УДК: 007 DOI: 10.53816/23061456_2022_1–2_22

МЕТОД КОНТРОЛЯ И ВОССТАНОВЛЕНИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ СЛОЖНЫХ ПРОГРАММНО-АППАРАТНЫХ ОБЪЕКТОВ

METHOD OF MONITORING AND RESTORING SOFTWARE OF COMPLEX HARDWARE AND SOFTWARE OBJECTS

Канд. техн. наук П.В. Закалкин

Ph.D. P.V. Zakalkin

Военная академия связи им. С.М. Буденного

Процессы глобализации и постоянно возрастающая технологическая сложность программно-аппаратных объектов привели к увеличению сложности самого объекта, а соответственно к увеличению затрат на его мониторинг и к увеличению времени восстановления объекта после сбоев в его функционировании. В статье рассматривается метод контроля и восстановления программного обеспечения сложных программно-аппаратных объектов. Предлагаемый метод позволяет сократить используемые вычислительные ресурсы и время, затрачиваемое на восстановление сложного программно-аппаратного объекта. Результат достигается за счет обоснованного корректирования периодичности контроля функций и параметров каждого элемента сложного программно-аппаратного объекта в соответствии с его структурно-функциональной схемой, и определения поэлементной последовательности восстановления сложного объекта.

Ключевые слова: сложный программно-аппаратный объект, техническое состояние, эксплуатационный отказ, программное обеспечение.

The processes of globalization and the ever-increasing technological complexity of software and hardware objects have led to an increase in the complexity of the object itself, and accordingly to an increase in the cost of its monitoring, and to an increase in the time for restoring the object from malfunctions in its operation. The article discusses the method of monitoring and restoring software of complex hardware and software objects. The proposed method allows to reduce used computing resources and time spent on restoration of complex hardware and software object. Result is achieved due to justified correction of periodicity of control of functions and parameters of each element of complex software-hardware object in accordance with its structure-functional diagram and determination of element-by-element sequence of restoration of complex object.

Keywords: complex software and hardware object, technical condition, operational failure, software.

Несмотря на высокую надежность существующих сложных программно-аппаратных объектов, достаточно актуальной задачей является подержание этих средств в работоспособном состоянии и обеспечение их оперативного восстановления в случае аппаратных или технических сбоев.

Для поддержания сложных программно-аппаратных объектов в работоспособном состоянии необходимо с минимальными временными

затратами и высокой степенью достоверности оценить причины приведшие к сбою, обеспечить оперативное устранение этих причин и привести объект в работоспособное состояние [1, 2].

Классические подходы подразумевают осуществление непрерывного контроля всех параметров сложного программно-аппаратного объекта, что требует привлечения больших измерительных и вычислительных ресурсов, при этом время реакции на изменения параметров и, следовательно, состояния объекта должно быть максимально высоким. Стоимость систем контроля и автоматизации управления объектом в этом случае может быть неприемлема высокой[3]. Таким образом, существующие подходы в области контроля и восстановления сложных объектов [4–6] обладают рядом недостатков, основными из которых являются:

– высокие требования к вычислительным ресурсам и времени, необходимому для восстановления сложных программно-аппаратных объектов;

– отсутствие обоснованной периодичности контроля функций и параметров каждого элемента сложного программно-аппаратного объекта и последовательности его восстановления.

Соответственно задача сокращения используемых вычислительных ресурсов и времени, затрачиваемого на восстановление сложного программно-аппаратного объекта, является актуальной. Одним из решений данной проблемы является периодичный контроль (в зависимости от степени критичности реализуемой функции) параметров качества каждой из *n*-функции, реализуемой в сложном программно-аппаратном объекте и контроль физических параметров элементов сложного программно-аппаратного объекта.

Предлагаемый метод позволяет решить эту задачу за счет обоснованного корректирования периодичности контроля функций и параметров каждого элемента сложного программно-аппаратного объекта в соответствии с его структурно-функциональной схемой и определения поэлементной последовательности восстановления сложного программно-аппаратного объекта.

Прежде, чем приступить к дальнейшему рассмотрению метода необходимо определиться с используемыми терминами и их определениями:

Функция — устойчивая совокупность однородных специализированных работ (действий, операций).

Элемент сложного программно-аппаратного объекта — составная часть объекта, которая выполняет определенную функцию (набор функций).

Сложный программно-аппаратный объект — это набор технических и программных средств, работающих совместно для выполнения одной или нескольких функций (набора функций).

Заявленный метод поясняется структурно-логической последовательностью, представленной на рис. 1.

В блоке 1 задают исходные данные. Формируют переносной носитель информации на котором создают базу данных в которую сохраняют исходные данные. Исходными данными метода являются:

– количество функций, реализуемых сложным объектом (*n*);

– периодичность выполнения и время реализации каждой из *n*-функций, реализуемой в сложном программно-аппаратном объекте;

– периодичность контроля параметров качества каждой из *n*-функций, реализуемой в сложном программно-аппаратном объекте;

– периодичность контроля физических параметров каждого элемента, используемого в составе сложного программно-аппаратного объекта. Задается исходя из минимального значения периодичности контроля функции из состава *n*-функций, реализуемых в сложном программно-аппаратном объекте;

– набор значений физических параметров, характеризующих функционирование элемента сложного программно-аппаратного объекта:

$$P_m = (P_1, P_2, P_3 ... P_z),$$

где P_m — эталонный набор физических параметров каждого элемента, используемого в составе сложного программно-аппаратного объекта; z — количество физических параметров элемента сложного программно-аппаратного объекта.

Набор значений допустимого отклонения физических параметров элемента, используемого в составе сложного программно-аппаратного объекта:

$$\Delta_m = (\Delta_1, \Delta_2, \Delta_3 ... \Delta_g),$$

где Δ_m — набор значений допустимого отклонения физических параметров элемента, ис-

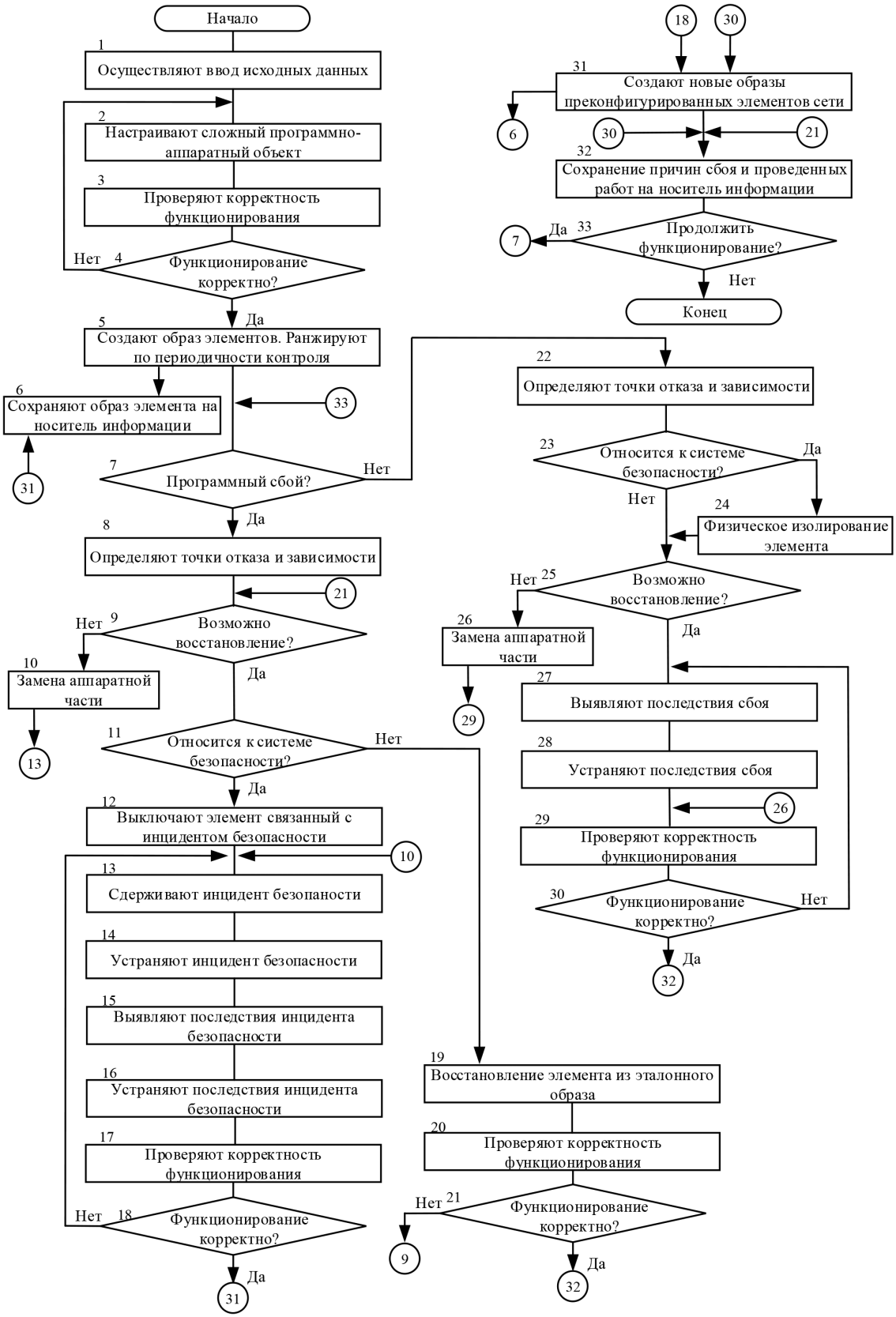

Рис. 1. Структурно-логическая последовательность метода контроля и восстановления программного обеспечения сложных программно-аппаратных объектов

пользуемого в составе сложного программно-аппаратного объекта; g — количество значений допустимого отклонения физических параметров элемента, используемого в составе сложного программно-аппаратного объекта.

Эталонный набор значений параметров качества каждой из n-функций, реализуемой в сложном программно-аппаратном объекте:

$$P_n = (P_1, P_2, P_3 ... P_k),$$

где P_n — эталонный набор значений параметров качества каждой из n-функций, реализуемой в сложном программно-аппаратном объекте; k — количество параметров качества реализуемой функции.

Набор значений допустимого отклонения параметров качества каждой из n-функций, реализуемой в сложном программно-аппаратном объекте:

$$\Delta_n = (\Delta_1, \Delta_2, \Delta_3 ... \Delta_l),$$

где Δ_n — набор значений допустимого отклонения параметров качества каждой из n-функций, реализуемой в сложном программно-аппаратном объекте; l — количество значений допустимого отклонения параметров качества реализуемой функции.

В блоке 2 настраивают сложный объект. Для реализации каждой из n-функций формируют последовательность задействования элементов сложного программно-аппаратного объекта. На основе сформированной последовательности задают последовательность снятия образов с элементов сложного программно-аппаратного объекта. После чего проверяют корректность функционирования объекта. Если набор значений физических параметров, характеризующих функционирование элемента (с учетом допустимого отклонения физических параметров), выходит за пределы, определенные в исходных данных, то возвращаются к этапу настройки программно-аппаратного объекта и на основе исходных данных осуществляют проверку настроек и дополнительную настройку объекта.

Если набор значений физических параметров, характеризующих функционирование элемента (с учетом допустимого отклонения физических параметров) соответствует определенным в исходных данных, то на основе заданной последовательности снятия образов с элементов сложного программно-аппаратного объекта последовательно снимают эталонные образы операционных систем каждого элемента. Полученные образы сохраняют в базу данных, которая находится на переносном носителе информации.

Учитывая, что каждый элемент сложного программно-аппаратного объекта может использоваться для реализации как одной функции, так и нескольких функций, в блоке 5 для каждого элемента сложного программно-аппаратного объекта ранжируют реализуемые им функции по периодичности контроля от минимального значения периодичности до максимального.

На этом этапе заканчивается настройка объекта и снятие образов элементов сложного программно-аппаратного объекта. Далее осуществляется процесс функционирования объекта и контроль физических параметров каждого элемента, входящего в его состав.

В процессе функционирования измеряют параметры качества каждой из n-функций, реализуемых сложным программно-аппаратным объектом [7], и физические параметры элементов, используемых в составе сложного программно-аппаратного объекта. При корректном функционировании сложного программно-аппаратного объекта с заданной периодичностью осуществляется снятие образов операционных систем элементов программно-аппаратного объекта.

Под корректным функционированием будем понимать соответствие текущих (полученных в результате измерения) параметров программно-аппаратного объекта эталонным (с учетом допустимого отклонения).

Если в процессе функционирования набор значений физических параметров, характеризующих функционирование элемента (с учетом допустимого отклонения физических параметров), выходит за пределы, определенные в исходных данных, то в блоке 7 определяют причину отказа: программный или технический сбой [2, 8].

Рассмотрим действия в случае программного сбоя. Под точкой отказа понимается тот элемент, о состоянии которого в текущий момент времени нет информации, и он в явном виде не работает. Например, если эксплуатируется модульный маршрутизатор, то в нем может отка-

зать как само шасси, так и входящие в него модули (блоки). Если у персонала достаточно компетенции для локализации и замены отказавших блоков в случае сбоя, то имеется несколько точек отказа в одном устройства, если нет — точка отказа одна [9–10].

Если программный сбой привел к выходу из строя аппаратной части элемента, то осуществляется его физическая замена, в противном случае переходят к блоку 11 и определяют тип сбоя. Если сбой не относится к нарушениям системы безопасности, последовательно извлекают из базы данных, находящейся на переносном носителе информации, образы операционных систем элементов, задействованных для реализации функции, и последовательно применяют их к элементам, реализующим функцию сложного программно-аппаратного объекта. После этого причины сбоя и виды проведенных работ сохраняются в базу данных.

В случае, если инцидент относится к системе безопасности, то дополнительно необходимо выполнить следующий комплекс мер (блоки 15–16), включающий в себя:

– сдерживание и устранение инцидента безопасности;

– выявление и устранение последствий инцидента безопасности.

На этапе сдерживания инцидента безопасности определяются источники и причины возникновения инцидента, а также осуществляется оценка его последствий. Действия по сдерживанию инцидента безопасности включают в себя:

– локализацию элемента, связанного с инцидентом безопасности;

– физическое отключение (выключение) элемента, связанного с инцидентом безопасности, с последующим отключением объектов, которые потенциально могут быть связаны с инцидентом безопасности;

– блокировка скомпрометированных учетных записей с последующей сменой учетных данных (паролей);

– блокировка трафика скомпрометированных элементов;

– мониторинг и блокировка несанкционированных каналов связи, а также несанкционированных на работу портов;

– другие мероприятия по сдерживанию инцидента безопасности.

На этапе устранения инцидента безопасности осуществляется устранение источника инцидента безопасности и исключение условий для его повторного возникновения. Основными действиями по устранению инцидента безопасности являются:

– поиск несанкционированно созданных учетных записей, в том числе привилегированных и обезличенных учетных записей;

– анализ событий изменения прав доступа;

– проверка элементов, ставших целью несанкционированного доступа, и источников инцидентов безопасности на предмет наличия уязвимостей, открытых портов, некорректно заданных параметров настроек оборудования и программного обеспечения;

– поиск и удаление вредоносного программного обеспечения;

– приведение параметров настройки оборудования и программного обеспечения в соответствие с политикой безопасности, установленной для элемента;

– другие мероприятия по устранению инцидента безопасности.

После этого осуществляют выявление последствий инцидента безопасности. Основными действиями по выявлению последствий инцидента безопасности являются:

– проверка целостности защищаемой информации (баз данных, конфигурационных файлов, резервных копий данных);

– анализ сформированных на этапе устранения инцидента безопасности подробных отчетов о зарегистрированных событиях на элемента, подвергшихся несанкционированному доступу;

– анализ действий, совершенных от имени скомпрометированной учетной записи элемента;

– другие мероприятия по выявлению последствий инцидента безопасности [11].

Действия по устранению последствий инцидента безопасности включают:

– подключение временно отключенных элементов и служб;

– снятие временных мер по ограничению сетевого трафика;

– восстановление данных из доверенных резервных копий;

– удаление несанкционированно зарегистрированных учетных записей;

– удаление обезличенных учетных записей;

– удаление несанкционированно подключенного коммутационного программного обеспечения.

Далее проверяют корректность функционирования. Если набор значений физических параметров, характеризующих функционирование элемента (с учетом допустимого отклонения физических параметров), соответствует определенным в исходных данных, то на основе заданной последовательности снятия образов с элементов сложного программно-аппаратного объекта последовательно снимают новые образы операционных систем элементов сложного программно-аппаратного объекта и сохраняют их в базу данных. После этого, причины сбоя и виды проведенных работ сохраняются в базу данных.

Рассмотрим действия в случае аппаратного отказа. Если в блоке 7 определен аппаратный отказ, то по аналогии, описанной ранее, определяют точки отказа и их зависимости (блок 22). Если отказ произошел на оборудовании, относящемся к системе безопасности (например: система обнаружения атак, криптомаршрутизатор и т.п.), то физически изолируют элемент программно-аппаратного объекта.

Далее, по аналогии с блоками 15 и 16 выявляют и устраняют последствия аппаратного сбоя и осуществляют проверку корректности функционирования сложного программно-аппаратного объекта. Данные о проведенных мероприятиях и выявленных причинах сбоя заносят в базу данных.

Выводы

Предлагаемый метод позволяет сократить используемые вычислительные ресурсы и время, затрачиваемое на восстановление сложного программно-аппаратного объекта, за счет обоснованного корректирования периодичности контроля функций и параметров каждого элемента сложного программно-аппаратного объекта в соответствии с его структурно-функциональной схемой и определения поэлементной последовательности восстановления сложного объекта.

Литература

1. Гречишников Е.В., Зубачев А.Б., Сазыкин А.М., Берлизев А.В. и др. Предложения по повышению быстродействия распределенной системы мониторинга компьютерных сетей, интегрированных в единую сети электросвязи // Вопросы оборонной техники. Серия 16. Технические средства противодействия терроризму. 2017. № 3–4 (105–106). С. 24–29.

2. Добрышин М.М., Закалкин П.В., Кузмич А.А. Система определения причин отказа в обслуживании в условиях эксплуатационных отказов и информационно - технических воздействий // Вопросы оборонной техники. Серия 16. Технические средства противодействия терроризму. 2020. № 7–8 (145–146). С. 38–43.

3. Стародубцев Ю.И., Иванов С.А., Вершенник Е.В. и др. Методика определения оптимальной периодичности контроля состояния сложного объекта // Вопросы оборонной техники. Серия 16. Технические средства противодействия терроризму. 2021. № 3–4 (153–154). С. 81–89.

4. Патент 2445686 Российская Федерация, МПК G06F 15/177. Способ установки, настройки, администрирования и резервного копирования программного обеспечения // Стручков И.В.; Заявитель и патентообладатель Стручков И.В. — 2010102826/08; заявл. 21.01.2010; опубл 20.03.2012. бюл. № 8. 24 с.

5. Закалкин П.В., Белов А.А. и др. Научно-технические предложения по совершенствованию системы диагностирования технического состояния средств связи // Проблемы технического обеспечения войск в современных условиях. Труды II межвузовской научно-практической конференции. 2017. С. 31–35.

6. Патент 2646309 Российская Федерация, МПК G06F 15/177 (2006.01). Способ резервного копирования // Анисимов В.В., Бегаев А.Н., Стародубцев Ю.И., Вершенник Е.В., Чукариков А.Г.; Заявитель и патентообладатель Бегаев А.Н. — 2017113265; заявл. 17.04.2017; опубл. 02.03.2018. бюл. № 7. 25 с.

7. Ваняшин С.В. Учебное пособие. Контроль качества предоставления услуг (SLA) в сетях IP/MPLS // Федерально агентство связи. ФГБОУ-ВПО «Поволжский государственный университет телекоммуникаций и информатики». — Самара. 2017. 100 с.

8. Гречишников Е.В., Добрышин М.М. Алгоритм мониторинга защищенности узла виртуальной частной сети от ddos-атак в условиях эксплуатационных отказов и сбоев // Проблемы

технического обеспечения войск в современных условиях. Труды научно-практической конференции. Военная академия связи. 2016. С. 48–51.

9. Планирование аварийного восстановления. Часть первая [Электронный ресурс] URL: https://habr.com/ru/post/225719/

10. Добрышин М.М., Белов А.А. и др. Научно-технические предложения по совершенствованию системы диагностирования технического состояния средств связи // Современное состояние и перспективы развития специальных систем радиосвязи и радиоуправления. Сборник докладов Всероссийской юбилейной научно-технической конференции, посвященной 60-летию образования Омского научно-исследовательского института приборостроения. 2018. С. 248–251.

11. Ададуров С.Е., Глухов А.П., Сидак А.А. и др. Реагирование на инциденты информационной безопасности в микропроцессорных системах железнодорожной автоматики и телемеханики // Двойные технологии. 2018. С. 76–81.

References

1. Grechishnikov E.V., Zubachev A.B., Sazykin A.M., Berlizev A.V. and others. Proposals to increase the speed of the distributed monitoring system of computer networks integrated into a single telecommunication network // Voprosy oboronnoi tekhniki. Seriia 16. Tekhnicheskie sredstva protivodestviia terrorizmu. 2017. № 3–4 (105–106). P. 24–29.

2. Dobryshin M.M., Zakalkin P.V., Kuzmich A.A. System for determining the causes of denial of service in conditions of operational failures and information and technical impacts // Voprosy oboronnoi tekhniki. Seriia 16. Tekhnicheskie sredstva protivodestviia terrorizmu. 2020. № 7–8 (145–146). P. 38–43.

3. Starodubtsev Yu.I., Ivanov S.A., Vershennik E.V. and others. Methodology for determining the optimal periodicity of monitoring the state of a complex object // Voprosy oboronnoi tekhniki. Seriia 16. Tekhnicheskie sredstva protivodestviia terrorizmu. 2021. № 3–4 (153–154). P. 81–89.

4. Patent 2445686 Russian Federation, IPC G06F 15/177. Method of installation, configuration, administration and backup of software // Struchkov I.V.; Applicant and patent holder Struchkov I.V. — 2010102826/08; declared. 21.01.2010; publ 20.03.2012. Bull. № 8. 24 p.

5. Zakalkin P.V., Belov A.A. and others. Scientific and technical proposals for improving the system for diagnosing the technical state of communications equipment // Problems of technical support of troops in modern conditions. Proceedings of the II interuniversity scientific and practical conference. 2017. P. 31–35.

6. Patent 2646309 Russian Federation, IPC G06F 15/177 (2006.01). Backup Method // Anisimov V.V., Begaev A.N., Starodubtsev Yu.I., Vershennik E.V., Chukarikov A.G.; Applicant and patent holder Begaev A.N. — 2017113265; declared. 17.04.2017; publ. 02.03.2018. Bull. № 7. 25 p.

7. Vanyashin S.V. Training manual. Quality of Service Quality Control (SLA) in IP/MPLS // Federal Communications Agency. FSBUVPO «Volga State University of Telecommunications and Informatics». — Samara. 2017. 100 p.

8. Grechishnikov E.V., Dobryshin M.M. Algorithm for monitoring the protection of a virtual private network node from ddos-attacks in conditions of operational failures and failures // Problems of technical support of troops in modern conditions. proceedings of the scientific and practical conference. Military Academy of Communications. 2016. P. 48–51.

9. Disaster recovery planning. Part One [Electronic Resource] URL: https://habr.com/ru/post/225719/

10. Dobryshin M.M., Belov A.A. and others. Scientific and technical proposals for improving the system for diagnosing the technical condition of communication equipment // Current state and prospects for the development of special radio communication and radio control systems. Collection of reports of the All-Russian Anniversary Scientific and Technical Conference dedicated to the 60th anniversary of the formation of the Omsk Scientific Research Institute of Instrument Making. 2018. P. 248–251.

11. Adadurov S.E., Glukhov A.P., Sidak A.A. and others. Response to information security incidents in microprocessor systems of railway automation and telemechanics // Dual technologies. 2018. P. 76–81.