

Information security in the contemporary society becomes a global gauge of the risks during the rise, the existence and the disappearance of the information. It is subject to control by relevant parameters with a reasonable compromise among the requirements for security, speed, performance and price.

Computer virology is a dynamically developing contemporary branch of science, in which top achievements of mathematics, computer science, physics, chemistry, biology, genetics, etc., are combined. It is particularly important for guaranteeing the necessary information security in the contemporary society with its communication globalization and mobility.

References:

1. RISKS, Forum on Risks, http://catless.ncl.ac.uk/Risks
2. CERT, advisories, ftp://ftp.cert.org/pub/cert_advisories/
3. CIAC, Computer Incident Advisory Capability, http://www.ciac.org/
4. BUGTRAQ, forum, http://www.securityfocus.com/

EUGENE NICKOLOV is director of the National Laboratory of Computer Virology. Associate Professor at the Bulgarian Academy of Sciences. Holds a M.Sc. degree in Computer Science and a PhD degree (Optimizing Investigations on the Design of Computer Devices) from the Technical University of Sofia, Bulgaria. Member of a number of national and international unions and associations. Address: NLCV-BAS, Acad. G.Bontchev Str., Building 8, 1113 Sofia, tel.: (++359 2) 973 3398, E-mail: eugene@nlcv.bas.bg.