The mathematical model of cyber attacks on the critical information system

S S Kochedykov, E V Grechishnikov, A V Dushkin and D E Orlova
ET Department, Voronezh Institute Federal Penitentiary Service of Russia, 1a Irkutskaya Street, Voronezh, RU

E-mail: a_dushkin@mail.ru

Abstract. This article proposes a model of a system of destabilizing effects on the software of an automated system that describes the strategy and tactics of the actions of certain types of violations with the aim of realizing threats to information security through malicious software. Based on the theory of system analysis and the theory of complex systems modeling, an approach is proposed to describe the processes of destabilizing effects that allow to take into account the structure of software, to monitor its interface and to exert malicious software effects on it by limiting its availability and integrity while the software components of the software of automated systems. The materials of the article provide a formalized description of the structure of the software, its interface, malicious software actions that lead to destabilization and the emergence of information conflicts in the operation of automated systems. The presented model and its software implementation will make it possible to comprehensively evaluate the effectiveness of information protection systems for automated control systems for critical objects.

1. Introduction
The successful solution of the problem of ensuring the information security of automated control systems (ACS) by critical objects (CO) is based on ensuring the protection of information, primarily processes related to information processing, as they are the object of most security threats to many information systems [1-3]. At the same time, the confidential nature of the information circulating in such systems, and the increasing demands for their protection, reinforce this trend [4, 5].

The development of measures to protect information adequate to the threats of processed information in the ACS of the CO led to the development of appropriate mathematical models for the design and creation of tools and systems for protecting information from unauthorized access (SPIFUA) [6, 7].

This article proposes a model of the system of destabilizing influences (SDI) on the software of the ACS, which describes the strategy and tactics of the actions of certain types of violations with the aim of realizing threats to information security through malicious software actions (MSA).

The actions of the violator, depending on its potential in the implementation of threats to information security, provide for the identification and use of vulnerabilities in microprogram, system-wide and application software, network equipment used in the automated control system, as well as in the organization of information protection and ACS CO.
In determining how to implement threats to information security, it is necessary to take into account the fact that threats to information security can be realized directly through access to ACS CO software and / or information or indirectly through the creation of conditions and (or) means that provide such access, and also due to access or impact on the servicing infrastructure. In this case, the local goal of the violator who does not have access (access rights) to the ACS software and / or information is usually to gain access to the software (including through external public communication networks) and to obtain the maximum possible rights and privileges in such access.

Violators can commit actions that result in a violation of information security, deliberately (deliberate thunderstorms of information security) or accidentally (unintended thunderstorms of information security).

Intentional actions of violators can consist in the implementation of targeted or non-targeted threats to information security, which depend on the structural and functional characteristics and features of the ACS CO software.

2. The formulation of the problem
Let's consider the conditional project on creation of some ACS of the CO. We will proceed from the fact that the projected software ACS is a set of software components (SC), both developed by the project executors independently, and purchased from third-party manufacturers. Each of the SC is specified by a set of external system characteristics, namely: the functions to be performed, control commands, information and resource requirements [8-12].

The task is to develop such an SDI model that could describe the implementation of the partial threats to information security for the subsequent security assessment of the ACS CO software and the effectiveness of the applied ISS.

3. Functional typology and formalized description of the structure of the ACS software
The SC (software components) is a modular "unit" of the ACS software, which is replaced within the operating environment of the ACS CO software. Its internal components are hidden, but access to SC functions can be obtained by using one or several well-defined provided interfaces. The SC can also have the required interfaces required for stable (highly reliable) operation. The required interface determines which functions and services it requires from other SC to operate the ISS, which in turn control and control the operation of the SC ACS. By combining the provided and required interfaces of several PC, you can create a larger PC, integrated into the application software of the automated control system, which has the specified functionality. It can be said that all applied software of ACS, in fact, is a SC of a higher level.

The task will be to develop such a typology of mutual influence of the SC that are part of the ACS CO software, which can serve as a basis for solving the problem of stable (highly reliable) functioning of these systems in the environment of external and internal IS threats while ensuring the required quality level of the ACS CO software. As the basis for the description of the system of functioning of the software for automated control systems, let's take the work of Makarenko S.I, in which the approach to the development of a multilevel dynamic model of information conflict of the communication model is proposed, by analogy with his work [13].

In order to solve the problem, we introduce the following notation:

\( N \) – total number of SC (software components), which are part of the ACS CO software, including ISS (the means and systems of information protection) and SDI (system destructive impacts);

\( Thr^{(ext)} \) - external threats IS (the external security threats) or malware (the malicious software viruses) - deliberate exposures, representing some of the available functions of the SC or embedded in the software by attackers, implemented by SDI;

\( Thr^{(int)} \) - internal threats IS (the internal security threats) which are internal system conflicts between SC that have arisen unintentionally due to the fault of the designer or during the operation of the SC, as well as due to the impact of external IS threats, also implemented by the subsystems of SDI;
VUL^{(int)} - software vulnerability (the internal software vulnerabilities), characterizing the properties of the SC for their susceptibility, both external and internal threats, due to individual characteristics of the SC;

\[ L = \{l_1, \ldots, l_M \} \] - levels of SC functioning (the levels of functioning software components) in accordance with its interface (the interface software components) \( ISC \), where \( m = 1 \div M \) number of the level of functioning of software ACS CO;

\( ISC_l \) - SC interface on \( l \)-th level of functioning of the software;

\( ISC_{l,j} - i \) SC interface on \( l \)-th level;

\[ I_l = \bigcup_{m} \{ ISC_{l,j} \} \] - a set of SC interfaces used at the \( l \)-th level of the functioning of the software;

\[ I_{l,Th_{l}} = \bigcup_{m} \{ ISC_{l,j}, Thr^{(ex)}_l \mid e \in E_{l,Sc}^{(fun)} \} \] - a number of SC interfaces used at the \( l \)-th level of the functioning of software that is exposed to an external threat \( Thr^{(ex)} \) on a variety of parameters of the operating environment software (the operating environment software) \( E_{l,Sc}^{(fun)} \);

\[ I_{l,Vul_{l}} = \bigcup_{m} \{ ISC_{l,j}, Vul^{(int)}_l \mid e \in E_{l,Sc}^{(fun)} \} \] - a set of SC interfaces used at the \( l \)-th level of the functioning of software that is exposed to the internal threat \( Vul^{(int)}_l \) on a variety of parameters of the software operation environment \( E_{l,Sc}^{(fun)} \);

\[ I_{l,Sc} = \bigcup_{m} \{ ISC_{l,j}, Vul^{(int)}_l \mid e \in E_{l,Sc}^{(fun)} \} \] - a set of SC interfaces used at the \( l \)-th level of software functioning, which has internal software \( Vul^{(int)}_l \) vulnerabilities on a variety of parameters of the software environment \( E_{l,Sc}^{(fun)} \);

\[ I = \bigcup_{l} I_l \] - a number of SC interfaces used at all levels of the ACS CO software management system;

\( SC_{l,Sc}^{(fun)} \) - a set of parameters that characterize the functionality of the software of the ACS CO with regard to each SC (functions of the software component), communication between which is carried out through the interface of the SC \( ISC_l \) at the \( l \)-th level;

\( Q_{l,Sc} \) - a set of indicators of the quality control subsystem at the \( l \)-th level of operation of the software, which provides the \( ISC_l \)-th SC interface;

\( C_{l,Sc} \) - a functional specifying the set of control actions on the part of the control subsystem for the \( ISC_l \) interface of the SC at the \( l \)-th level of operation of the software at various times \( T \) in order to collect information on the quality of the work \( Q_{l,Sc} \);

\( E_{l,Sc}^{(fun)} \) - a set of parameters of the operating environment of the ACS software that define the parametric space for the \( ISC_l \)-th SC interface at the \( l \)-th level of functioning, taking into account the implementation of malware of deliberate and unintended external \( Thr^{(ex)}_l \) and internal \( Thr^{(int)}_l \) threats;

\( R_{l,Sc} \) - information and computing resource allocated for the implementation of the interface \( ISC_l \) at the \( l \)-th level of functioning.

4. Model of SDI on the ACS software

For the sake of convenience, visual perception is depicted in Fig. 1 structure of software ACS CO, taking into account controlled SDI parameters.
Figure 1. Structural diagram of software ACS CO, taking into account the controlled parameters of SDI and ISS

The monitoring channel (channel surveillance) behind the interface $Isc_l$ at $l$ the level of the software operation on the part of the attacker, implemented by SDI-1, functions in the same way as $Ch_{l,sc}^{(mon)}$ the monitoring channel for the output parameters of the software ASU ISS-1 control system (subsystems for controlling destabilizing and harmful effects):

$$Ch_{l,sc}^{(sur)} = \left\{ P_{l,sc}^{(out)} \times Q_{l,sc}^{(mon)} \right\},$$

where $P_{l,sc}^{(out)} \subseteq P_{l,sc}^{(out)}$-output parameters SC and $Q_{l,sc}^{(mon)} \subseteq Q_{l,sc}^{(out)}$-quality of service indicators that provide an interface $Isc_l$, as elements of sets $P_{l,sc}^{(out)}$ and $Q_{l,sc}^{(mon)}$ observed by an attacker's CDI-1.

At the same time $Ch_{l,sc}^{(sur)}$, there is one fundamental feature. The view that the ability of an attacker to monitor below the capabilities $Ch_{l,sc}^{(mon)}$ of the monitoring channel output parameters of the functioning of the software ACS, implemented ISS-1, the attacker can not get full access to all the observed parameters $P_{l,sc}^{(out)}$, and $Q_{l,sc}^{(mon)}$ vectors $Ch_{l,sc}^{(mon)}$. Thus, the number of interface $Isc_l$ parameters at the $l$-th level of software operation from the attacker's side through the channel $Ch_{l,sc}^{(sur)}$ is usually less than the number of parameters observed by the ISS-1, i.e.

$$Ch_{l,sc}^{(sur)} \leq Ch_{l,sc}^{(mon)} \leq \left\{ P_{l,sc}^{(out)} \times Q_{l,sc}^{(mon)} \right\},$$

while equality $Ch_{l,sc}^{(sur)} = Ch_{l,sc}^{(mon)}$ is ensured only in cases where the SDI-1 malefactor has full access to the ISS-1 surveillance channel (for example, by interception of the surveillance channel messages $Ch_{l,sc}^{(mon)}$).
The purpose of the surveillance $ChI_{l,Is_{c}}^{(sur)}$ is to determine $pos_{Is_{c}}$, the current state of the interface $Is_{c}$ at the $l$ level of operation of the software and to make decisions about the development of deliberate external $Thr_{l,Is_{c}}^{(ext)}$ and internal $Thr_{l,Is_{c}}^{(int)}$ effects on the interface $Is_{c}$.

$$pos_{Is_{c}} = con_{l,Is_{c}}\left(t_{0}, t, pos_{Is_{c}}(t_{0}), Pos_{l,Is_{c}}, E^{(int)}_{l,Is_{c}}, C_{l,Is_{c}}, A_{l,Is_{c}}\right),$$  

(3)

where $Pos_{l,Is_{c}} = \{pos_{Is_{c}}\}$ - a set of states $Is_{c}$ of the SC interface at the level of operation of the software, as well as $Pos_{l} = \{Pos_{l,Is_{c}}\} \cup S_{l}$ in accordance with the structure of the multi-functional links $S_{l} = \{(Is_{c}, Is_{c})\}$ between the SC interfaces $Is_{c} \in ISC$ at the $l$ level of the software functioning;

$t_{0}$ - the initial time of operation of the ACS software;

t - total operating time of the ACS software;

$pos_{Is_{c}}(t_{0})$ - the state of the SC interface at the initial time of the functioning of the ACS software;

$Pos_{l,Is_{c}}$ - set of states of the $Is_{c}$-th SC interface at the $l$-m level of the functioning of the software;

$E^{(int)}_{l,Is_{c}} = \left\{p_{l,Is_{c}} \times R_{l,Is_{c}} \times Thr_{l,Is_{c}}^{(ext)} \times Thr_{l,Is_{c}}^{(int)}\right\}$ - a set of parameters of the operating environment of the ACS software that define the parametric space for the $Is_{c}$-th SC interface at the $l$ level of the software functioning $R_{l,Is_{c}}$, taking into account the resources allocated to the interface, $p_{l,Is_{c}} = \left\{pie_{l,Is_{c}}(Vu_{l,Is_{c}}^{(int)})\right\}$ - the parameters of the initial operating environment (without considering the impact of malware, but having a vulnerability in the software $Vu_{l,Is_{c}}^{(int)}$) malware when they realize intentional and unintended external $Thr_{l,Is_{c}}^{(ext)}$ and internal threats $Thr_{l,Is_{c}}^{(int)}$;

$C_{l,Is_{c}}$ - a functional specifying the set of control actions on the part of the control subsystem for the $Is_{c}$-th interface of the SC at the $l$ level of operation of the software at various times $T$ in order to collect information on the quality of the work $Q_{l,Is_{c}}$;

$A_{l,Is_{c}}$ - a lot of algorithms in the interface $Is_{c}$ at the $l$ level of software functioning.

Taking into account expression (1), it can be concluded that the SDI-1 always observes the interface $Is_{c}$ with a lesser degree of reliability than the ISS-1.

When studying the software of the ACS of the CO by an attacker at the first stage, it is necessary to solve the problem of identifying a set of private interfaces $\{Is_{c}\}$ used at the $l$ level among a set of $ ISC_{l}$ monitoring interfaces $ChI_{l,Is_{c}}^{(sur)}$. As a rule, this problem is solved on the basis of the theory of system analysis with subsequent identification and classification [14-16]. At the second stage, the task of determining $pos_{Is_{c}}$ the state for each specific interface is solved $Is_{c}$. Both tasks can be solved using a simulation model that takes into account the special features of the implementation of the SDI-1, by creating a character space and constructing observation trajectories $\{chI_{l,Is_{c}}(t)\} \in ChI_{l,Is_{c}}^{(sur)}$ in it during $t^{o\text{bser}}$ or the time of observation sufficient to accumulate data and identify specific interfaces $\{Is_{c}\} \in I_{l}$ and their states $\{pos_{Is_{c}}\}$.

The main problematic aspects associated with the monitoring channel of the interface from the attacker are:

- the presence of a time delay associated with the accumulation of observational data during the course of observation;

- observation of the group of interfaces $I_{l}$-of the $l$ level by their general quality indicators $Q_{l}$ and output parameters $P_{l}^{(out)}$ that are common for the whole group:
\[
I'_t = \bigcup \left( \text{Isc}_j \left| \bigcap_{l} Q_{l,\text{Isc},\text{Ch}_1}^{(\text{sur})} = Q_{l,\text{Ch}_1}^{(\text{sur})} \neq \emptyset \right. \wedge \bigcap_{l} \mathcal{P}_{l,\text{Isc},\text{Ch}_1}^{(\text{out})} = \mathcal{P}_{l,\text{Ch}_1}^{(\text{out})} \neq \emptyset \right) \),
\]

\[
\text{Ch}_1^{(\text{sur})} = \left\{ Q_{l,\text{Ch}_1}^{(\text{sur})} \times \mathcal{P}_{l,\text{Ch}_1}^{(\text{out})} \right\} = \left( \bigcap_{l} Q_{l,\text{Isc},\text{Ch}_1}^{(\text{sur})} \right) \bigcup \left( \bigcap_{l} \mathcal{P}_{l,\text{Isc},\text{Ch}_1}^{(\text{out})} \right)
\]

which significantly complicates the identification of the state \( \text{pos}_{\text{Isc}} \) of each of the interfaces included \( \text{Isc}_j \in I'_t \) in the group:

- the complexity of reliable determination of the interface \( \text{pos}_{\text{Isc}} \) state under the conditions of parametric constriction:

\[
\{ \text{pos}_{\text{Isc}} \} \rightarrow \{ \text{Isc}_j \} \rightarrow I_t \rightarrow \{ \mathcal{P}_{l,\text{Isc},\text{Ch}_1}^{(\text{out})} \times Q_{l,\text{Ch}_1}^{(\text{sur})} \} \rightarrow \{ \mathcal{P}_{l,\text{Isc},\text{Ch}_1}^{(\text{out})} \times Q_{l,\text{Ch}_1}^{(\text{sur})} \} \rightarrow \text{Ch}_1^{(\text{sur})} \rightarrow \text{Ch}_1^{(\text{sur})}
\]

and the narrowing of the reaction associated with the discreteness constraints \( T_{\text{ch}}^{(\text{sur})} = \{ t_1, t_2, \ldots, t_n \} \) and the total time of observation \( t_{\text{obs}}^{(\text{sur})} \):

- the ability of the ISS to actively counter the surveillance channel of an attacker, by implementing measures to counter act malware \( (\text{ch}_1^{(\text{sur})} = 0) \), as well as creating false SC and software systems that implement an intruder’s misinformation \( \text{Ch}_1^{(\text{sur})} \) on the true operating modes and about the interfaces used in the ACS software of the CO;

- presence of requirements for stealth surveillance.

To increase the level of reliability of the observation of SDI-1, test actions \( \text{Ch}_1^{(\text{sur})} \) can be issued that correspond to the uniquely identifiable trajectory of the parameters change \( Q_{l,\text{Isc},\text{Ch}_1}^{(\text{sur})} (t) \) and \( \mathcal{P}_{l,\text{Isc},\text{Ch}_1}^{(\text{out})} (t) \) observed in the channel \( \text{Ch}_1^{(\text{sur})} \). The uncompromising use of such test actions \( \text{Ch}_2^{(\text{sur})} \), selected for the identification of interfaces \( \text{Isc}_j \), as well as their states \( \{ \text{pos}_{\text{Isc}} \} \) at the initial stage of the conflict, will allow gathering information on the composition and functional structure of the \( l \) level, and at the stages of conflict development, control the effectiveness of the impact.

5. The model of malicious software actions on the part of SDI-1 through the channel \( \text{Ch}_2^{(\text{sur})} \)

Based on the results of the observation \( \text{Ch}_1^{(\text{sur})} \) of SDI-1 in the decision making subsystem, based on the available algorithms of the attacker, depending on the method and method of harmful impact, SDI-1 decides on the strategy and tactics for the implementation of harmful effects \( \text{Ch}_2^{(\text{sur})} = \{ \text{ch}_2^{(\text{sur})} (\text{Thr}_1^{(\text{ext})} \times \text{Thr}_1^{(\text{int})}) \} \) from a variety of impacts \( \text{Ch}_2^{(\text{sur})} \) that can be implemented at the implementation of malware of deliberate and unintended external \( \text{Thr}_1^{(\text{ext})} \) and internal \( \text{Thr}_1^{(\text{int})} \) threats.

The choice of impact \( \text{Ch}_2^{(\text{sur})} \) must take into account the set of factors of the initial environment of the functioning of the ACS \( \text{Pie}_{l,\text{Isc}}^{(\text{sur})} \) software, the resource allocated to the interface \( R_{l,\text{Isc}} \), as well as the composition of the interface \( \text{Isc}_j \) support by algorithms \( \text{A}_{l,\text{Isc}} \) and its parameters \( \text{Pa}_{l,\text{Isc}} \). In fact, the aggregate of the interface \( \text{Isc}_j \) and the SDI-1 form, as it were, another control loop \( \text{C}_{l,\text{Isc}}^{(\text{sur})} \) with an antagonistic criterion of achieving the goal of the impact-reducing the specified or all the quality indicators of the interface \( q_{l,\text{Isc}} \) below the required values \( q_{l,\text{Isc}}^{(\text{req})} \) .
A finite set of impacts \( \left\{ c_{l,lc}^2(\text{sur}) \right\} \) on the time interval \( T_{\text{sur}} \) form a plan of harmful effects.

The plan of harmful effects can be realized:
- in discrete time
  \[
  Ch_{l,lc}^{(\text{sur})} = \left\{ Ch_{l,lc}^{(\text{sur})}(t_1), Ch_{l,lc}^{(\text{sur})}(t_2), \ldots, Ch_{l,lc}^{(\text{sur})}(t_n), \ldots \right\}, \quad T_{\text{sur}} = \{ t_1, t_2, \ldots, t_n, \ldots \};
  \]
- in continuous time
  \[
  Ch_{l,lc}^{(\text{sur})}(t) = Ch_{l,lc}^{(\text{sur})}(t), \quad t \in T_{\text{sur}}^{(\text{sur})}.
  \]

The implementation plan for malware \( Ch_{l,lc}^{(\text{sur})} \) for any initial state of the interface \( pos_{lc}(t_0) \) and the presence of control actions \( C_{l,lc} \) should ensure that criterion (6) is met, i.e. existence of a limit
\[
\lim_{t \to T_{\text{sur}}^{(\text{sur})}} f_{l,lc}(t, pos_{lc}(t), \text{Pie}_{l,lc}^{(\text{sur})}, C_{l,lc}, A_{l,lc})
\]
\[
\exists \left\{ q_{l,lc}^\text{req} < q_{l,lc}^\text{req} \right\} \in Q_{l,lc}
\]
\[
\text{Pie}_{l,lc}^{(\text{sur})} = \left\{ R_{l,lc} \times \text{pie}_{l,lc}^{(\text{sur})} \times Ch_{l,lc}^{(\text{sur})} \right\}
\]
\[
C_{l,lc} = \left\{ \left\{ c_{l,lc} \right\} \times T_{\text{sur}}^{(\text{sur})} \right\}
\]

The presence of the malware \( Ch_{l,lc}^{(\text{sur})} \) plan should provide for the objective (the mission) \( Ch_{l,lc}^{(\text{sur})}(t_{s+1}) \) to ensure that the criterion (6) is met at a finite number of steps \( n, \) at future times.

The impact plan should provide for the formation of target impacts \( Ch_{l,lc}^{(\text{sur})}(t_{s+1}) \) depending on the current observation \( Ch_{l,lc}^{(\text{sur})}(t_s) \) and obey a certain strategy (the strategy) associated with achieving the goal of the impact, provided \( \exists \left\{ q_{l,lc} < q_{l,lc}^{\text{req}} \right\} \). With reference to SDI-1, consider the following main strategies for influencing a single interface \( \text{Isc}_l \), assuming that decision-making on impact \( Ch_{l,lc}^{(\text{sur})}(t_{s+1}) \) generation is carried out at each step irrespective of previous impacts and is based on observation results \( Ch_{l,lc}^{(\text{sur})}(t_s) \).

The tactic of the impact \( g \) will consist in the allocation of individual objects to which the malware is planned in the next step \( t_{s+1} \). The result of the implementation of the threat to information security is determined by the impact of the threat on each property of information security (confidentiality, integrity, availability).

### 6. Computational experiment

Investigation of the effectiveness of counteracting malware on SC ACS by various types of ISS-1 and ISS-2 is proposed to be carried out by methods of computational experiments [17] using queuing theory and algorithms considered in the thesis [6].

The simulation model implements a maintenance scheme that provides a serial, in accordance with the algorithm of functioning of the ACS software, the simulation of information processing processes and its protection of various types of ISS under malware conditions.
The input streams of information $\{ q^{(i)}_g, g = 1, 2, ..., G \}$ and service $\{ q^{(s)}_g, g = 1, 2, ..., G \}$ applications, as well as threats to MSA, to information processing processes in the ACS of the CO $\{ q^{(de)}_s, s = 1, 2, ..., S \}$ over time $A_n$ are given by intensities $\lambda^{(i)}_1, \lambda^{(s)}_1$ and $\lambda^{(de)}_1$, accordingly, are formed under the assumption that the simulated processes are stationary $A_n$, within the limits of ordinary flows and the absence of aftereffects in them [18].

According to this maintenance scheme, each $i$-th $i = 1, 2, \ldots I_j$, $j$-th $j = 1, 2, \ldots J$ function, of the task of processing information or its protection, for example, ISS-1 of the assigned type, is presented as a separate maintenance phase. In this case, each phase is represented by the nine states of the ACS of the CO:

- Incorrect operation
- Correct operation
- Input information
- External threats
- Control Vector
- Status of malfunction
- Status of correctness identification
- Status of identification of traces of threats
- Status of identification of the attacker
- Status of identification of the consequences of threats
- Status of restoring correctness of information processes

The graph of the states of the maintenance scheme of the simulation model is shown in Figure 2:

In the presence of the effect of a malicious program on the process of processing information SC, SSI-1 with probability $P_{m}^{(i)}(t_m^{(i)})$ and ISS-2 with probability $P_{m}^{(s)}(t_m^{(s)})$ is controlled the correctness of...
information processing (the probability of detection malicious activity) received in the state \(C_m^{(sic)}\). In the absence of signs of incorrect functioning of the PC, the next service phase is carried out according to the information processing algorithm. In this case, the \(m\)-th service phase goes to the next state \(C_{m+1}^{(sic)}\). If signs of incorrect functioning are detected, then according to the expression:

\[
P^{(dna)} = 1 - \left( 1 - P_m^{(sic)} \right) \prod_{f \text{-index} \{P^{(f)}\}} P_f^{(sic)}
\]

The probability of identification of the malware threat is determined, where \(F^{(y)}\) - the list of maintenance phases in which the mean of incorrect operation of the corresponding SC are detected by means of ISS-1 and ISS-2.

The fact that the \(i = 1, 2, \ldots, I\) particular problems of information protection solved by the type-dependent ISS-1 are dependent, as an indicator of effectiveness (the effectiveness of information security), use the expression:

\[
E^{(SCS-1)} = \prod_{i=1}^{I} E_i^{(SCS-1)}.
\]

Where \(E_i^{(SCS-1)}\) is the efficiency of solving the particular problem of information protection performed by ISS-1, which is the probability \(P^{(sic)}\) of an event \(A^{(sic)}\) consisting in the prevention of malware threats, i.e. \(E_i^{(SCS-1)} = P_i^{(sic)}\). At the same time, the probability \(P^{(sic)}\) can be determined in terms of probability \(P^{(dna)}\) by the formula: \(P^{(sic)} = 1 - P^{(dna)}\).

As a private indicator of the effectiveness of counteracting ISS-2 malware threats of an embedded type, it is reasonable to use the probability of meeting the condition:

\[
\tau^{(rscip)} \leq \tau^{(rscip)\text{-required}},
\]

i.e.

\[
E^{(SCS-2)} = P(\tau^{(rscip)} \leq \tau^{(rscip)\text{-required}})
\]

The effectiveness of countering threats to malware on the SCACS by means of information protection both attached and embedded type is expedient to determine us in generalized indicator of the form:

\[
E^{(protection)} = 1 - (1 - E^{(SCS-1)}) \times (1 - E^{(SCS-2)}).
\]

7. Analysis of experimental results

Based on the results of the simulation, we will analyze the functioning of the automated control system in order to determine the best strategy to counter threats to information security to the SC ACS CO.

Statistical sampling of the results of the assessment of the effectiveness of identification of the malware threat on the ACS of the CO with data protection means of various types is presented in Table 1.

| ISS-1 attached type | ISS-2 built-in type | ISS of two types simultaneously |
|---------------------|---------------------|--------------------------------|
| \(E^{(SCS-1)} = 0,265;\) | \(E^{(SCS-2)} = 0,688;\) | \(E^{(protection)} = 0,773;\) |
In the course of the experiment, it was established that the complex use of different types of ISS makes it possible to increase the information protection efficiency from 60% to 75% in comparison with the use of only the ISS type of the type to be delivered and to increase the protection efficiency from 15% to 30% in comparison with the use of only the ISS of the embedded type. As a result of the implementation of the simulation model, a fact was established that confirms that the lack of the use of the ISS reduces the timeliness of the implementation of information processing tasks in the ACS from 60% to 80%. The use of the ISS type of the type being assigned increases the timeliness of information processing in the ACS from 25% to 50%. The use of an ISS embedded type improves the timeliness of the implementation of information processing tasks in the ACS from 45% to 60%. The complex use of different types of ISS within the framework of the method proposed in the dissertation [6] increases the timeliness of the implementation of information processing tasks in the ACS from 35% to 65%. The graphic interpretation of the calculation is shown in figure 3.
Figure 3. Graphical interpretation of the experiments on the evaluation of the malware counteraction effectiveness index

8. Conclusion

Thus, the presented model of SDI can be described as a dynamic system functioning in a complex parametric environment, while the interface of the SC itself is actually part of two control loops - ISS-1 control and monitoring of the qualitative and safe functioning of the SC, responsible for ensuring the required level of security in the context of the impact on the ACS of external and internal threats of IS and SDI-I of an impact management system with antagonistic performance objectives. The model can be used as a test model for modeling security threats in assessing the effectiveness of the subsystem for monitoring and detecting MSA of a structurally-dependent IS of an embedded type that ensures the integrity control of software ACS CO in conflict with hackers, insiders, phreakers, crackers and other computer crime subjects [19].

The proposed mathematical model of SDI can be used to develop new strategies for managing the operation of ACS CO software in the context of the complex impact of malicious software tools for computer reconnaissance.

References

[1] Kochedykov S S, Ilyin A A, Potanin V E and Bataronov I L 2001 Features of the functioning of information and telecommunication systems in the presence of threats to the safety of processed information Information and Security I pp 63-68

[2] Zima V, Moldavian A and Moldavian N 2000 Security of global network technologies (BHV-Petersburg) p 320

[3] Common Criteria for Information Technology Security Evaluation. National Institute of Standards and Technology & National Security Agency (USA)1999 Communication Security Establishment (Canada), Communications-Electronics Security Group (United Kingdom), Bundesamt fur Sichereit in der Informationstechnik (Germany), Service Central de la Securite des Systemes d'Information (France), National Communications Security Agency (Netherlands)

[4] Requirements to ensure the protection of information in automated control systems for production and technological processes in critical facilities, potentially hazardous facilities, as well as objects that present a heightened danger to life and health of people and the
environment (approved by order of the Federal Service for Technical Regulation of the Russian Federation on 14.03.2014.)

[5] General requirements for ensuring information security in key information infrastructure systems (approved by order of the Federal Service for Technical Regulation of the Russian Federation on May 18.05.2007).

[6] Kochedykov S S 2002 Mathematical model of counteraction of NDT to ITCS by various types of GIS in conditions of minimizing the computational resource abstraction: dis.candidate technical sciences (Voronezh) p 165

[7] Gasser M 1988 Building a secure computer system p 253

[8] Kochedykov S S, Kravchenko A S, Rodin S V, Perminov G V and Dushkin A V 2015 On the issue of the information structure of the process of distribution of a temporary resource between different types of protection against unauthorized access Basic research 5-1 pp 89-93

[9] BalanV P, Dushkin A V, Novoseltsev V I and Sumin V I 2015 Conflict Management (Hot line – Telecom) p 144

[10] Barkalov S A, Dushkin A V, Kolodyazhny S A, Sumin V I and Novoseltsev V I 2017 Introduction to system analysis (Moscow: Hot line – Telecom) p 234

[11] Dushkin A V, Novoseltsev V I and Sumin V I 2015 Modeling of management systems and information technology (Hot line – Telecom) p 192

[12] Dushkin A V, Kravchenko A S, Novoseltsev V I, Smolentseva T E, Sumin V I 2014 Mathematical models and information processes of complex object management: Monograph (Scientific Book) p 125

[13] Makarenko S I 2015 Dynamic model of the communication system in the conditions of a functional-different-level information conflict of observation and suppression Control, communication and safety systems 3 pp 122-185

[14] Kochedykov S S, Novoseltsev V I, Noev A N, Orlova D E 2017 Deterministic methods for diagnosing crisis situations in automated critical systems Bulletin of the Voronezh Institute of the Ministry of Internal Affairs of Russia 1 pp 86-89.

[15] Sumin V I, Dushkin A V, Belokurov S V and Kravchenko A S 2014 Modeling systems (Scientific Book) p 222

[16] Novoseltsev V I, Kochedykov S S and Orlova D E 2017 Tensor analysis of Crohn and its applications: Monograph (Scientific Book) p 260

[17] Yakovlev S A 1986 Planning problems of simulation experiments in the design of information systems (Leningrad) p 254

[18] Eglshart D L and Scheldler D S 1984 Regenerative modeling of queuing networks (Radio and communication) p 136

[19] Sushkov P F, Kochedykov S S, Kiselev V V and Artemov A A 2001 The methodology of conducting primary investigative actions in the investigation of high-tech crimes Bulletin of the Voronezh Institute of the Ministry of Internal Affairs of Russia 2(9) pp 152-155