УДК 004.056 Дата подачи статьи: 02.10.16
DOI: 10.15827/0236-235X.118.314-319 2017. Т. 30. № 2. С. 314–319

МЕТОД ФОРМИРОВАНИЯ МНОЖЕСТВ АЛЬТЕРНАТИВНЫХ ВАРИАНТОВ ПОСТРОЕНИЯ ПОДСИСТЕМ, ВХОДЯЩИХ В СОСТАВ СИСТЕМЫ ЗАЩИТЫ ОТ КОМПЬЮТЕРНЫХ АТАК

Е.Б. Дроботун, к.т.н., докторант, drobotun@xakep.ru;
Е.П. Угловский, начальник лаборатории;
И.Ш. Замалтдинов, к.т.н., старший научный сотрудник
(Военная академия воздушно-космической обороны им. Маршала Советского Союза Г.К. Жукова, ул. Жигарева, 50, г. Тверь,170100, Россия)

Построение рациональной системы защиты от компьютерных атак для информационно-вычислительной или автоматизированной системы предполагает формирование множества конфигураций системы защиты, состоящей, в свою очередь, из множества отдельных программных и программно-аппаратных компонентов, и дальнейший выбор из сформированного множества рационального варианта построения системы защиты от компьютерных атак по определенным критериям. При формировании данного множества, помимо соответствия системы защиты необходимым функциональным требованиям, следует учитывать как параметры самой защищаемой системы (ее структуру и многоуровневость построения), так и программную и аппаратную совместимость компонентов между собой, а также совместимость компонентов с программно-аппаратной платформой, на базе которой построена защищаемая система.

В статье представлен один из подходов к формированию множества возможных вариантов построения системы защиты от компьютерных атак с учетом ее декомпозиции на три подсистемы: подсистему обнаружения компьютерных атак, подсистему противодействия компьютерным атакам и подсистему устранения последствий применения компьютерных атак.

Ключевые слова: *автоматизированная система управления, компьютерная атака, информационная безопасность, проектирование системы защиты, защита от компьютерных атак.*

Решение задачи структурно-параметрического синтеза системы защиты информационно-вычислительных и автоматизированных систем различного назначения от компьютерных атак заключается в выборе наилучших вариантов системы защиты в условиях ограничений и предполагает наличие множества альтернативных вариантов ее построения.

С целью уменьшения количества вариантов, упрощения их формирования и проведения расчетов показателей, характеризующих эти варианты, необходимо осуществить декомпозицию задачи выбора рационального варианта построения системы защиты от компьютерных атак. Как показано в [1], декомпозицию целесообразно осуществлять как по последовательности выполнения процедур формирования и выбора вариантов, так и по подсистемам системы защиты информационно-вычислительных и автоматизированных систем от компьютерных атак.

В целом систему защиты от компьютерных атак для информационно-вычислительных и автоматизированных систем можно декомпозировать на три подсистемы: подсистему обнаружения компьютерных атак, подсистему противодействия компьютерным атакам и подсистему устранения последствий применения компьютерных атак. При этом при декомпозиции по последовательности выполнения процедур формирования и выбора вариантов формирование рационального варианта построения системы защиты в целом необходимо начать с формирования множества альтернативных вариантов построения подсистем, входящих в систему защиты от компьютерных атак.

Формирование множества альтернативных вариантов построения подсистем, входящих в систему защиты от компьютерных атак, осуществляется в два этапа:

– формирование полного множества вариантов построения подсистем, входящих в систему защиты от компьютерных атак;

– отбор из сформированного множества вариантов построения подсистем технически реализуемых для конкретной конфигурации защищаемой информационно-вычислительной или автоматизированной системы.

Формирование полного множества вариантов построения подсистем, входящих в систему защиты от компьютерных атак

Полное множество вариантов построения каждой из подсистем, входящих в систему защиты от компьютерных атак, для информационно-вычислительных и автоматизированных систем $V = \{V_{обн}, V_{пр}, V_{устр}\}$ целесообразно формировать путем решения комбинаторной задачи перебора всех возможных компонентов, которые могут входить в состав каждой из подсистем, и задач, возложенных на подсистему, с учетом многоуровневости построения защищаемой информационно-вычислительной или автоматизированной системы [2] (например, уровень представления информации (операторский),

уровни обработки информации и ввода/вывода информации) (рис. 1).

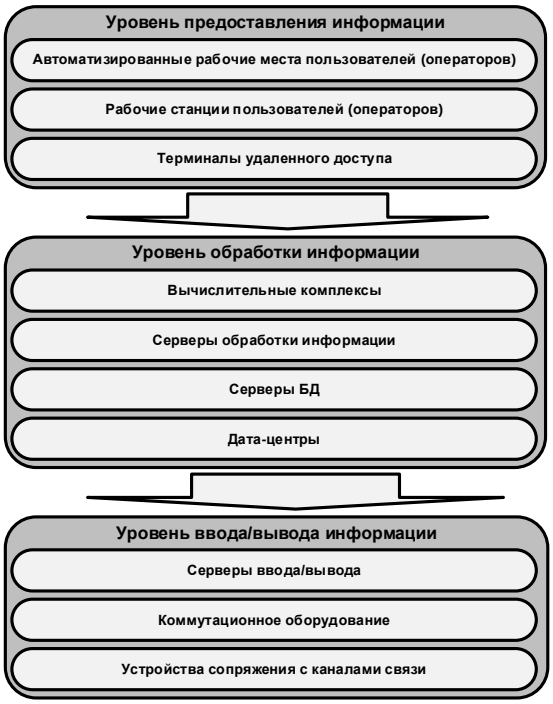

Рис. 1. Пример трехуровневой структуры автоматизированной системы управления

Fig. 1. The example of three-level structure of an automated control system

Тогда формирование множеств вариантов построения подсистем, входящих в систему защиты от компьютерных атак, будет включать:

– формирование множества вариантов построения подсистемы для каждого из уровней с учетом задач, возложенных на подсистемы для каждого уровня защищаемой информационно-вычислительной или автоматизированной системы;

– формирование множества вариантов построения подсистемы для всех уровней защищаемой информационно-вычислительной или автоматизированной системы в целом из множеств, полученных на предыдущем этапе.

Формирование множества вариантов построения подсистемы, входящей в состав системы защиты от компьютерных атак, для каждого уровня защищаемой системы. Необходимость отдельного формирования множества возможных вариантов построения подсистем системы защиты от компьютерных атак для каждого уровня построения защищаемой системы возникает вследствие, во-первых, разных функциональных требований к каждой из подсистем для каждого уровня построения защищаемой информационно-вычислительной или автоматизированной системы, а во-вторых, построения каждого уровня защищаемой системы на разных программно-аппаратных платформах и, соответственно, с разным составом компонентов средств защиты для каждой из подсистем. Исходя из этого множество всех возможных вариантов V построения подсистем, входящих в состав системы защиты от компьютерных атак, будет выглядеть следующим образом: $V = \{\{V_{обн1}, V_{обн2}, \ldots V_{обнl}\}, \{V_{пр1}, V_{пр2}, \ldots, V_{прl}\}, \{V_{устр1}, V_{устр2}, \ldots, V_{устрl}\}\}$, где l – количество уровней построения защищаемой системы.

В общем, формирование множества вариантов построения подсистемы для одного уровня построения защищаемой информационно-вычислительной (автоматизированной) системы осуществляется в несколько этапов.

Этап 1. Построение таблицы выполнения требований каждым компонентом средств защиты, входящим в подсистему, для рассматриваемого уровня (табл. 1). В данной таблице отражены функциональные требования ($f_j \in F_k$, $j = \overline{1, J}$, где J – число функциональных требований) для рассматриваемой k-й подсистемы соответствующего уровня к отдельным компонентам ($s_i \in S_k$, $i = \overline{1, I}$, где I – число компонентов средств защиты), входящим в состав рассматриваемой k-й подсистемы соответствующего уровня.

Элементы данной таблицы формируются с учетом следующего условия:

$$v_{ij} = \begin{cases} 1, & \text{если } i\text{-й компонент удовлетворяет } j\text{-му требованию,} \\ 0, & \text{если } i\text{-й компонент не удовлетворяет } j\text{-му требованию.} \end{cases}$$

Таблица 1

Анализ выполнения требований каждым компонентом средств защиты для каждого уровня защищаемой системы

Table 1

The analysis of compliance with requirements by each component of protection means for each level of a protected system

Отдельные компоненты	Функциональные требования					
	f_1	f_2	f_3	f_4	…	f_J
s_1	v_{11}	v_{12}	v_{13}	v_{14}	…	v_{1J}
s_2	v_{21}	v_{22}	v_{23}	v_{24}	…	v_{2J}
s_3	v_{31}	v_{32}	v_{33}	v_{34}	…	v_{3J}
s_4	v_{41}	v_{42}	v_{43}	v_{44}	…	v_{4J}
…	…	…	…	…	…	…
s_I	v_{I1}	v_{I2}	v_{I3}	v_{I4}	…	v_{IJ}

При этом критерии для каждой подсистемы определяются отдельно и, к примеру, согласно [3–8], могут включать в себя функциональные требования

– к идентификации и аутентификации субъектов доступа к объектам доступа;

– к управлению доступом субъектов доступа к объектам доступа;

– к ограничению программной среды;

– к защите машинных носителей информации;

Рис. 2. Пример определения функциональных требований для отдельных подсистем, входящих в состав системы защиты от компьютерных атак

Fig. 2. The example of defining functional requirements for separate subsystems which are a part of a computer attack protection system

– к регистрации событий безопасности;

– к антивирусной защите информации;

– к обнаружению (предотвращению) вторжений;

– к контролю (анализу) защищенности информации;

– к обеспечению целостности информационно-вычислительной (автоматизированной) системы и информации;

– к обеспечению доступности информации.

Возможный вариант определения требований для отдельных подсистем, входящих в состав системы защиты от компьютерных атак, показан на рисунке 2.

Этап 2. Проверка всей совокупности компонентов на выполнение функциональных требований. Данный этап реализуется следующим образом (рис. 3):

– осуществляется поэлементная дизъюнкция каждой строки из полученной на первом этапе таблицы: $\{v_1, v_2, v_3, \ldots, v_J\} = \{v_{11}, v_{12}, v_{13}, \ldots, v_{1J}\} \vee \vee \{v_{21}, v_{22}, v_{23}, \ldots, v_{2J}\} \vee \ldots \vee\} = \{v_{I1}, v_{I2}, v_{I3}, \ldots, v_{1J}\}$;

– выполняется конъюнкция всех элементов получившегося множества $\{v_1, v_2, v_3, \ldots, v_J\}$: $v = v_1 \& v_2 \& v_3 \& \ldots \& v_J$;

– проверяется условие равенства получившегося результата (v) единице: если данное условие выполняется ($v = 1$), вся совокупность компонентов средств защиты подсистемы удовлетворяет функциональным требованиям; в противном случае ($v \neq 1$) компонентов средств защиты, из которых может быть построена подсистема, недостаточно, и поэтому необходимо расширять весь возможный спектр применяемых компонентов средств защиты для рассматриваемой подсистемы соответствующего уровня.

Этап 3. Выбор вариантов построения подсистемы, входящей в состав системы защиты от компьютерных атак, для соответствующего уровня построения защищаемой информационно-вычислительной или автоматизированной системы. Суть данного этапа в отборе конфигураций компонентов подсистемы, которые позволят реализовать все функциональные требования к подсистеме.

В таблице 2 показан пример анализа выполнения функциональных требований к одной из подсистем системы защиты.

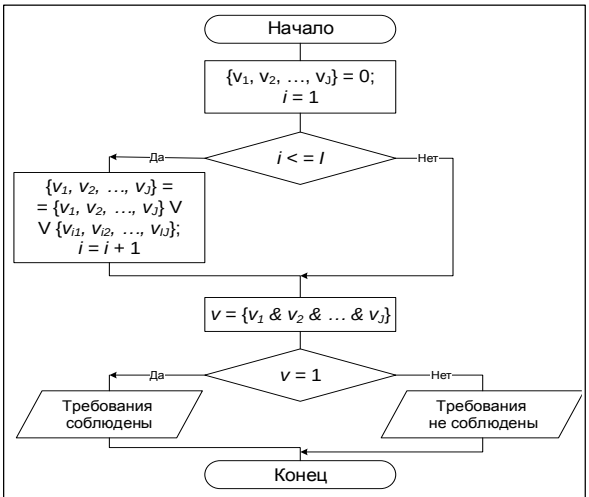

Рис. 3. Алгоритм проверки соответствия всей совокупности компонентов средств защиты функциональным требованиям

Fig. 3. Algorithm of checking the compliance with functional requirements of all set protection means

Таблица 2

Пример анализа выполнения функциональных требований к одной из подсистем системы защиты для отдельного уровня защищаемой системы

Table 2

The example of the analysis of implementing functional requirements to one of protection system subsystems for a separate level of the protected system

Отдельные компоненты, с помощью которых возможна реализация k-й подсистемы S_k	Функциональные требования					
	f_1	f_2	f_3	f_4	f_5	f_6
s_1	1	1	0	1	0	0
s_2	1	0	1	0	1	0
s_3	0	1	0	0	0	1
s_4	1	0	0	0	0	1
s_5	0	1	1	1	0	1

Для данного примера множество всех возможных компонентов, с помощью которых возможна реализация k-й подсистемы, включает в себя пять элементов: $S_k = \{s_1, s_2, s_3, s_4, s_5\}$.

Функциональные требования в данном случае заданы множеством из шести элементов: $F_k = \{f_1, f_2, f_3, f_4, f_5, f_6\}$.

Отбор возможных конфигураций компонентов k-й подсистемы для соответствующего уровня, которые позволят реализовать все функциональные требования к данной подсистеме, осуществляется следующим образом:

– формируются все возможные варианты комбинаций компонентов, из которых возможно построение k-й подсистемы для рассматриваемого уровня защищаемой подсистемы; при этом общее количество всех возможных вариантов $N=2^m-1$, где m – количество всех компонентов, из которых возможно построение k-й подсистемы;

– проверяется выполнение функциональных требований каждого варианта комбинаций компонентов, из которых возможно построение k-й подсистемы для рассматриваемого уровня защищаемой системы, путем поэлементной дизъюнкции строк, относящихся к компонентам, входящим в анализируемый вариант, далее выполняются поэлементная конъюнкция получившегося множества и проверка выполнения условия равенства полученного результата единице (рис. 4); при этом для каждого элемента строки осуществляется операция конъюнкции с числом t, которое получается путем сдвига вправо на один разряд значения n (операция shr(n)) и логического умножения результата этой операции на 1.

В качестве примера можно рассмотреть проверку на выполнение функциональных требований нескольких вариантов конфигурации применительно к таблице 2.

Пример 1. Конфигурация в составе $\{s_1, s_2, s_3, s_4\}$. Производим поэлементную дизъюнкцию строк, относящихся к компонентам s_1, s_2, s_3 и s_4: $v = \{1, 1, 0, 1, 0, 0\} \vee \{1, 0, 1, 0, 1, 0\} \vee \{0, 1, 0, 0, 0, 1\} \vee \{1, 0, 0, 0, 0, 1\} = \{1, 1, 1, 1, 1, 1\}$.

Осуществляем поэлементную конъюнкцию получившегося множества v и получаем единицу. Таким образом, данная конфигурация позволяет выполнить все функциональные требования, предъявленные к анализируемой подсистеме.

Пример 2. Конфигурация в составе $\{s_1, s_3, s_4, s_5\}$. Производим поэлементную конъюнкцию строк, относящихся к компонентам s_1, s_3, s_4 и s_5: $v = \{1, 1, 0, 1, 0, 0\} \vee \{0, 1, 0, 0, 0, 1\} \vee \{1, 0, 0, 0, 0, 1\} \vee \{0, 1, 1, 1, 0, 1\} = \{1, 1, 1, 1, 0, 1\}$.

Осуществляем поэлементную конъюнкцию получившегося множества v и получаем ноль. Таким образом, данная конфигурация не позволяет выполнить все функциональные требования, предъявленные к анализируемой подсистеме.

Для случая, приведенного в таблице 2, возможны 11 вариантов построения k-й подсистемы (табл. 3).

Таблица 3

Пример вариантов построения k-й подсистемы для отдельного уровня защищаемой системы

Table 3

The example of k-th subsystem constructing variants for a particular level of the protected system

Вариант построения k-й подсистемы	Компоненты, входящие в вариант построения k-й подсистемы
S_{k1}	$\{s_1, s_2, s_3\}$
S_{k2}	$\{s_1, s_2, s_4\}$
S_{k3}	$\{s_1, s_2, s_3, s_4\}$
S_{k4}	$\{s_2, s_5\}$
S_{k5}	$\{s_1, s_2, s_5\}$
S_{k6}	$\{s_2, s_3, s_5\}$
S_{k7}	$\{s_1, s_2, s_3, s_5\}$
S_{k8}	$\{s_2, s_4, s_5\}$
S_{k9}	$\{s_1, s_2, s_4, s_5\}$
S_{k10}	$\{s_2, s_3, s_4, s_5\}$
S_{k11}	$\{s_1, s_2, s_3, s_4, s_5\}$

Формирование множества вариантов построения подсистем, входящих в систему защиты от компьютерных атак, для всех уровней защищаемой системы в целом. Сформированные множества вариантов построения подсистемы, входящей в состав системы защиты от компьютерных атак, для каждого уровня защищаемой системы в отдельности будут иметь вид $V_{kl} = \{S_{k1}, S_{k2}, \ldots, S_{kN}\}$, где k – подсистема, входящая в систему защиты от компьютерных атак; l – уровень построения защищаемой системы; N – количество всех возможных вариантов построения k-й подсистемы для l-го уровня защищаемой системы ($l = \overline{1, L}$, где L – количество уровней в защищаемой системе).

Формирование множества вариантов V_k построения k-й подсистемы, входящей в систему защиты от компьютерных атак, для всех уровней защищаемой системы в целом осуществляется путем прямого произведения множеств всех возможных ва-

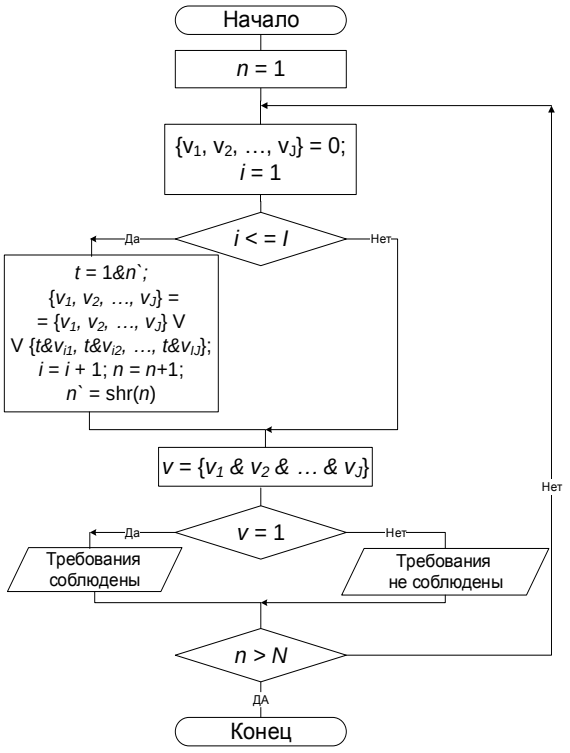

Рис. 4. Алгоритм проверки соответствия функциональным требованиям компонентов средств защиты для одного из уровней построения защищаемой системы

Fig. 4. Algorithm of checking the compliance with functional requirements of set of components of protection means for one of constructing levels of the protected system

риантов построения k-й подсистемы для каждого l-го уровня защищаемой системы [9]: $V_k = \prod_{l=1}^{L} V_{kl}$.

Для подсистемы обнаружения компьютерных атак данное выражение будет выглядеть следующим образом: $V_{\text{обн}} = \prod_{l=1}^{L} V_{\text{обн}l}$.

Для подсистемы противодействия компьютерным атакам оно приобретет вид $V_{\text{пр}} = \prod_{l=1}^{L} V_{\text{пр}l}$.

Для подсистемы устранения последствий применения компьютерных атак выражение будет следующим: $V_{\text{устр}} = \prod_{l=1}^{L} V_{\text{устр}l}$.

Отбор вариантов построения подсистем для конкретной конфигурации защищаемой информационно-вычислительной (автоматизированной) системы

Данная процедура осуществляется путем проверки совместимости компонентов подсистем системы защиты с программно-аппаратной платформой, на базе которой построена защищаемая информационно-вычислительная (автоматизированная) система, компонентов между собой в пределах отдельной подсистемы одного уровня, компонентов подсистем разных уровней между собой, а также компонентов, входящих в состав одной подсистемы всех уровней, с компонентами, входящими в состав других подсистем всех уровней.

Совместимость компонентов, на которых построены подсистемы системы защиты с программно-аппаратной платформой, определяется путем проверки отсутствия конфликтов при обращении к программным и аппаратным ресурсам информационно-вычислительной (автоматизированной) системы (функции ядра операционной системы, системные области памяти, порты ввода-вывода и т.п.).

Совместимость компонентов между собой в пределах отдельной подсистемы одного уровня определяется исходя из возможности выполнения всех функциональных требований для данной подсистемы при совместной работе компонентов, входящих в состав этой подсистемы. Если какой-либо компонент блокирует выполнение какой-либо функции другим компонентом и при этом блокирование данной функции влечет за собой невыполнение функциональных требований, определенных для данной подсистемы, эти два компонента считаются несовместимыми в пределах подсистемы и данный вариант построения подсистемы в формируемое множество альтернативных вариантов построения подсистем не включается. Если какой-либо компонент блокирует выполнение какой-либо функции другим компонентом, но при этом все функциональные требования, определенные для подсистемы, выполняются, данный вариант построения подсистемы включается во множество альтернативных вариантов.

Совместимость компонентов подсистем, входящих в состав системы защиты разных уровней, между собой и компонентов, входящих в состав одной подсистемы всех уровней, с компонентами других подсистем всех уровней также проверяется исходя из возможности выполнения всех функциональных требований, определенных для отдельных подсистем, входящих в систему защиты от компьютерных атак.

Предложенный в статье подход позволяет формировать множество возможных вариантов построения системы защиты от компьютерных атак с учетом выполнения всех функциональных требований, возложенных на подсистемы, входящие в состав системы защиты, с учетом особенностей функционирования и построения защищаемой системы, а также с учетом декомпозиции системы защиты от компьютерных атак на три подсистемы: обнаружения компьютерных атак, противодействия компьютерным атакам и устранения последствий применения компьютерных атак.

Литература

1. Дроботун Е.Б. Синтез систем защиты автоматизированных систем управления от разрушающих программных воздействий // Программные продукты и системы. 2016. Т. 29. № 3. С. 51–59.
2. Федоров Ю.Н. Справочник инженера по АСУ ТП: Проектирование и разработка. М.: Инфра-Инженерия, 2008. 928 с.
3. Малюк А.А., Пазизин С.В., Погожин Н.С. Введение в защиту информации в автоматизированных системах. М.: Горячая линия–Телеком, 2001. 148 с.
4. Основы проектирования и эксплуатации автоматизированных систем управления военного назначения; [под ред. В.Л. Лясковского]. М.: Изд-во МГТУ им. Н.Э. Баумана, 2016. 188 с.
5. Проскурин В.Г., Крутов С.В., Мацевич И.В. Программно-аппаратные средства обеспечения информационной безопасности. Защита в операционных системах. М.: Радио и связь, 2000. 168 с.
6. Сердюк В.А. Организация и технология защиты информации: обнаружение и предотвращение информационных атак в автоматизированных системах предприятий. М.: Изд-во ВВШЭ, 2011. 572 с.
7. Шумский А.А., Шелупанов А.А. Системный анализ в защите информации. М.: Гелиос АРВ, 2005. 224 с.
8. Бородакий Ю.В., Добродеев А.Ю., Нащекин П.А., Бутусов И.В. Основной объект воздействия противника // Воздушно-космическая оборона. 2014. № 2. С. 30–37.
9. Осипова В.А. Основы дискретной математики. М.: ФОРУМ: ИНФРА-М, 2006. 160 с.
10. Поляничко М.А. Архитектура системы автоматизированного обнаружения и разрешения конфликтов программных средств защиты информации // Изв. ПГУПС. 2013. № 1. С. 39–45.

Software & Systems
DOI: 10.15827/0236-235X.118.314-319

Received 02.10.16
2017, vol. 30, no. 2, pp. 314–319

METHOD OF GENERATING SETS OF ALTERNATIVE VARIANTS OF BUILDING SUBSYSTEMS WHICH ARE A PART OF A COMPUTER ATTACK PROTECTION SYSTEM

E.B. Drobotun [1], *Ph.D. (Engineering), Doctoral Student, drobotun@xakep.ru*
E.P. Uglovsky [1], *Head of Laboratory*
I.Sh. Zamaltdinov [1], *Ph.D. (Engineering), Senior Researcher*

[1] *Military Academy of the Aerospace Defense, Zhigareva St. 50, Tver, 170100, Russian Federation*

Abstract. Constructing a rational computer attack protection system for an information or automated system assumes creating a set of protection system configurations consisting of a set of separate program and hardware-software components and a further choice of a rational option of creating a computer attack protection system from the created set according to certain criteria. When generating this set, in addition to protection system compliance with necessary functional requirements, it is necessary to consider the parameters of the protected system (its structure and a multiple-level creation system), as well as software and hardware compatibility of components, and also compatibility of components with a hardware-software platform, which is the basis for the constructed protected system.

The article presents one of the possible approaches to forming a set of possible options of creating a computer attack protection system, taking into account its decomposition on three subsystem types. They are: computer attack detection subsystems, computer attack counteraction subsystems and subsystems of elimination of consequences of computer attack application.

Keywords: automated control system, computer attack, information security, protection system design, computer attack protection.

References

1. Drobotun E.B. Synthesis of protection systems of automated control systems against destroying program influence. *Programmnye produkty i sistemy* [Software & Systems]. 2016, vol. 29, no. 3, pp. 51–59 (in Russ.).
2. Fedorov Yu.N. *Spravochnik inzhenera po ASU TP: Proektirovanie i razrabotka* [Automated Process Control System Engineer's Reference Book: Design and Development]. Moscow, Infra-Inzheneriya Publ., 2008, 928 p.
3. Maluk A.A., Pazizin S.V., Pogozhin N.S. *Vvedenie v zashchitu informatsii v avtomatizirovannykh sistemakh* [Introduction to Information Protection in Automated Systems]. Moscow, Goryachaya liniya–Telekom Publ., 2001, 148 p. (in Russ.).
4. Lyaskovsky V.L. (Ed.) *Osnovy proektirovaniya i expluatatsii avtomatizirovannykh sistem voennogo naznacheniya* [Fundamentals of Design and Operation of Military Automated Control Systems]. Study guide. Moscow, N.E. Bauman MSTU Univ. Publ., 188 p.
5. Proskurin V.G., Krutov S.V., Matsevich I.V. *Programmno-apparatnye sredstva obecpecheniya bezopasnosti. Zaschita v operatsionnykh sistemakh* [Hardware-software Means of Ensuring Information Security. Protection in Operating Systems]. Study guide. Moscow, Radio i svyaz Publ., 2000, 168 p.
6. Serduk V.A. *Organizatsiya i tekhnologiya zashchity informatsii: obnaruzhenie i predotvrashchenie informatsionnykh atak v avtomatizirovannykh sistemakh predpriyaty* [Information Protection Organization and Technology: Detection and Prevention of Information Attacks in Enterprise Automated Systems]. Moscow, HSE Publ., 2011, 572 p.
7. Shumsky A.A., Shelupanov A.A. *Sistemny analiz v zashchite informatsii* [System Analysis in Information Security]. Moscow, Gelios ARV Publ., 2005, 224 p.
8. Borodaky Yu.V., Dobtodeev A.Yu., Nashchekin P.A., Butusov I.V. The Main Object of Enemy's Influence. *Vozdushno-kosmicheskaya oborona* [Aerospace Defense]. 2014, no. 2, pp. 30–37 (in Russ.).
9. Osipova V.A. *Osnovy diskretnoy matematiki* [Fundamentals of Discrete Mathematics]. Study guide. Moscow, FORUM: INFRA-M Publ., 2006, 160 p.
10. Polyanichko M.A. Architecture of the system of detection and resolution of the software security conflicts. *Izvestiya PGUPS* [Proc. of Petersburg Transport Univ.]. 2013, no. 1(34), pp. 39–45 (in Russ.).