Журнал зарегистрирован 01.03.2021 Федеральной службой по надзору в сфере связи, информационных технологий и массовых коммуникаций. Свидетельство о регистрации ПИ № ФС 77-80320

Адрес издателя и редакции: 630073, г. Новосибирск, пр. К. Маркса, 20.
E-mail: office@publish.nstu.ru и digital-tech-security@mail.ru
Web site: http://publish.nstu.ru и http://journals.nstu.ru/digital-tech-security/

Publisher and editorial office adress: 20 K. Marx Prospekt, Novosibirsk, 630073, Russian Fe-deration

До номера 1 (100) 2021 г. включительно журнал выходил под названием «Сборник научных трудов НГТУ» (ISSN 2307-6879)

16+

БЕЗОПАСНОСТЬ ЦИФРОВЫХ ТЕХНОЛОГИЙ

ISSN 2782-2230

№ 3 (110) **2023**

СОДЕРЖАНИЕ

Выпускающий редактор *И.П. Бровановa*
Корректор *Л.Н. Киншт*
Компьютерная верстка *С.И. Ткачева*

Лицензия № ИД 04303 от 20.03.01. Подписано в печать 19.09.2023. Выход в свет 28.09.2023
Формат 60×84/16. Бумага офсетная. Тираж 300 экз. Уч.-изд. л. 5,11
Печ. л. 5,5. Изд. № 175. Заказ № 253. Цена свободная

Отпечатано в типографии
Новосибирского государственного технического университета
630073, г. Новосибирск, пр. К. Маркса, 20

Publisher and editorial adress: 20 K. Marx Prospekt, Novosibirsk, 630073, Russian Federation
E-mail: office@publish.nstu.ru, digital-tech-security@mail.ru
Web site: http://publish.nstu.ru, http://journals.nstu.ru/digital-tech-security/

DIGITAL TECHNOLOGY SECURITY

ISSN 2782-2230

№ 3 (110) **2023**

CONTENTS

Publishing Editor *I.P. Brovanova*
Editor *L.N. Kinsht*
Computer imposition *S.I. Tkacheva*

License № ID 04303 from 20.03.01. Signed in print September 19, 2023
Date of publication September 28, 2023. Format 60 × 84 1/16
Offset Paper. Circulation is 300 copies. Educational-ed. liter. 5,11. printed pages 5,5
Publishing number 175. Order number 253

It is printed in printing house of Novosibirsk State Technical University
630073, Novosibirsk, 20 K. Marx Prospekt

БЕЗОПАСНОСТЬ ЦИФРОВЫХ ТЕХНОЛОГИЙ. – 2023. – № 3 (110). – 9–22

АВТОМАТИЗАЦИЯ И УПРАВЛЕНИЕ ТЕХНОЛОГИЧЕСКИМИ ПРОЦЕССАМИ И ПРОИЗВОДСТВАМИ

УДК 621.376 DOI: 10.17212/2782-2230-2023-3-9-22

РАЗРАБОТКА СИНХРОННОГО ДЕТЕКТОРА ДЛЯ ИЗМЕРЕНИЯ СОПРОТИВЛЕНИЙ НАНОСТРУКТУР*

С.М. СМАГИН[1], Е.С. КОПТЕВ[2], М.М. БАБИЧЕВ[3]

[1] *630073, РФ, г. Новосибирск, пр. Карла Маркса, 20, Новосибирский государственный технический университет, ассистент кафедры систем сбора и обработки данных. E-mail: kukishmen1972@mail.ru*
[2] *630073, РФ, г. Новосибирск, пр. Карла Маркса, 20, Новосибирский государственный технический университет, канд. физ.-мат. наук, доцент кафедры защиты информации. E-mail: koptev@corp.nstu.ru*
[3] *630073, РФ, г. Новосибирск, пр. Карла Маркса, 20, Новосибирский государственный технический университет, старший преподаватель кафедры защиты информации. E-mail: babichev@corp.nstu.ru*

Развитие электронной техники связано с необходимостью постоянного проведения всё более новых и сложных исследований. В наше время много внимания уделяется изучению свойств полупроводниковых наноструктур, получивших широкое распространение в наноэлектронике, наномеханике, материаловедении и многих других областях. Схема измерения сопротивлений полупроводниковых наноструктур (представляет собой четырехпроводную схему на квазипостоянном токе), используемая на данный момент, имеет недостаток, не позволяющий измерять большие значения сопротивлений (килоомы и более). В видоизмененной схеме, в которой этот недостаток устранен, для работы необходимо использование дополнительного измерительного прибора. Главной целью работы была разработка синхронного детектора для измерения значений тока в четырехпроводной схеме измерения сопротивлений наноструктур. В тексте статьи раскрываются основные принципы построения синхронных детекторов, их особенности, функции и области применения. Также приведены схемы самого измерительного устройства и его составных частей. Основная цель статьи – раскрыть принцип работы предложенного синхронного детектора. В конце статьи представлены характеристики разработанного устройства и приведены графики, иллюстрирующие работу каждой его части.

Ключевые слова: полупроводниковые наноструктуры, измерение сопротивления, синхронный детектор, четырехпроводная схема, квазипостоянный ток, опорный сигнал, умножитель, фильтр нижних частот

* Статья получена 02 июня 2023 г.

ВВЕДЕНИЕ

Для разработки электронных устройств, различных приборов и схем важно постоянно проводить новые и более сложные исследования. В настоящее время большое внимание уделяется изучению свойств полупроводниковых наноструктур. Наноструктуры широко используются в наномеханике, наноэлектронике, материаловедении и многих других областях. В настоящей статье рассмотрена установка, которая используется для измерения сопротивлений полупроводниковых наноструктур. Установка реализует четырехпроводную измерительную схему на квазипостоянном токе. Для повышения точности измерений и подавления шума в этой схеме используется синхронный детектор. Однако из-за некоторых ограничений с помощью этой схемы можно измерить только низкие значения сопротивлений (ориентировочно менее 1 кОм). В статье описывается метод расширения диапазона измеряемых сопротивлений (до сотен тысяч и миллионов ом), процесс разработки дополнительного синхронного детектора, необходимого для новой схемы.

1. ИЗМЕРИТЕЛЬНАЯ СХЕМА

На рис. 1 изображена измерительная схема.

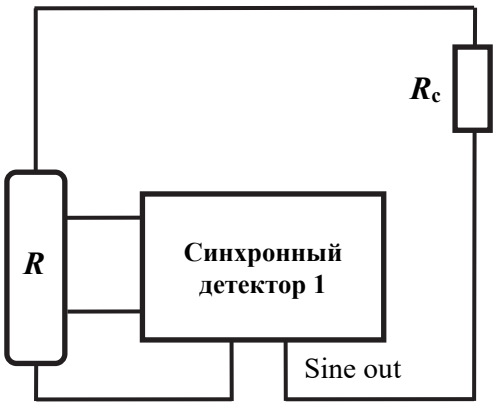


Рис. 1. Четырехпроводная схема измерения сопротивления

Fig. 1. Four-wire resistance measurement circuit

Это четырехпроводная схема измерения сопротивлений со стабилизацией тока. В ней напряжение смещения представляет собой синусоидальный сигнал, поступающий с выхода (sine out) синхронного детектора. Для стабилизации тока в цепи используется высокоомный резистор R_c (несколько мегаом). Синхронный детектор измеряет напряжение на проводнике. Однако основная особенность этой схемы заключается в том, что стабилизирующее сопротивление должно быть значительно выше измеряемого сопротивления $(R << R_c)$. Для измерения более высоких сопротивлений наноструктур в схеме необходимо использовать более высокие значения R_c и напряжения смещения.

Для расширения диапазона измеряемых сопротивлений эту схему необходимо модифицировать. В результате была разработана схема со стабилизацией напряжения (рис. 2).

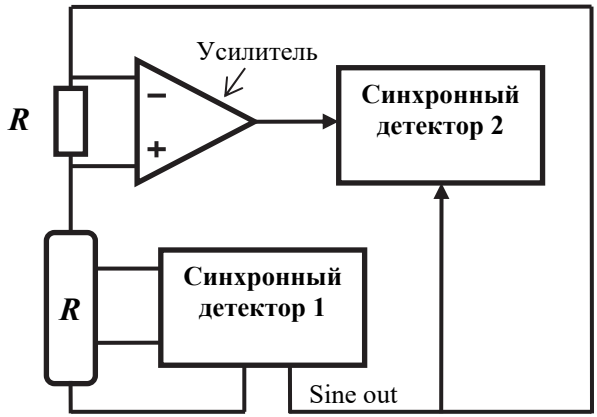


Рис. 2. Схема четырехпроводного измерения сопротивления со стабилизацией по напряжению

Fig. 2. Four-wire resistance measurement circuit with voltage stabilisation

В этой схеме для измерения напряжения на наноструктуре используется один синхронный детектор [1]. Для измерения тока в цепи стоит шунтирующий резистор (R на рис. 2) с очень низким сопротивлением и предусилитель. Выход предусилителя подключен ко второму синхронному детектору. Первый синхронный детектор устанавливает ток в цепи и генерирует опор-

ный сигнал для второго. Целью работы была разработка, создание и изучение характеристик дополнительного синхронного детектора. Этот синхронный детектор используется для измерения напряжения на шунте. С помощью новой схемы можно измерять сопротивления со значениями в несколько мегаом.

2. ПРИНЦИП СИНХРОННОГО ДЕТЕКТИРОВАНИЯ

Синхронный детектор – это устройство, используемое для обнаружения и измерения очень малых сигналов переменного тока (до нескольких нановольт) [2]. С помощью синхронного детектора можно проводить точные измерения даже при наличии шума, в несколько тысяч раз превышающего сам сигнал. Основной принцип работы синхронного детектора основан на использовании метода, известного как фазовое обнаружение. Этот метод позволяет выделить часть измеряемого сигнала на определенной опорной частоте и фазе. В то же время шумовые сигналы с частотами, отличающимися от опорной частоты, подавляются и не влияют на измерения [3]. На рис. 3 показана схема простейшего синхронного детектора.

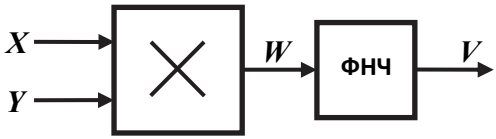


Рис. 3. Схема простейшего синхронного детектора

Fig. 3. The scheme of the simplest lock-in

Для работы синхронные детекторы используют опорный сигнал. В обычном случае сигнал, который необходимо измерить, имеет фиксированную частоту (на генераторе сигналов), и на этой опорной частоте детектор будет определять результаты измерений. Измеряемый сигнал синусоидальной формы будет выглядеть как $V_s \sin(\omega_s t + \varphi_s)$, где V_s – амплитуда измеряемого сигнала; ω_s – частота измеряемого сигнала; t – время; φ_s – фаза измеряемого сигнала. Опорный сигнал имеет вид $V_r \sin(\omega_r t + \varphi_r)$, где V_r – амплитуда опорного сигнала; ω_r – частота опорного сигнала; φ_r – фаза опорного сигнала. Пусть измеряемый сигнал будет *X*, а опорный *Y*. В синхронном детекторе

эти сигналы усиливаются и перемножаются. В результате на выходе умножителя будут две синусоидальных волны (1):

$$W = V_s V_r \sin(\omega_s t + \omega_s)\sin(\omega_r t + \omega_r) = \frac{1}{2} V_s V_r \cos([\omega_s - \omega_r]t + \omega_s - \omega_r) -$$

$$- \frac{1}{2} V_s V_r \cos([\omega_s + \omega_r]t + \omega_s + \omega_r)\,. \qquad (1)$$

Один из сигналов, полученных после умножения, представляет собой сумму частот $(\omega_s - \omega_r)$, другой – разность частот $(\omega_s + \omega_r)$. Далее полученный сигнал пропускается через фильтр нижних частот, где подавляются высокочастотные составляющие. В общем случае сигнал на выходе будет равен нулю, однако при равенстве ω_c и ω_o сигнал в виде разности частот станет постоянным и в результате на выходе фильтра будет

$$V = \frac{1}{2} V_s V_r \cos(\varphi_s - \varphi_r)\,. \qquad (2)$$

Постоянный сигнал, полученный на выходе фильтра (2), будет пропорционален амплитуде измеряемого сигнала.

Как уже упоминалось ранее, синхронный детектор может подавлять шумовые сигналы. Обычно входной сигнал содержит шумовую составляющую. Поскольку фильтр нижних частот пропускает только сигналы с частотами, очень близкими к опорной частоте, шумовые сигналы с частотами, сильно отличающимися от опорной частоты, будут подавляться на выходе фильтра. В результате шумовые сигналы с частотами, отличающимися от опорной частоты, будут давать на выходе очень малые значения.

Параметр подавления шума в значительной степени зависит от крутизны характеристики фильтра, а также от его полосы пропускания. Фильтр с более крутой характеристикой будет подавлять шумовые сигналы с частотами, близкими к опорной частоте, а более узкая полоса пропускания уменьшит возможность передачи ненужных сигналов. Поскольку ширина полосы пропускания фильтра определяет ширину обнаружения сигнала, необходимый результат на выходе фильтра даст (не будет зависеть от фильтра) только сигнал с частотой, согласованной с опорной.

3. ЗАВИСИМОСТЬ ОТ ФАЗЫ

Как было описано ранее, для измерения синхронному детектору необходим опорный сигнал, соответствующий частоте измеряемого сигнала $(\omega_s = \omega_r)$. Кроме того, фазы также должны быть одинаковыми и не меняться с течением времени. Так как в противном случае в (2) $\cos(\varphi_s - \varphi_r)$ будет меняться, и из-за этого выходной сигнал не будет постоянным. Однако есть способ полностью устранить эту фазовую зависимость.

На выходе синхронного детектора будет сигнал, пропорциональный $V = V_s \cos(\varphi_s - \varphi_r)$, где $\varphi = (\varphi_s - \varphi_r)$, φ – это разность фаз между опорным сигналом и измеряемым сигналом. Если изменять φ_r, эту разность можно сделать равной нулю. В результате этого можно измерить V_s (так как $\cos(0) = 1$). И наоборот, если $\varphi = 90°$, на выходе будет нуль. Синхронные детекторы, в устройстве которых один умножитель, называются однофазными и на выходе имеют $V_s \cos(\varphi)$.

Если в устройство добавить второй умножитель, эту фазовую зависимость можно убрать. При этом опорный сигнал, подаваемый на второй умножитель, необходимо повернуть на 90° (т. е. $V_r \sin(\omega_r t + \varphi_r + 90°)$). На выходе фильтра будет

$$V = \frac{1}{2} V_s V_r \cos(\varphi_s - \varphi_r),$$
$$V_2 \sim V_s \sin(\varphi), \tag{3}$$

где V_2 – сигнал, полученный со второго умножителя.

В результате на выходе получается два сигнала, один из которых пропорционален $\cos(\varphi)$, а другой – $\sin(\varphi)$. Первый из них можно назвать A, а второй – B:

$$A = V_s \cos(\varphi),$$
$$B = V_s \sin(\varphi). \tag{4}$$

Уравнения (4) представляют измеряемый сигнал как вектор, относящийся к опорному сигналу; A – это синфазная составляющая, а B – квадратурная составляющая, так как при $\varphi = 0$ A будет определять сигнал, а B будет равен нулю.

При вычислении модуля R вектора сигнала фазовой зависимости нет:

$$R = \sqrt{A^2 + B^2} = V_s \,, \tag{5}$$

R определяет амплитуду сигнала и не будет зависеть от разности фаз опорного и входного сигналов.

Как итог, двухфазный синхронный детектор состоит из двух умножителей. На его вход подаются два опорных сигнала, друг от друга отличающихся по фазе на 90°. С помощью такого вида детекторов можно непосредственно измерять A и B. Кроме этого, появляется возможность определения разности фаз измеряемого и опорного сигналов:

$$\varphi = \mathrm{arctg}\left(\frac{A}{B}\right). \tag{6}$$

Далее будут подробно рассмотрены основные части синхронного детектора, разобраны различные варианты их создания и описаны принципы работы.

4. РАЗРАБОТКА СИНХРОННОГО ДЕТЕКТОРА

На рис. 4 показана схема разработанного синхронного детектора. Также на ней изображена последовательность операций, выполняемых в каждом блоке.

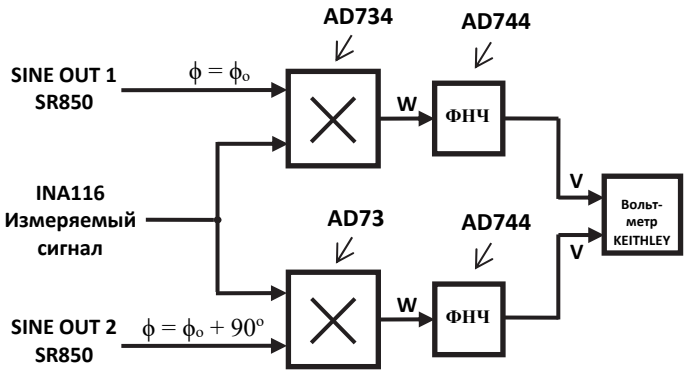


Рис. 4. Схема разработанного синхронного детектора

Fig. 4. The scheme of the developed lock-in

В синхронном детекторе были сделаны два канала. На один из них подается опорный сигнал. Опорный сигнал, подаваемый на второй, повернут на 90°. Также на вход каждого умножителя [4] подается измеряемый сигнал. Затем каждый из выходных сигналов умножителей, показанных на рис. 5, подается на фильтр нижних частот. Высокочастотная составляющая сигналов подавляется фильтрами. После этого выходные сигналы поступают на программируемый вольтметр. Вольтметр выполняет расчеты, необходимые для получения конечного значения, и вносит корректировки для повышения точности измерений.

Умножитель был основан на микросхеме AD734 [5]. Базовая схема умножения, рекомендованная производителем AD734, показана на рис. 5. В данном случае использовались следующие входы: X_1, X_2, Y_1, Y_2, VP, VN, W. Умножаемые сигналы (опорный и измеряемый) подаются на входы X_1, X_2, Y_1, Y_2; W (выход умножителя) был подключен к фильтру нижних частот; VP был подключен к источнику +15 В, а VN к источнику –15 В. Входы U_0, U_1, U_2 и Z_2 были подключены к земле. Кроме того, оба входа питания были подключены к земле через конденсатор. Конденсаторы были взяты на основе предложенной схемы и равны 0,1 мкФ.

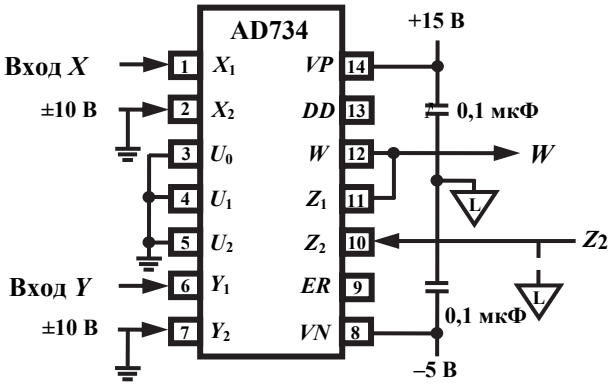


Рис. 5. Базовая схема умножения

Fig. 5. The basic multiplication circuit

На вход подавались два сигнала с частотой 10 Гц. Напряжение этих сигналов изменялось от нуля до 9 В. На рис. 6 показаны графики, основанные на измеренных значениях, и графики, основанные на теоретических

расчетах. Коэффициенты умножения, рассчитанные в ходе изучения умножителей, равны 0,0266 и 0,0273. Для моделирования схемы было использовано программное обеспечение EasyEDA [6].

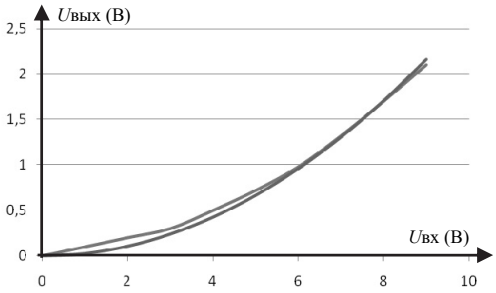


Рис. 6. Зависимость выходного напряжения от входного

Fig. 6. The dependence of the output voltage on the input

Следующей разработанной частью синхронного детектора является ФНЧ [7]. Поскольку ФНЧ будет использоваться в синхронном детекторе, его передаточная характеристика должна иметь резкий спад и узкую полосу пропускания. Эти параметры важны для того, чтобы фильтр мог пропускать только самые близкие частоты сигналов (опорного и измеряемого) и подавлять остальные [8].

В качестве ФНЧ была взята схема фильтра Чебышева третьего порядка со спадом 24 дБ на октаву. Этот тип фильтров обладает наиболее резкой передаточной характеристикой по сравнению с другими фильтрами того же порядка. Схема этого фильтра на основе микросхемы AD744 [9] показана на рис. 7. Частота среза 2 Гц. Предполагаемые частоты, на которых будет работать этот фильтр, составляют 10…15 Гц.

После сборки ФНЧ была проверена их работа. На фильтр подавался синусоидальный сигнал амплитудой 5 В. Характеристики фильтров показаны на рис. 8. Рассчитанные отклонения в полосе пропускания составили 11,9 для первого фильтра и 11,2 для второго фильтра.

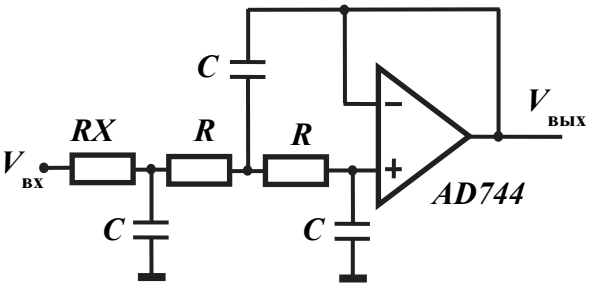


Рис. 7. Схема ФНЧ

Fig. 7. LPF circuit

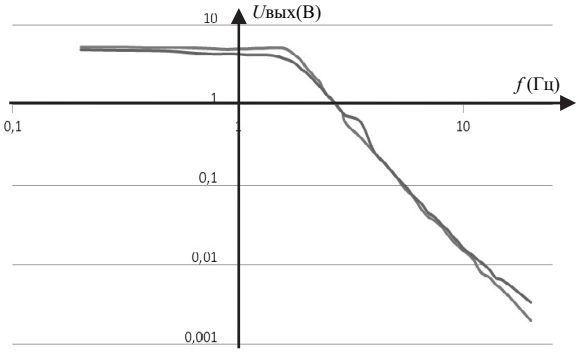


Рис. 8. Характеристики ФНЧ

Fig. 8. LPF characteristics

После сборки и изучения работы синхронного детектора были получены следующие характеристики (таблица). Микросхема INA116 [10] использовалась для схемы предварительного усилителя.

Параметры синхронного детектора

Lock-in characteristics

Характеристики	Параметры	
	Название параметра	Значение
1	Входы напряжения	Дифференциальный/синфазный
2	Макс. входное напряжение	12.5 В
3	Порог чувствительности	Ниже 100 мкВ
4	Входное сопротивление	2.5 МОм
5	Входная емкость	5 пФ
6	Предел погрешности	± 0.8 %
7	Динамический диапазон	Более 45 дБ
8	Частотный диапазон	От 2 Гц до 8.5 МГц

ЗАКЛЮЧЕНИЕ

В предлагаемой работе показана модификация четырехпроводной схемы измерения для высоких сопротивлений полупроводниковых наноструктур (порядка 10^6 Ом). Одной из основных частей этой схемы является синхронный детектор для измерения тока. Исследование направлено на его разработку и изучение.

Полученный детектор имеет характеристики, соответствующие поставленным целям. Материал статьи может быть использован для создания высокоточных схем, способных измерять не только сопротивление, но и другие электрические характеристики, такие как фазовый сдвиг. Это может быть полезно для дальнейших исследований полупроводниковых наноструктур.

СПИСОК ЛИТЕРАТУРЫ

1. Stanford Research Systems. MODEL SR850. DSP Lock-In Amplifier Manual. – URL: https://www.thinksrs.com/downloads/pdfs/manuals/SR850m.pdf (accessed: 31.08.2023).

2. *Степанов А.В.* Синхронный детектор: практикум / Московский государственный университет им. М.В. Ломоносова, кафедра физики колебаний. – М., 1997. – 16 с.

3. Синхронный детектор // Физическая энциклопедия: сайт. – URL: http://femto.com.ua/articles/part_2/3657.html (дата обращения: 31.08.2023).

4. Аналоговые перемножители сигналов. – URL: https://mydocx.ru/1-31102.html (дата обращения: 31.08.2023).

5. Analog Devices AD734 – Datasheet. – URL: https://static.chipdip.ru/lib/826/DOC031826798.pdf (accessed: 07.09.2023).

6. EasyEDA. Online PCB design & circuit simulator. – URL: https://easyeda.com/ (accessed: 31.08.2023).

7. Фильтр низкой частоты – что это такое? – URL: https://www.lcard.ru/lexicon/low_pass_filter (дата обращения: 31.08.2023).

8. *Сафронова Ю.Ф., Павлейно М.А.* Активные фильтры на основе операционного усилителя / Санкт-Петербургский государственный университет, кафедра радиофизики. – СПб., 2019. – 29 с.

9. Analog Devices AD744 Datasheet. – URL: https://doc.platan.ru/pdf/datasheets/analogdevices/AD744-877391.pdf (accessed: 07.09.2023).

10. Texas Instruments INA116 Datasheet. – URL: https://datasheet.dip8.ru/10d63ea819387dd64c94ba1e0258fda6.pdf (accessed: 07.09.2023).

Смагин Семен Михайлович, ассистент кафедры систем сбора и обработки данных Новосибирского государственного технического университета. Основное направление научных исследований – повышение пространственного разрешения в голографии и микроскопии. E-mail: kukishmen1972@mail.ru

Коптев Евгений Сергеевич, кандидат физико-математических наук, доцент кафедры защиты информации Новосибирского государственного технического университета. Область научных интересов – информационно-измерительные технологии и метрология. E-mail: koptev@corp.nstu.ru

Бабичев Михаил Михайлович, старший преподаватель кафедры защиты информации Новосибирского государственного технического университета. Область научных интересов – метрология и измерительные устройства. E-mail: babichev@corp.nstu.ru

DOI: 10.17212/2782-2230-2023-3-9-22

Development of a lock-in for measuring the resistances of nanostructures*

S.M. Smagin[1], E.S. Koptev[2], M.M. Babichev[3]

[1] *Novosibirsk State Technical University, 20 Karl Marx Prospekt, Novosibirsk, 630073, Russian Federation, assistant the Department of Data Acquisition and Processing Systems. E-mail: kukishmen1972@mail.ru*
[2] *Novosibirsk State Technical University, 20 Karl Marx Prospekt, Novosibirsk, 630073, Russian Federation, candidate of physical and mathematical sciences, Department of Information Security. E-mail: koptev@corp.nstu.ru*
[3] *Novosibirsk State Technical University, 20 Karl Marx Prospekt, Novosibirsk, 630073, Russian Federation, senior lecturer of the Department of Information Security. E-mail: babichev@corp.nstu.ru*

This paper presents a circuit which is used to measure the resistances of nanostructures and describes the principle of its operation. Next, stated a problem in the operation of this circuit and presented a method for solving this problem. The main goal of the work was to develop a lock-in for measuring current values in a four-wire circuit for measuring the resistance of nanostructures. The text reveals the basic principles of construction of lock-in amplifiers, its nuances and functions and where they can be used. The schemes of the device itself, its components are also given. The main purpose of this paper is to convey information on how lock-in amplifiers are work how they can be made and what they require for proper functioning. At the end of the article, the characteristics of the developed lock-in are presented and graphs illustrating the operation of each part of it are given.

Keywords: semiconductor nanostructures, lock-in, four-wire circuit, reference signal, multiplier, low-pass filter

REFERENCES

1. Stanford Research Systems. *MODEL SR850. DSP Lock-In Amplifier Manual*. Available at: https://www.thinksrs.com/downloads/pdfs/ /manuals/SR850m.pdf (accessed 31.08.2023).
2. Stepanov A.V. *Sinkhronnyi detektor: praktikum* [Lock-in amplifier. Practicum]. Lomonosov Moscow State University, Department of Oscillation Physics. Moscow, 1997. 16 p.
3. Sinkhronnyi detektor [Lock-in amplifier]. *Fizicheskaya entsiklopediya* [A physical encyclopedia]. Website. Available at: http://femto.com.ua/articles/part_2/3657.html (accessed 31.08.2023).
4. *Analogovye peremnozhiteli signalov* [Analog signal multipliers]. Available at: https://mydocx.ru/1-31102.html (accessed 31.08.2023).

* *Received 02 June 2023.*

5. *Analog Devices AD734 – Datasheet*. Available at: https://static.chipdip.ru/lib/826/DOC031826798.pdf (accessed 07.09.2023).

6. EasyEDA. *Online PCB design & circuit simulator*. Available at: https://easyeda.com/ (accessed 31.08.2023).

7. *Fil'tr nizkoi chastoty – chto eto takoe?* [Low-pass filter – what is it?]. Lcard.ru https://www.lcard.ru/lexicon/low_pass_filter (accessed 31.08.2023).

8. Safronova Yu.F., Pavleino M.A. *Aktivnye fil'try na osnove operatsionnogo usilitelya* [Active filters based on an operational amplifier]. St. Petersburg State University, Department of Radiophysics. St. Petersburg, 2019. 29 p.

9. *Analog Devices AD744 Datasheet*. Available at: https://doc.platan.ru/pdf/datasheets/analogdevices/AD744-877391.pdf (accessed 07.09.2023).

10. *Texas Instruments INA116 Datasheet*. Available at: https://datasheet.dip8.ru/10d63ea819387dd64c94ba1e0258fda6.pdf (accessed 07.09.2023).

Для цитирования:

Смагин С.М., *Коптев Е.С., Бабичев М.М.* Разработка синхронного детектора для измерения сопротивлений наноструктур // Безопасность цифровых технологий. – 2023. – № 3 (110). – С. 9–22. – DOI: 10.17212/2782-2230-2023-3-9-22.

For citation:

Smagin S.M., Koptev E.S., Babichev M.M. Razrabotka sinkhronnogo detektora dlya izmereniya soprotivlenii nanostruktur [Development of a lock-in for measuring the resistances of nanostructures]. *Bezopasnost' tsifrovykh tekhnologii = Digital Technology Security*, 2023, no. 3 (110), pp. 9–22. DOI: 10.17212/2782-2230-2023-3-9-22.

БЕЗОПАСНОСТЬ ЦИФРОВЫХ ТЕХНОЛОГИЙ. – 2023. – № 3 (110). – 23–39

МЕТОДЫ И СИСТЕМЫ ЗАЩИТЫ ИНФОРМАЦИИ, ИНФОРМАЦИОННАЯ БЕЗОПАСНОСТЬ

УДК 004.422 DOI: 10.17212/2782-2230-2023-3-23-39

РАЗРАБОТКА АНАЛИЗАТОРА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ С ФУНКЦИЕЙ МОНИТОРИНГА*

В.А. МАЗУРЕНКО[1], А.В. ИВАНОВ[2]

[1] *630087, РФ, г. Новосибирск, пр. Карла Маркса, 20, Новосибирский государственный технический университет, магистрант кафедры защиты информации. E-mail: mazurenko.2017@stud.nstu.ru*
[2] *630073, РФ, г. Новосибирск, пр. Карла Маркса, 20, Новосибирский государственный технический университет, кандидат технических наук, заведующий кафедрой защиты информации. E-mail: andrej.ivanov@corp.nstu.ru*

В реалиях современного мира, в том числе и перехода из WEB 2.0 в WEB 3.0, ежедневно появляются новые продукты, которые делают жизнь человека удобнее. Это касается и программного обеспечения (ПО), которое используется в бизнесе или повседневной жизни. Появление нового программного обеспечения от специалистов разного уровня несет в себе как положительные, так и отрицательные моменты. Вслед за новыми технологиями появляются и новые угрозы. За 2022 год 70 % самых эксплуатируемых уязвимостей были связаны именно с уязвимостями в ПО. В рамках настоящей статьи предпринимается попытка разработки эффективного анализатора уязвимостей ПО с функцией мониторинга, способного обнаруживать новые уязвимости без оказания нагрузки на сеть. В статье описаны этапы разработки статьи, принцип работы программы, информация об источниках получения данных о целевой системе и о базе уязвимостей. Описан процесс мониторинга и возможности оператора при получении информации о имеющихся уязвимостях. Для проверки эффективности работы подготовлен тестовый стенд с уязвимостями, на котором проверится эффективность разработанного анализатора и инструмента с похожим функционалом.

Ключевые слова: уязвимости, разработка анализатора, программное обеспечение, контроль уязвимостей, мониторинг, кибербезопасность, сканер уязвимости, CVE, NIST

* Статья получена 05 июня 2023 г.

ВВЕДЕНИЕ

В современном мире технологии играют огромную роль в жизни, включая различные аспекты, такие как коммуникация, работа и развлечения. Однако с ростом использования технологий также растет угроза киберпреступности. Хакеры и другие злоумышленники постоянно ищут уязвимости в программном обеспечении (ПО), чтобы получить несанкционированный доступ к системам и данным. В этой ситуации необходимо разработать эффективный инструмент для обнаружения уязвимостей в ПО и предотвращения их эксплуатации.

Сегодня киберпреступность является одной из самых быстрорастущих проблем в области безопасности информации. С каждым днем появляются новые уязвимости и способы атаки на программное обеспечение (ПО), которые могут привести к серьезным последствиям для пользователей, организаций и даже государств.

Для защиты от кибератак разработчики ПО и эксперты по безопасности информации создают новые инструменты, которые позволяют обнаруживать и устранять уязвимости. Однако существующие инструменты анализа уязвимостей не всегда могут обеспечить полноценную защиту от новых угроз, которые могут возникнуть в будущем.

В настоящей статье будет рассмотрена разработка анализатора уязвимостей ПО с функцией мониторинга, который позволит проактивно обнаруживать и предотвращать новые угрозы безопасности. Основная цель работы заключается в разработке и реализации анализатора уязвимостей, способного находить новые и уже известные уязвимости, а также проводить их анализ и мониторинг.

Таким образом, разработка такого анализатора будет способствовать повышению уровня безопасности программного обеспечения, защите пользователей и организаций от потенциальных киберугроз и обеспечению устойчивости информационной инфраструктуры в целом.

1. АРХИТЕКТУРА АНАЛИЗАТОРА УЯЗВИМОСТЕЙ

Архитектура анализатора уязвимостей ПО может быть представлена как совокупность компонентов и модулей, выполняющих функции сканирования, обнаружения и анализа уязвимостей в ПО.

Главными компонентами анализатора уязвимостей ПО являются:

- сбор информации (информации о целевой системе, включая информацию относительно установленного ПО, а также наполнение базы данных существующими уязвимостями);

- сканирование уязвимостей (сканирование системы на наличие известных уязвимостей в ПО и службах, установленных в системе);
- анализ результатов (анализ результатов сканирования и выдача отчета о выявленных уязвимостях);
- мониторинг уязвимостей (мониторинг системы на наличие новых уязвимостей в ПО). Компонент использует систему оповещения.

Преимущества архитектуры анализатора уязвимостей ПО с функцией мониторинга:

- обнаружение новых уязвимостей. Благодаря мониторингу уязвимостей анализатор ПО может обнаруживать новые уязвимости сразу после их появления, что позволяет быстро реагировать на угрозы безопасности;
- автоматизация процесса. Анализатор ПО может автоматически сканировать систему на наличие уязвимостей и выдавать отчеты, что упрощает процесс обнаружения и устранения уязвимостей.

2. ПРИНЦИП РАБОТЫ АНАЛИЗАТОРА

2.1. СБОР ИНФОРМАЦИИ

Информацию об уязвимостях в ПО можно получить из различных источников:

- официальные сайты производителей ПО. Производители выпускают исправления уязвимостей и публикуют информацию об этом на своих сайтах;
- национальные и международные базы данных уязвимостей. Такие организации, как CERT, NIST, CVE, OWASP и другие, поддерживают базы данных уязвимостей, которые содержат информацию о последних уязвимостях, исправлениях и рекомендациях по их устранению;
- форумы, блоги и сообщества. Пользователи и эксперты могут публиковать информацию об уязвимостях, их исправлениях и методах защиты на форумах, в блогах и в социальных сетях;
- журналы безопасности. Журналы, такие как SecurityFocus, Dark Reading, SC Magazine и другие, публикуют статьи, новости и аналитические материалы о последних уязвимостях и их влиянии на информационную безопасность.

В целом, для получения информации об уязвимостях в ПО следует использовать различные источники и подписываться на обновления, чтобы быть в курсе последних событий.

В качестве главных источников информации о существующих уязвимостях решено взять National Institute of Standards and Technology (NIST).

NIST – это Национальный институт стандартов и технологий, агентство Министерства торговли США, созданное для развития и применения технологий, метрологических и стандартных методов в различных отраслях промышленности, включая информационные технологии и кибербезопасность [1].

Основные задачи NIST в области кибербезопасности включают разработку и публикацию стандартов и руководств, проведение исследований и анализа уязвимостей, а также разработку методов и инструментов для защиты информации и кибербезопасности.

NIST также поддерживает каталог уязвимостей (NVD – National Vulnerability Database), который является базой данных уязвимостей и содержит информацию о новых и известных уязвимостях, а также о связанных с ними угрозах и рекомендациях по их устранению.

Кроме того, NIST также разрабатывает и публикует руководства и стандарты в области кибербезопасности, такие как Стандарты шифрования данных (AES, SHA), Стандарты безопасности информационных систем (SP 800) и многое другое.

CVE – это стандарт идентификации уязвимостей в компьютерных системах и программном обеспечении (Common Vulnerabilities and Exposures), который был создан для облегчения обмена информацией об уязвимостях между различными организациями и производителями ПО [2].

Каждая уязвимость, опубликованная в CVE, имеет уникальный идентификатор, который состоит из префикса CVE-, за которым следует уникальный номер, например, CVE-2022-1234. Этот идентификатор используется для идентификации и отслеживания уязвимостей в различных базах данных уязвимостей.

CVE представляет собой совместный проект, который управляется организацией MITRE в партнерстве с CERT/CC, NIST и др. В рамках этого проекта эксперты по безопасности из различных организаций и компаний работают вместе для идентификации и описания новых уязвимостей, определения уровня угрозы и рекомендаций по устранению уязвимостей.

CVE является важным инструментом для анализа и управления уязвимостями в информационных системах и программном обеспечении. Благодаря стандартной идентификации уязвимостей различные организации и производители ПО могут легко обмениваться информацией об уязвимостях, быстро реагировать на новые угрозы и устранять уязвимости в своих продуктах. Из CVE настроена автоматическая выгрузка базы уязвимостей для последующего сравнения с ПО, установленным в системе, а также возможность ручного добавления уязвимостей. Сбор информации об установленном ПО в системе будем осуществлять с помощью выгрузки данных о серверах из Itop, ручного добавления ПО в программу, а также Ansible.

2.2. СКАНИРОВАНИЕ УЯЗВИМОСТЕЙ

Сканирование уязвимостей будет происходить на сервере, сам процесс сканирования заключается в сравнении выгруженной базы CVE и списка установленного ПО, разработка происходила на языке Python [3]. Из ПО извлекается его версия. Сканирование выполняется с помощью функции (листинг 1). Данная функция принимает на вход две версии ПО и оператор сравнения и возвращает True, если оператор сравнения выполняется для этих версий, и False, если оператор сравнения не выполняется.

Листинг 1 – Код функции сравнения

```
def _if_condition_true(v1, op, v2):
    v1 = _get_str_without_last_dot(v1)
    v2 = _get_str_without_last_dot(v2)

    # в версии могут быть буквы, их не учитываем
    v1, letters1 = _extract_letters_at_the_end(v1)
    v2, letters2 = _extract_letters_at_the_end(v2)

    # если версии равны - то сразу возвращаем True
    if v1 == v2 and (op == "<=" or op == ">="):
        return True

    list_v1 = v1.split(".")
    list_v2 = v2.split(".")

    # добиваем меньшую версию нулями до длины большей
    if len(list_v2) > len(list_v1):
        list_v1 = _fill_versions(list_v1, len(list_v2))
    elif len(list_v1) > len(list_v2):
        list_v2 = _fill_versions(list_v2, len(list_v1))
    # LOGIC
    # сравниваем части версий между точками
    l = len(list_v1)
    # здесь уже равенства быть не может, только > <
    # _get_value_to_compare: возвращает int, а при невозможности - str
    for i in range(0, len(list_v1) - 1):
        if op == ">":
            if _get_value_to_compare(list_v1[i]) < _get_value_to_compare(list_v2[i]):
                return False
        elif op == ">=":

        if _get_value_to_compare(list_v1[i]) < _get_value_to_compare(list_v2[i]):
            return False
    elif op == "<":
        if _get_value_to_compare(list_v1[i]) > _get_value_to_compare(list_v2[i]):
            return False
    elif op == "<=":
        if _get_value_to_compare(list_v1[i]) > _get_value_to_compare(list_v2[i]):
            return False
  return True
```

Сначала входные версии подготавливаются к сравнению. Удаляется последний символ (точка), если он присутствует в каждой версии, чтобы избежать сравнения строк различной длины, которые в конце содержат точки. Затем от версий отделяются возможные буквенные символы в конце, которые не учитываются при сравнении.

Далее происходит заполнение меньшей версии нулями до большей длины. Это делается для удобства сравнения, чтобы разные части версии находились на одинаковых позициях в списке. Если на входе есть разные по длине версии, то короткая версия дополняется нулями до большей длины.

После этого происходит логическое сравнение версий. Для этого входные версии представляются в виде списков чисел, разделенных точками. Затем в цикле сравниваются части версии между точками в списке. Если текущая часть одной версии меньше соответствующей части другой версии, то возвращается False. Если же текущая часть одной версии больше или равна соответствующей части другой версии, то цикл продолжается. Если все части версии одной версии больше или равны соответствующим частям другой версии, то возвращается True.

Если входные версии равны и оператор сравнения <= или >=, то функция возвращает True.

2.3. АНАЛИЗ РЕЗУЛЬТАТОВ

В итоге выполнения поиска уязвимостей получена таблица с перечнем уязвимых ПО (рис. 1).

Таблица состоит из следующих столбцов: id, название приложения, его версия, имя уязвимости, ее краткое описание, критичность уязвимости; сервер, на котором найдена уязвимость, и источник, откуда взята информация о наличии данного ПО в системе. Так как система постоянно обновляется, есть вероятность того, что при анализе найденных уязвимостей будет обнаружено, что данного ПО уже нет в системе, но по каким-либо причинам в iTop оно числится, или найденная уязвимость может быть неприменима к системе по ряду причин, в таком случае возможно либо исключить ненужное ПО, либо добавить найденную уязвимость на одном из серверов по id.

Информация о системе хранится в таблице «Таблица ПО с автоматическим добавлением», основные столбцы несут информацию о имени приложения, его версии, вендере, а также об источнике, сообщающем о данном ПО.

web.py

Найденные уязвимости

Id	Приложение	Версия	Уязвимость	Описание
40	jdk	1.4.2	CVE-2011-3561	Unspecified vulnerability in the Java Runtime Environment component in Oracle...
83	proftpd	1.3.6	CVE-2011-4130	Use-after-free vulnerability in the Response API in ProFTPD before 1.3.3g all...
41	jdk	1.4.2	CVE-2012-0504	Unspecified vulnerability in the Java Runtime Environment (JRE) component in ...
42	jdk	1.4.2	CVE-2012-0547	Unspecified vulnerability in the Java Runtime Environment (JRE) component in ...
43	jdk	1.4.2	CVE-2012-1725	Unspecified vulnerability in the Java Runtime Environment (JRE) component in ...
44	jdk	1.4.2	CVE-2012-1726	Unspecified vulnerability in the Java Runtime Environment (JRE) component in ...
45	jdk	1.4.2	CVE-2012-2739	Oracle Java SE before 7 Update 6, and OpenJDK 7 before 7u6 build 12 and 8 bef...
46	jdk	1.4.2	CVE-2012-3136	Unspecified vulnerability in the Java Runtime Environment (JRE) component in ...
47	jdk	1.4.2	CVE-2012-5089	Unspecified vulnerability in the Java Runtime Environment (JRE) component in ...
48	jdk	1.4.2	CVE-2013-1493	The color management (CMM) functionality in the 2D component in Oracle Java S...
49	jdk	1.4.2	CVE-2013-2439	Unspecified vulnerability in the Java Runtime Environment (JRE) component in ...
50	jdk	1.4.2	CVE-2013-2440	Unspecified vulnerability in the Java Runtime Environment (JRE) component in ...
51	jdk	1.4.2	CVE-2013-3743	Unspecified vulnerability in the Java Runtime Environment (JRE) component in ...
52	jdk	1.4.2	CVE-2013-3744	Unspecified vulnerability in the Java Runtime Environment (JRE) component in ...
53	jdk	1.4.2	CVE-2013-4578	jarsigner in OpenJDK and Oracle Java SE before 7u51 allows remote attackers t...
94	tomcat	5.5.28	CVE-2013-4590	Apache Tomcat before 6.0.39, 7.x before 7.0.50, and 8.x before 8.0.0-RC10 all...
54	jdk	1.4.2	CVE-2013-5838	Unspecified vulnerability in Oracle Java SE 7u25 and earlier, and Java SE Emb...
55	jdk	1.4.2	CVE-2013-5850	Unspecified vulnerability in Oracle Java SE 7u40 and earlier, Java SE 6u60 an...
56	jdk	1.4.2	CVE-2013-5852	Unspecified vulnerability in Oracle Java SE 7u40 and earlier, Java SE 6u60 an...
57	jdk	1.4.2	CVE-2013-5854	Unspecified vulnerability in Oracle Java SE 7u40 and earlier and JavaFX 2.2.4...

Рис. 1. Найденные уязвимости в системе

Fig. 1. Vulnerabilities found in our system

Таблица ПО с автоматическим добавлением показана на рис. 2. Прямо из веб-интерфейса пользователь имеет возможность редактировать, к примеру, версию приложения либо удалить информацию о нем. В данной таблице хранится информация о всех приложениях, добавленных как автоматически, так и вручную.

Уязвимости ПО могут быть классифицированы по критичности на основе потенциальных последствий и уровня уязвимости. Обычно уязвимости классифицируются следующим образом.

1. Критические уязвимости – уязвимости, которые могут привести к серьезному нарушению безопасности системы (например, к повышению привилегий на сервере или к получению удаленного доступа к системе без аутентификации).

2. Высокие уязвимости – уязвимости, которые могут привести к существенному нарушению безопасности системы, но не так серьезны, как крити-

ческие уязвимости. Например, это может быть уязвимость, позволяющая злоумышленнику получить доступ к конфиденциальной информации.

3. Средние уязвимости – уязвимости, которые могут привести к некоторому нарушению безопасности системы, но не имеют такого высокого уровня серьезности. Это могут быть, например, уязвимости, которые позволяют провести атаку переполнения буфера или осуществить XSS-атаку.

4. Низкие уязвимости – уязвимости, которые могут привести к небольшому нарушению безопасности системы. Это могут быть, например, уязвимости, связанные с недостаточной обработкой ошибок или с использованием слабых алгоритмов шифрования.

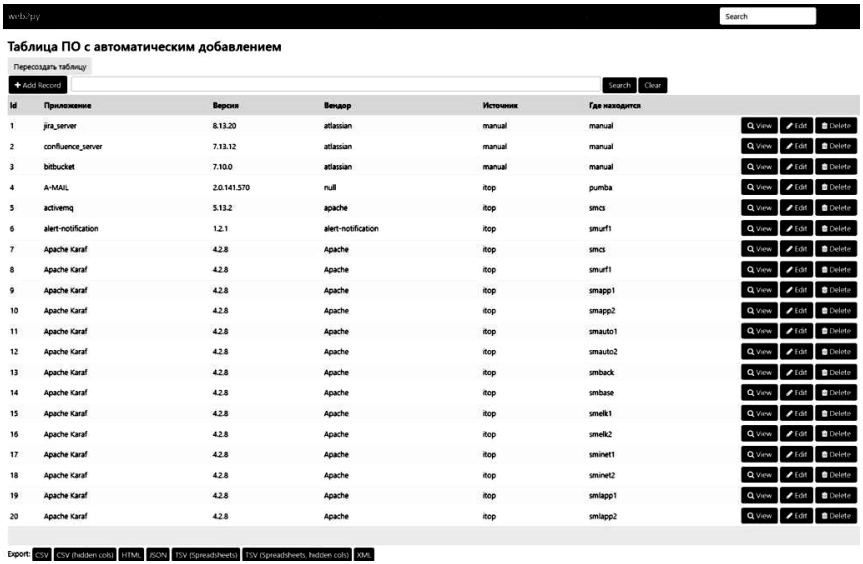


Рис. 2. ПО с автоматическим добавлением

Fig. 2. Software with automatic addition

Классификация уязвимостей по критичности помогает организациям определить, какие уязвимости нужно в первую очередь устранять, а какие можно отложить на более поздний срок. Она также позволяет оценить, насколько важно заботиться о безопасности системы и какие меры безопасности должны быть приняты.

После отсеивания ненужных нам уязвимостей в рамках разработки анализатора уязвимостей ПО будем считать, что необходимые нам уязвимости,

несут информацию об уязвимостях в ПО и ОС. Итоговый вариант таблицы, необходимый для последующего анализа, представлен на рис. 3.

web2py

Search

Перечень уязвимых версий продуктов

Search Clear

Cve Id	Type	Vendor	Product	Version	Cpe23Uri	Versionstartexcluding	Versionendexcluding	Versionstartincluding	Versionendincluding
CVE-2023-27986	Application	gnu	emacs	*	cpe:2.3:a:gnu:emacs:*:*:*:*:*:*:*:*			28.1	28.2
CVE-2023-27974	Application	bitwarden	bitwarden	*	cpe:2.3:a:bitwarden:bitwarden:*:*:*:*:browser:*:*:*				2023.2.1
CVE-2023-27905	Application	jenkins	update-center2	3.14	cpe:2.3:a:jenkins:update-center2:3.14:*:*:*:*:jenkins:*:*				
CVE-2023-27905	Application	jenkins	update-center2	3.13	cpe:2.3:a:jenkins:update-center2:3.13:*:*:*:*:jenkins:*:*				
CVE-2023-27904	Application	jenkins	jenkins	*	cpe:2.3:a:jenkins:jenkins:*:*:*:*:-:*:*:*		2.394		
CVE-2023-27904	Application	jenkins	jenkins	*	cpe:2.3:a:jenkins:jenkins:*:*:*:*:lts:*:*:*		2.375.4		
CVE-2023-27891	Application	rami	pretix	4.16.0	cpe:2.3:a:rami:pretix:4.16.0:*:*:*:*:*:*:*				
CVE-2023-27891	Application	rami	pretix	4.17.0	cpe:2.3:a:rami:pretix:4.17.0:*:*:*:*:*:*:*				
CVE-2023-27891	Application	rami	pretix	*	cpe:2.3:a:rami:pretix:*:*:*:*:*:*:*:*		4.15.1	1.16.0	
CVE-2023-27850	OS/Firmware	netgear	rax30_firmware	*	cpe:2.3:o:netgear:rax30_firmware:*:*:*:*:*:*:*:*		1.0.10.94		
CVE-2023-27641	Application	lsoft	listserv	*	cpe:2.3:a:lsoft:listserv:*:*:*:*:*:*:*:*		17.0		
CVE-2023-27635	Application	debian	debmany	0.88.1	cpe:2.3:a:debian:debmany:0.88.1:*:*:*:*:*:*:*				
CVE-2023-27574	Application	shadowsocks	shadowsocksx-ng	1.10.0	cpe:2.3:a:shadowsocks:shadowsocksx-ng:1.10.0:*:*:*:*:*:*:*				
CVE-2023-27567	OS/Firmware	openbsd	openbsd	7.2	cpe:2.3:o:openbsd:openbsd:7.2:*:*:*:*:*:*:*				
CVE-2023-27566	Application	live2d	cubism_editor	4.2.03	cpe:2.3:a:live2d:cubism_editor:4.2.03:*:*:*:*:*:*:*				
CVE-2023-27561	Application	linuxfoundation	runc	*	cpe:2.3:a:linuxfoundation:runc:*:*:*:*:*:*:*:*				1.1.4
CVE-2023-27561	OS/Firmware	redhat	enterprise_linux	8.0	cpe:2.3:o:redhat:enterprise_linux:8.0:*:*:*:*:*:*:*				
CVE-2023-27561	Application	redhat	openshift_container_platform	4.0	cpe:2.3:a:redhat:openshift_container_platform:4.0:*:*:*:*:*:*:*				
CVE-2023-27561	OS/Firmware	redhat	enterprise_linux	9.0	cpe:2.3:o:redhat:enterprise_linux:9.0:*:*:*:*:*:*:*				
CVE-2023-27560	Application	phpseclib	phpseclib	*	cpe:2.3:a:phpseclib:phpseclib:*:*:*:*:*:*:*:*		3.0.19	3.0.0	

Export: CSV CSV (hidden cols) HTML JSON TSV (Spreadsheets) TSV (Spreadsheets, hidden cols) XML

Рис. 3. Перечень уязвимых версий продуктов

Fig. 3. A list of vulnerable product versions

2.4. МОНИТОРИНГ УЯЗВИМОСТЕЙ

Функция мониторинга уязвимостей необходима для того, чтобы обнаруживать новые уязвимости в ПО и реагировать на них в максимально короткие сроки. Без такой функции организации могут оставаться уязвимыми для атак на неопределенный период времени, что может привести к краже данных, нарушению конфиденциальности, ущербу репутации и финансовым потерям [6].

Функция мониторинга уязвимостей позволяет непрерывно сканировать системы и приложения на предмет новых уязвимостей, которые могут появиться в результате обновлений ПО или изменений в конфигурации системы [7]. Кроме того, мониторинг уязвимостей также позволяет отслеживать изменения в степени критичности уже известных уязвимостей и немедленно реагировать на угрозы безопасности.

Таким образом, функция мониторинга уязвимостей является необходимой для обеспечения безопасности информации и защиты от внешних угроз [8].

Согласно требованиям Payment Card Industry Data Security Standard (PCI DSS) уязвимости, определенные в рамках процесса сканирования, должны быть исправлены в течение заданного временного периода, известного как «срок устранения уязвимостей» (vulnerability remediation timeframe).

PCI DSS требует, чтобы организации, обрабатывающие платежные данные, устанавливали срок устранения уязвимостей в течение 30 дней [9]. Этот срок отсчитывается от даты обнаружения уязвимости. Однако если уязвимость имеет высокий уровень критичности, PCI DSS требует ее устранения максимум в течение семи дней.

В случае если исправление уязвимости в указанный срок невозможно, организация должна разработать и представить план мер по снижению рисков, связанных с данной уязвимостью [10]. План должен быть одобрен уполномоченным представителем организации, а также должен содержать информацию о дополнительных контролях, принятых для минимизации рисков, связанных с данной уязвимостью.

Согласно PCI DSS проведение сканирования системы на уязвимости требуется в рамках выполнения требования 11.2.2. Оно определяет, что необходимо выполнять квартальное внешнее сканирование сетевых уязвимостей и внутреннее сканирование сетевых уязвимостей, проводимое как минимум ежегодно, а также после любых изменений в сетевой инфраструктуре. Это позволяет выявлять и устранять уязвимости до того, как злоумышленники могут использовать их для атаки на систему.

В России требования по обеспечению безопасности персональных данных регулируются Федеральным законом «О персональных данных» от 27 июля 2006 года № 152-ФЗ [11]. Согласно статье 19 этого закона операторы персональных данных должны принимать меры по защите персональных данных, в том числе обеспечивать их конфиденциальность и недоступность для третьих лиц.

При этом статья 19.2.4. ФЗ «О персональных данных» гласит о необходимости использования средств защиты информации при ее обработке и хранении. В число этих средств входят в том числе системы обнаружения и предотвращения вторжений, антивирусные программы и тесты на проникновение. В контексте данного вопроса тесты на проникновение могут рассматриваться как сканирование на уязвимости.

Кроме того, требования по обеспечению безопасности информации содержатся в законодательстве в области критической информационной инфраструктуры (КИИ). Например, Федеральный закон от 28 июля 2012 года № 187-ФЗ «О безопасности критической информационной инфраструктуры

Российской Федерации» устанавливает обязанность субъектов КИИ разрабатывать и реализовывать меры по защите своей информационной инфраструктуры. В данном случае сканирование на уязвимости может быть одной из таких мер.

Также Федеральный закон от 27 июля 2010 года № 224-ФЗ «О противодействии коррупции» устанавливает требования по обеспечению безопасности информации при ее обработке и передаче. Согласно статье 7.1 этого закона организации должны обеспечивать защиту информации, в том числе от несанкционированного доступа к ней.

3. ОЦЕНКА ЭФФЕКТИВНОСТИ АНАЛИЗАТОРА УЯЗВИМОСТЕЙ ПО

3.1. МЕТОДОЛОГИЯ СРАВНЕНИЯ

Для оценки эффективности работы анализатора ПО с функцией мониторинга было проведено сканирование сторонним сканером, выбор пал на OpenVAS.

OpenVAS работает по следующему алгоритму [12].

1. Подготовка настроек сканирования. Пользователь настраивает параметры сканирования, такие как диапазон IP-адресов для сканирования, порты, которые следует проверить, и типы уязвимостей, которые следует искать [13].

2. Сканирование сети. OpenVAS начинает сканирование сети, отправляя запросы на различные порты и протоколы, чтобы проверить, есть ли на устройстве уязвимости.

3. Сбор информации. OpenVAS анализирует ответы от устройств и определяет, какие уязвимости присутствуют. Он также собирает информацию об устройствах и запущенных на них сервисах [14].

4. Анализ результатов. OpenVAS использует базу данных уязвимостей для сравнения результатов сканирования и определения, какие уязвимости обнаружены. Каждая уязвимость получает уровень критичности, и пользователь получает отчет о найденных уязвимостях.

5. Разрешение уязвимостей. OpenVAS предоставляет советы по разрешению найденных уязвимостей, что помогает улучшить безопасность системы.

6. Повторное сканирование. OpenVAS может использоваться для регулярного повторного сканирования, чтобы убедиться, что уязвимости были решены и система защищена.

В целом, принцип работы сканера уязвимостей OpenVAS заключается в том, чтобы отправлять запросы на порты и протоколы на устройствах [15], чтобы определить наличие уязвимостей и собирать информацию об устройствах. Затем результаты анализируются и сравниваются с базой данных уяз-

вимостей, чтобы определить, какие уязвимости обнаружены, и дать советы по разрешению этих уязвимостей.

Тестирование проводилось на хосте с заведомо известными тремя уязвимостями в программном обеспечении:

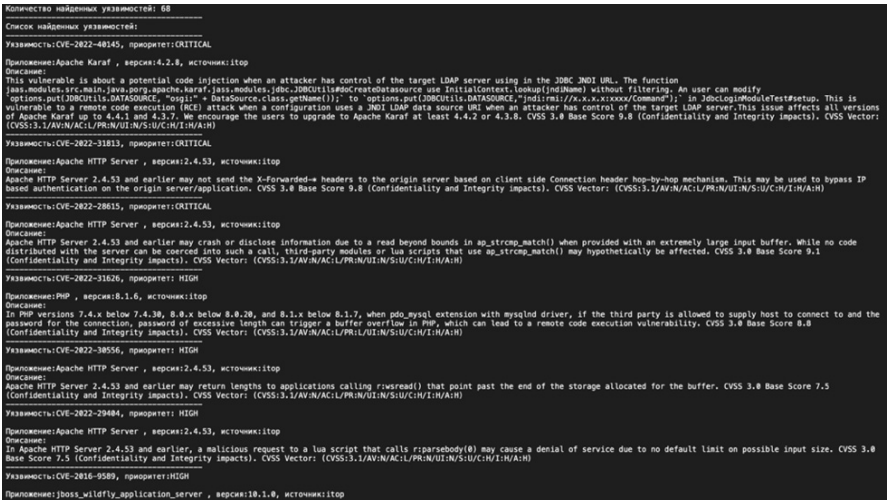
1) Apache HTTP Server – необходимость обновить версию;
2) OpenSSL – необходимость обновить версию;
3) PHP – необходимость обновить версию.

Критерии, по которым оценивается тестирование:

1) эффективность тестирования;
2) скорость тестирования;
3) нагрузка на сеть при тестировании;
4) оценка простоты использования;
5) дополнительная информация.

В результате работы OpenVAS сканером были обнаружены все заведомо известные уязвимости.

В результате работы анализатора уязвимостей ПО были определены не все уязвимости, что понизило точность анализатора. Это было связано с тем, что в ITOP нет информации относительно OpenSSL и его версии. Уязвимости, которые выявил анализатор уязвимостей ПО, показаны на рис. 4.



```
Количество найденных уязвимостей: 68
-----------------------------------------
Список найденных уязвимостей:
-----------------------------------------
Уязвимость:CVE-2022-40145, приоритет:CRITICAL

Приложение:Apache Karaf , версия:4.2.8, источник:itop
Описание:
This vulnerable is about a potential code injection when an attacker has control of the target LDAP server using in the JDBC JNDI URL. The function
jass.modules.src.main.java.porg.apache.karaf.jass.modules.jdbc.JDBCUtils#doCreateDatasource use InitialContext.lookup(jndiName) without filtering. An user can modify
`options.put(JDBCUtils.DATASOURCE, "osgi:" + DataSource.class.getName());` to `options.put(JDBCUtils.DATASOURCE,"jndi:rmi://x.x.x.x:xxxx/Command");` in JdbcLoginModuleTest#setup. This is
vulnerable to a remote code execution (RCE) attack when a configuration uses a JNDI LDAP data source URI when an attacker has control of the target LDAP server.This issue affects all versions
of Apache Karaf up to 4.4.1 and 4.3.7. We encourage the users to upgrade to Apache Karaf at least 4.4.2 or 4.3.8. CVSS 3.0 Base Score 9.8 (Confidentiality and Integrity impacts). CVSS Vector:
(CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H)
-----------------------------------------
Уязвимость:CVE-2022-31813, приоритет:CRITICAL

Приложение:Apache HTTP Server , версия:2.4.53, источник:itop
Описание:
Apache HTTP Server 2.4.53 and earlier may not send the X-Forwarded-* headers to the origin server based on client side Connection header hop-by-hop mechanism. This may be used to bypass IP
based authentication on the origin server/application. CVSS 3.0 Base Score 9.8 (Confidentiality and Integrity impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H)
-----------------------------------------
Уязвимость:CVE-2022-28615, приоритет:CRITICAL

Приложение:Apache HTTP Server , версия:2.4.53, источник:itop
Описание:
Apache HTTP Server 2.4.53 and earlier may crash or disclose information due to a read beyond bounds in ap_strcmp_match() when provided with an extremely large input buffer. While no code
distributed with the server can be coerced into such a call, third-party modules or lua scripts that use ap_strcmp_match() may hypothetically be affected. CVSS 3.0 Base Score 9.1
(Confidentiality and Integrity impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H)
-----------------------------------------
Уязвимость:CVE-2022-31626, приоритет: HIGH

Приложение:PHP , версия:8.1.6, источник:itop
Описание:
In PHP versions 7.4.x below 7.4.30, 8.0.x below 8.0.20, and 8.1.x below 8.1.7, when pdo_mysql extension with mysqlnd driver, if the third party is allowed to supply host to connect to and the
password for the connection, password of excessive length can trigger a buffer overflow in PHP, which can lead to a remote code execution vulnerability. CVSS 3.0 Base Score 8.8
(Confidentiality and Integrity impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:H)
-----------------------------------------
Уязвимость:CVE-2022-30556, приоритет: HIGH

Приложение:Apache HTTP Server , версия:2.4.53, источник:itop
Описание:
Apache HTTP Server 2.4.53 and earlier may return lengths to applications calling r:wsread() that point past the end of the storage allocated for the buffer. CVSS 3.0 Base Score 7.5
(Confidentiality and Integrity impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H)
-----------------------------------------
Уязвимость:CVE-2022-29404, приоритет: HIGH

Приложение:Apache HTTP Server , версия:2.4.53, источник:itop
Описание:
In Apache HTTP Server 2.4.53 and earlier, a malicious request to a lua script that calls r:parsebody(0) may cause a denial of service due to no default limit on possible input size. CVSS 3.0
Base Score 7.5 (Confidentiality and Integrity impacts). CVSS Vector: (CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H)
-----------------------------------------
Уязвимость:CVE-2016-9589, приоритет:HIGH

Приложение:jboss_wildfly_application_server , версия:10.1.0, источник:itop
```

Рис. 4. Результат, который выдал анализатор уязвимостей ПО

Fig. 4. The result of the software vulnerability analyzer

В отличие от OpenVAS анализатор никак не проявляет себя в корпоративной сети и не оказывает никакой положительной нагрузки, что не может не радовать, так как инструмент остается незаметным. Анализатор выдает базовую информацию, необходимую для ознакомления с уязвимостями. Свежий отчет приходит ежедневно в виде сообщения на корпоративную почту, где удобно отслеживать изменения.

Скорость выполнения анализа определяется скоростью выполнения программы, что, в свою очередь, в сотни раз быстрее, чем выполнение сканирования.

3.2. ИТОГИ СРАВНЕНИЯ

По итогам сравнения можно сделать следующие выводы.

1. Анализатор не оказывает нагрузку на сеть, в отличие от OpenVAS.
2. Анализатор работает по принципу сверки полученных данных из двух баз, в то время как OpenVAS сам ищет информацию о системе с помощью сканирования и отправки запросов.
3. Анализатор уязвимостей работает в сотни раз быстрее и предоставляет свежий ежедневный отчет о новых уязвимостях, в то время как деликатное сканирование OpenVAS даже одного хоста занимает значительное время.
4. Анализатор более прост и удобен в использовании; OpenVAS обладает функционалом, понимание которого в первое время может вызывать затруднение.
5. Анализатор и OpenVAS предоставляют описание уязвимости, но OpenVAS это делает в более удобной форме, в отличие от анализатора.

ЗАКЛЮЧЕНИЕ

Разработанный анализатор уязвимостей – удобный инструмент для инженеров информационной безопасности, позволяющий получать информацию о появлении новых уязвимостей или о статусе уже существующих на ежедневной основе. Интерфейс является гибким и дружественным к пользователю: он поддерживает ручное добавление источников информации, CVE, ресурсов для мониторинга, а также имеет удобную систему отчетов.

Анализатор не оказывает нагрузки на корпоративную сеть, не вызывает ложных срабатываний мониторинга, позволяет сократить время поиска уязвимостей до шести минут, уменьшив затрачиваемое время в 7 раз.

Данное решение идеально подойдет для домашнего использования, так как имеет функцию ручного заполнения, которая будет полезна для проведения инвентаризации ПО на компьютере с целью последующего мониторинга ситуации на ПК, а также решение может быть актуально для небольших корпоративных сетей.

СПИСОК ЛИТЕРАТУРЫ

1. Towards deep learning models resistant to adversarial attacks / A. Madry, A. Makelov, L. Schmidt, D. Tsipras, A. Vladu // 6th International Conference on Learning Representations, ICLR 2018. – Vancouver, BC, Canada, 2018.

2. *Mell P., Scarfone K., Romanosky S.* Common vulnerability scoring system // IEEE Security & Privacy. – 2006. – Vol. 4 (6). – P. 85–89.

3. *Прохоренко Н.А., Дронов В.А.* Python 3 и PyQt5. Разработка приложений. – 2-е., перераб. и доп. изд. – СПб.: БХВ-Петербург, 2018. – 832 с.

4. *Varghese S., Kurian R.* Identifying vulnerabilities in a website using Uniscan and Comparing Uniscan, Grabber, Nikto // Proceedings of the National Conference on Emerging Computer Applications (NCECA). – 2021. – Vol. 3 (1). – P. 225–229.

5. *Воеводин В.А., Ганенков Д.С., Королев С.Д.* Об актуальности применения программного средства MaxPatrol 8 для целей аудита автоматизированных систем // Радиоэлектронные устройства и системы для инфокоммуникационных технологий («РЭУС-2022»). – М., 2022. – С. 240–244.

6. *Sekharan S.S., Kandasamy K.* Profiling SIEM tools and correlation engines for security analytics // 2017 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET). – IEEE, 2017. – P. 717–721.

7. О мониторинге угроз и уязвимостей информационной системы / С.Т. Мамбетов, Е.Е. Бегимбаева, С.К. Джолдасбаев, Б.О. Куламбаев, Г.Н. Казбекова // Известия НАН РК. Серия физико-математическая. – 2022. – № 4. – С. 68–80.

8. *Walkowski M., Oko J., Sujecki S.* Vulnerability management models using a common vulnerability scoring system // Applied Sciences. – 2021. – Vol. 11. – P. 1–25. – DOI: 10.3390/app11188735.

9. *Ataya G.* PCI DSS audit and compliance // Information Security Technical Report. – 2010. – Vol. 15 (4). – P. 138–144.

10. *Kotyan S.* A reading survey on adversarial machine learning: Adversarial attacks and their understanding. – URL: https://arxiv.org/pdf/2308.03363.pdf (accessed: 08.09.2023).

11. *Борисенко О.В.* Анализ Федерального закона № 152-ФЗ «О персональных данных» // Электронное приложение к Российскому юридическому журналу. – 2012. – № 2. – С. 26–30.

12. *Рахалкар С.* Краткое руководство по тестированию на проникновение. – Нью-Йорк: Springer Science + Business Media, 2019. – 139 с.

13. *Vimala K., Fugkeaw S.* VAPE-BRIDGE: bridging OpenVAS results for automating metasploit framework // 14th International Conference on Knowledge and Smart Technology (KST). – IEEE, 2022. – P. 69–74.

14. *Кравченко А.С., Шарапова В.О.* О применении сканеров уязвимостей информационных систем в УИС // Техника и безопасность объектов уголовно-исполнительной системы. – Воронеж, 2022. – С. 243–244.

15. *Shahriar H., Zulkernine M.* Taxonomy and classification of automatic monitoring of program security vulnerability exploitations // Journal of Systems and Software. – 2011. – Vol. 84. – P. 250–269.

Мазуренко Виктор Александрович, студент магистратуры Новосибирского государственного технического университета. Основное направление научных исследований – техническая защита информации. E-mail: mazurenko.2017@stud.nstu.ru

Иванов Андрей Валерьевич, заведующий кафедрой защиты информации Новосибирского государственного технического университета. Основное направление научных исследований – информационная безопасность, техническая защита информации. E-mail: andrej.ivanov@corp.nstu.ru

DOI: 10.17212/2782-2230-2023-3-23-39

Development of a software analyzer with monitoring function*

V.A. Mazurenko[1], A.V. Ivanov[2]

[1] *Novosibirsk State Technical University, 20 Karl Marx Prospekt, Novosibirsk, 630087, Russian Federation, master's student of the Information Security Department. E-mail: mazurenko.2017@stud.nstu.ru*

[2] *Novosibirsk State Technical University, 20 Karl Marx Prospekt, Novosibirsk, 630073, Russian Federation, Candidate of Technical Sciences, Head of the Information Security Department. E-mail: andrej.ivanov@corp.nstu.ru*

In the realities of today's world, including the transition from WEB 2.0 to WEB 3.0. Every day there are new products that make human life more convenient. This applies to software that is used in business or everyday life. The emergence of new software from different levels of experts bear in themselves both positive and negative aspects. Following the emergence of new technologies, new threats also appear. Over 2022, 70 % of the most exploited vulnerabilities were due to software vulnerabilities. This article attempts to develop an effective software

* *Received 05 June 2023.*

vulnerability analyzer with monitoring functionality capable of detecting new vulnerabilities without causing a load on the network.
The article will describe the stages of the article's development, the principle of the program's work, information about the sources of obtaining data on the target system and on the vulnerability database. The monitoring process and operator's capabilities in obtaining information about the vulnerabilities will be described. To test the efficiency of the work, a test bench with vulnerabilities will be prepared on which the efficiency of the developed analyzer and the tool with similar functionality will be tested.

Keywords: vulnerabilities, analyzer development, software, vulnerability control, monitoring, cybersecurity, vulnerability scanner, CVE, NIST

REFERENCES

1. Madry A., Makelov A., Schmidt L., Tsipras D., Vladu A. Towards deep learning models resistant to adversarial attacks. *6th International Conference on Learning Representations, ICLR 2018*, Vancouver, BC, Canada, 2018.

2. Mell P., Scarfone K., Romanosky S. Common vulnerability scoring system. *IEEE Security & Privacy*, 2006, vol. 4 (6), pp. 85–89.

3. Prokhorenko N.A., Dronov V.A. *Python 3 i PyQt5. Razrabotka prilozhenii* [Python 3 and PyQt5. Application development]. 2nd ed., rev. St. Petersburg, BHV-Peterburg Publ., 2018. 832 p.

4. Varghese S., Kurian R. Identifying vulnerabilities in a website using Uniscan and Comparing Uniscan, Grabber, Nikto. *Proceedings of the National Conference on Emerging Computer Applications (NCECA)*, 2021, vol. 3 (1), pp. 225–229.

5. Voevodin V.A., Ganenkov D.S., Korolev S.D. [On the relevance of the software tool MaxPatrol 8 for the audit of automated systems]. *Radioelektronnye ustroistva i sistemy dlya infokommunikatsionnykh tekhnologii («REUS-2022»)* [The Radio-electronic devices and systems for the infocommunication technologies ("REDS-2022")]. Moscow, 2022, pp. 240–244. (In Russian).

6. Sekharan S.S., Kandasamy K. Profiling SIEM tools and correlation engines for security analytics. *2017 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)*. IEEE, 2017, pp. 717–721.

7. Mambetov S., Begimbaeva E., Joldasbaev S., Kulambaev B., Kazbekova G. O monitoringe ugroz i uyazvimostei informatsionnoi sisetmy [On monitoring of threats and vulnerabilities of information system]. *Izvestiya NAN RK. Seriya fiziko-matematicheskaya = News of the National Academy of Sciences of the Republic of Kazakhstan. Physico-matematical series*, 2022, no. 4, pp. 68–80.

8. Walkowski M., Oko J., Sujecki S. Vulnerability management models using a common vulnerability scoring system. *Applied Sciences*, 2021, vol. 11, pp. 1–25. DOI: 10.3390/app11188735.

9. Ataya G. PCI DSS audit and compliance. *Information Security Technical Report*, 2010, vol. 15 (4), pp. 138–144.

10. Kotyan S. *A reading survey on adversarial machine learning: Adversarial attacks and their understanding*. Available at: https://arxiv.org/pdf/2308.03363.pdf (accessed 08.09.2023).

11. Borisenko O.V. Analiz Federal'nogo zakona № 152-FZ «O personal'nykh dannykh» [Analysis of Federal Law N 152-FZ "On personal data"]. *Elektronnoe prilozhenie k Rossiĭskomu yuridicheskomu zhurnalu = Electronic supplement to Russian Juridical Journal*, 2012, no. 2, pp. 26–30.

12. Rahalkar S. *Kratkoe rukovodstvo po testirovaniyu na proniknovenie* [Quick start guide to penetration testing]. New York, Springer Science + Business Media, 2019. 139 p. (In Russian).

13. Vimala K., Fugkeaw S. VAPE-BRIDGE: bridging OpenVAS results for automating metasploit framework. *14th International Conference on Knowledge and Smart Technology (KST)*. IEEE, 2022, pp. 69–74.

14. Kravchenko A.S., Sharapova V.O. [On the use of vulnerability scanners of information systems in the penal system]. *Tekhnika i bezopasnost' ob"ektov ugolovno-ispolnitel'noi sistemy* [Technics and security of penal system facilities]. Voronezh, 2022, pp. 243–244. (In Russian).

15. Shahriar H., Zulkernine M. Taxonomy and classification of automatic monitoring of program security vulnerability exploitations. *Journal of Systems and Software*, 2011, vol. 84, pp. 250–269.

Для цитирования:

Мазуренко В.А., Иванов А.В. Разработка анализатора программного обеспечения с функцией мониторинга // Безопасность цифровых технологий. – 2023. – № 3 (110). – С. 23–39. – DOI: 10.17212/2782-2230-2023-3-23-39.

For citation:

Mazurenko V.A., Ivanov A.V. Razrabotka analizatora programmnogo obespecheniya s funktsiei monitoringa [Development of a software analyzer with monitoring function]. *Bezopasnost' tsifrovykh tekhnologii = Digital Technology Security*, 2023, no. 3 (110), pp. 23–39. DOI: 10.17212/2782-2230-2023-3-23-39.

БЕЗОПАСНОСТЬ ЦИФРОВЫХ ТЕХНОЛОГИЙ. – 2023. – № 3 (110). – 40–53

МЕТОДЫ И СИСТЕМЫ ЗАЩИТЫ ИНФОРМАЦИИ, ИНФОРМАЦИОННАЯ БЕЗОПАСНОСТЬ

УДК 004 DOI: 10.17212/2782-2230-2023-3-40-53

ВЫБОР И ОЦЕНКА ФУНКЦИОНАЛЬНОСТИ ЗАМКНУТЫХ СРЕД ВЫПОЛНЕНИЯ ПРОГРАММ («ПЕСОЧНИЦ») ДЛЯ ТЕСТИРОВАНИЯ И ДЕТЕКТИРОВАНИЯ ПОТЕНЦИАЛЬНО ОПАСНЫХ ФАЙЛОВ И ПРОГРАММ*

А.Б. АРХИПОВА[1], А.С. БЕРЕЖНОЙ[2]

[1] *630073, РФ, г. Новосибирск, пр. Карла Маркса, 20, Новосибирский государственный технический университет, доцент кафедры защиты информации. E-mail: arhipova@corp.nstu.ru*
[2] *630073, РФ, г. Новосибирск, пр. Карла Маркса, 20, Новосибирский государственный технический университет, инженер кафедры защиты информации. E-mail: kaf_zi@corp.nstu.ru*

Технологии «песочниц» предоставляют самые эффективные механизмы по защите от целевых атак и атак с использованием уязвимостей нулевого дня. Принцип работы «песочницы» заключается в том, что подозрительное программное обеспечение запускается в специально подготовленной для него среде, изолированной от остальной инфраструктуры. В работе выполнен анализ методов реализации «песочниц» для оценки и выбора функционала замкнутых сред. Рассмотрены локальные «песочницы», которые входят в состав многих антивирусов. Они обеспечивают изоляцию на базе частичной виртуализации файловой системы и реестра. Проанализированы сетевые «песочницы», которые имеют меньше ограничений, чем локальные, так как не снижают производительности компьютера пользователя и позволяют проверять потенциальные угрозы на различных операционных системах. Для выбора и оценки замкнутых сред в работе были выбраны две компании, которые предоставляют «песочницы» как в программно-аппаратном виде, так и с использованием готовых образов для сред виртуализации: FortiSandBox от компании Fortinet и PT Sandbox от компании Positive Technologies. Тестирование продуктов происходило в деморежиме, в виртуальной среде VMware workstation.

Ключевые слова: информационная безопасность, замкнутая среда, «песочница», тестирование продуктов, виртуализация, контейнеризация, функциональные возможности, гипервизор

* Статья получена 10 июня 2023 г.

ВВЕДЕНИЕ

При проведении целевых атак киберпреступники зачастую используют так называемые угрозы нулевого дня. Это вредоносные программы и эксплойты, которые только появились или были написаны специально для конкретной атаки и еще не попали в сигнатурные базы традиционных средств защиты. Подобные инструменты порой незаметны даже для наиболее современных антивирусов, межсетевых экранов и систем предотвращения вторжений. Решить задачу идентификации ранее неизвестных образцов вредоносных программ помогают сетевые или локальные «песочницы» – системы защиты, позволяющие оценить безопасность программного обеспечения путем его запуска и анализа в изолированном виртуальном окружении. Вместо применения сигнатурных методов поиска вредоносной активности «песочница» анализирует действия программы в среде, имитирующей типовые автоматизированные рабочие места и серверы организации. Стоит отметить, что правильная защита должна быть многослойной, и поэтому представители класса сетевых «песочниц» отнюдь не пренебрегают интеграцией с антивирусами, системами предотвращения вторжений и другими традиционными сигнатурными средствами.

1. ПОНЯТИЕ И МЕТОДЫ РЕАЛИЗАЦИИ «ПЕСОЧНИЦЫ»

Под «песочницей» (англ. sandbox) понимают ограниченную среду в компьютерной системе, предназначенную для исполнения потенциально опасных программ без их доступа к системным объектам операционной системы и иных приложений. «Песочницы» часто используют для запуска непротестированного кода, непроверенного кода из неизвестных источников, а также для запуска и обнаружения вирусов.

Технологии «песочниц» предоставляют самые эффективные механизмы по защите от целевых атак и атак с использованием уязвимостей нулевого дня. Принцип работы «песочницы» заключается в том, что подозрительное программное обеспечение (ПО) запускается в специально подготовленной для него среде, изолированной от остальной инфраструктуры. Известный и заведомо вредоносный код не попадает в «песочницу», поскольку он блокируется на уровне межсетевого экрана или сигнатурного анализа. А вот если у этих средств не набирается достаточного объема данных для принятия решения, файл направляется в «песочницу» [2].

Использование изолированных виртуальных машин для выполнения проверяемых объектов и эмуляция взаимодействия с пользователем позволяют подробно отследить характер выполняемых действий потенциально небез-

опасного ПО и решить, можно ли возвращать объект пользователю для запуска на рабочей станции. Интеграция анализа в «песочнице» с сигнатурным анализом и другими способами проверки в стандартных продуктах безопасности позволяет повысить эффективность выявления потенциальных угроз и улучшить от целевых атак.

Существуют различные типы «песочниц».

1. Локальные «песочницы». Входят в состав многих антивирусов. Они реализуют изоляцию на базе частичной виртуализации файловой системы и реестра. Вместо того чтобы создавать для каждого проверяемого процесса отдельную виртуальную машину, локальная «песочница» создает для них дубликаты объектов файловой системы и реестра [2]. Получается безопасная среда-«песочница» на компьютере пользователя. Если процесс изменит файлы или запишет что-то в реестр, изменятся лишь копии внутри песочницы, а реальные объекты не будут затронуты.

Изоляция файлов от основной системы обеспечивается с помощью контроля прав пользователей. Достоинством такого подхода является относительная простота реализации и невысокие затраты системных ресурсов. А в качестве недостатков можно отметить необходимость постоянной очистки контейнеров виртуализации для запуска каждого проверяемого файла.

Помимо этого, встречаются обходы такой реализации «песочницы», которые позволяют вредоносному коду перейти в основную систему [3].

Более защищенный вариант локальной «песочницы» предполагает создание отдельной виртуальной машины, копирующей рабочее окружение. Но затраты ресурсов на такой вариант, как правило, оказываются неприемлемо высокими, поэтому вместо него используются сетевые «песочницы», которые располагаются на выделенном сервере внутри сети компании или в облаке производителя антивирусного решения [4].

2. Сетевые «песочницы». Имеют меньше ограничений, чем локальные: они не снижают производительность компьютера пользователя и позволяют проверять потенциальные угрозы на различных операционных системах (ОС). Таким образом, система полностью изолирована от рабочего компьютера пользователя [5]. При необходимости такие «песочницы» могут эмулировать подключение к Интернету и работу со съемными носителями. При работе с сетевыми «песочницами» на компьютерах пользователей устанавливается агент – служба, которая отправляет попавшие под подозрения файлы в сетевую «песочницу». Передача файлов на анализ в облако занимает больше времени, чем при взаимодействии с сервером в сети компании.

В совокупности с длительностью анализа время ожидания результата может составить несколько минут, на протяжении которых запуск приложения будет «поставлен на паузу» до получения разрешения от «песочницы». В связи

с этим разработчики «песочниц» указывают максимальное время ожидания в SLA [6].

Вредоносное ПО, ориентированное на конкретную компанию, как правило, проверяет окружение, в котором оно запущено. И даже если ПО не содержит проверку на запуск в «песочнице», несоответствие окружения может привести к тому, что полезная нагрузка во время анализа не сработает и файл будет считаться безопасным. Чтобы избежать такой ситуации, нужно, чтобы рабочая среда, которую эмулирует «песочница», максимально точно соответствовала рабочим станциям реальных пользователей [7].

В случае с облачными «песочницами» добиться такого соответствия сложнее, в то время как загрузка образа рабочей станции на сервер компании не составляет сложности. Главное, чтобы выбранный вариант сервера-«песочницы» поддерживал работу с пользовательскими образами. Другими словами, чтобы максимально приблизить конфигурацию виртуальных машин внутри «песочницы» к корпоративной среде, нужно иметь возможность тонко настраивать их содержимое: изменять настройки операционной системы, редактировать перечень установленных языков, драйверов периферийных устройств, устанавливать дополнительный либо нестандартный софт и даже управлять содержимым рабочего стола, поскольку всё это и многое другое может расцениваться киберзлоумышленниками как условие для запуска либо незапуска вредоносных инструкций. Использование же стандартизированных образов для разворачивания виртуальных машин внутри «песочниц» легко отслеживается и позволяет применить механизмы обхода детектирования в «песочницах». «Песочницы» поддерживают возможность загрузки пользовательских образов вычислительных машин, что на практике неоднократно показывало более высокую эффективность при обнаружении вредоносного ПО в сравнении с «песочницами» производителей, использующих стандартизированные образы ВМ для анализа.

Анализ литературы показал, что исходя из соображений необходимости обеспечения максимальной эффективности на сегодняшний день предпочтение следует отдать сетевому варианту. Реализации облачных «песочниц» на сегодняшний день ни у одного из производителей не поддерживают в качестве среды тестирования пользовательских образов виртуальных машин, точно отражающих инфраструктуру конкретного заказчика [8]. Облачные же «песочницы» могут рассматриваться в качестве более доступной по стоимости альтернативы либо если инфраструктура компании территориально распределена. В этом случае затраты на обеспечение необходимой сетевой маршрутизации могут превысить выгоду от разницы между облачным и серверным решением компании.

2. ВЫБОР И ОЦЕНКА ЗАМКНУТЫХ СРЕД

Для выбора и оценки замкнутых сред были выбраны две компании, которые предоставляют «песочницы» как в программно-аппаратном виде, так и с использованием готовых образов для сред виртуализации: FortiSandBox от компании Fortinet и PT Sandbox от компании Positive Technologies. Тестирование продуктов происходило в деморежиме, в виртуальной среде VMware workstation.

1. FortiSandbox – это замкнутая среда от Fortinet, предназначенная для обнаружения потенциально сложных атак в «песочнице». Она осуществляет анализ и выявление потенциально опасных угроз в локальной сети компании с эмуляцией кода в виртуальной защищенной среде. Песочница FortiSandbox, является как программно-аппаратным комплексом FortiSandBox 1000D/3000E/3500D, виртуальным устройством (FortiSandbox-VM), так и облачным решением с интегрированным межсетевым экраном FortiGate (FortiSandBox Cloud). FortiSandbox обеспечивает детонацию угроз, обнаружение и минимизацию последствий атак.

Для развертывания «песочницы» есть несколько подходов. В самом простом изолированном режиме «песочница» подключается к SPAN-порту коммутатора. Такое подключение лучше всего подходит для защиты от сложных угроз к уже имеющимся видам защиты. При таком подходе есть возможность загружать подозрительные файлы на проверку с помощью веб интерфейса. В продвинутом режиме внедрения возможно перехватывать или передавать контент для анализа из других продуктов компании Fortinet в FortiSandbox. Данный метод является наиболее эффективным, чтобы своевременно блокировать известные атаки без потери в производительности сети и обеспечивать мгновенное восстановление и генерацию отчетов от этих устройств. Есть возможность настроить оповещение о блокировке атак или о генерации отчетов через электронную почту.

Функциональные возможности FortiSandbox

В FortiSandbox используется метод эмуляции кода для раскрытия поведения и обнаружения неизвестных угроз и целенаправленных атак. Для оценки угроз исполняемых файлов, zip-архивов и других файловых расширений используется виртуальная среда, которая виртуализирует рабочую среду ОС и программное обеспечение. FortiSandbox имеет поддержку создания собственных образов, в которых может быть определенный набор программ, используемых заказчиком.

Анализ файла, который передается в изолированную среду, – затратный процесс как по времени, так и по ресурсам системы, что может значительно

снизить производительность и ограничить число проверенных файлов. Для оптимизации это процесса подозрительные файлы проходят предварительную фильтрацию: происходит процесс антивирусного сканирования и сравнения сигнатур файлов с базой. Если в ходе первой проверки антивирусным сканером не было подтверждено наличие или отсутствие угрозы в файле, то образец передается для дальнейшего анализа в «песочницу». В случае когда файл оказывается вредоносным, то «песочница» загружает данные о найденных угрозах в отчет, содержащий сведения о перехваченных пакетах, логах, трассировке, самих файлах и скриншотах.

Начальный доступ для настройки осуществляется с помощью средств ssh или com-порта для программно-аппаратного комплекса. В таком режиме выполняется только начальная настройка системы, поскольку настройка всего функционала происходит внутри графического интерфейса.

Список файлов, поддерживаемых по умолчанию в FortiSandbox, представлен в табл. 1 [20].

Таблица 1

Table 1

Поддержка файлов для анализа FortiSandbox

File support for analysis FortiSandbox

Тип файла	Расширения
Исполняемые	BAT, CMD, DLL, EXE, JAR, MSI, PS1, UPX, WSF и VBS. Поскольку не все DLL-файлы могут быть выполнены в пределах виртуальной машины, рекомендуется для этого типа файлов включить предварительную фильтрацию
Архивные	7Z, ARB, BZIP, BZIP2, CAB, EML, GZIP, LZW, RAR, TAR, XZ и т. д.
Скриптовые	JavaScript/HTML, Batch Script, Power Shell, VBS
Microsoft Office	Word, Excel, PowerPoint, Outlook и т. д.
Adobe	PDF, SWF, Flash
Статические веб-файлы	HTML, JS, URL, LNK
Файлы ОС Android	APK

Повышение производительности и минимизация ложных срабатываний достигается за счет использования белого и черного списков. Эти списки хранят хеш, контрольные суммы, а также список демонов, с которых загружаются файлы.

Отметим, что сетевое устройство безопасности FortiSandbox является полнофункциональной сетевой «песочницей», интегрированной с анализом угроз и механизмами безопасности продуктов Fortinet, таких как FortiGate, что позволяет в режиме реального времени обеспечить безопасность контролируемой сети и конечных точек на каждом уровне защиты.

Физические устройства FortiSandbox предназначены для защиты вычислительных сетей крупных компаний и корпораций. Благодаря различным вариантам встраивания устройства могут распределять нагрузку и защищать удаленные сегменты сети. Использование таких устройств обусловлено их высокой производительностью. Однако использование дополнительных функций безопасности влечет за собой увеличение себестоимости решения. Для небольших компаний Fortinet предлагает альтернативный вариант сетевой «песочницы» без снижения скорости обнаружения и реагирования на угрозы безопасности – FortiSandbox Cloud.

Достоинства FortiSandbox:

- высокая производительность и скорость анализа сетевого трафика;
- наличие большого числа сетевых портов и возможность расширения за счет дополнительных интерфейсов;
- поддержка нескольких сценариев развертывания;
- интеграция с другими продуктами Fortinet;
- поддержка масштабируемости и кластеризации;
- Microsoft Office (только для WINXPVM и WIN7X86VM);
- управление единым средством централизованного управления FortiSandbox через веб-интерфейс.

Недостатки FortiSandbox:

- отсутствие русской локализации;
- использование дополнительных функций безопасности и образов виртуальных машин требует приобретение дополнительных лицензий.

2. PT Sandbox. «Песочница» PT Sandbox от компании Positive Technologies, имеет возможность гибко настраивать виртуальные среды так, чтобы они не отличались от реальных рабочих станций с ОС. Есть возможность загружать в них помимо стандартного офисного пакета программ тот или иной пакет специализированного ПО, используемого в организации. Эта особенность наряду с глубоким комплексным анализом файлов позволяет продукту обеспечивать защиту от целевых и массовых атак, сопровождающихся вредоносными программами и угрозами нулевого дня.

Замкнутая среда PT Sandbox анализирует объекты, которые попадают в инфраструктуру компании из разных источников, а именно: по электронной почте, скачиваются из сети, размещаются в корпоративных файловых хранилищах или загружаются пользователями через веб-интерфейс продукта.

Любой файл, отправленный в «песочницу», проходит комплексную проверку, которая включает в себя статистический и поведенческий анализ. По ее результатам формируется итоговый отчет. Статистический анализ предусматривает антивирусную проверку и анализ файла с помощью уникальных экспертных правил.

Для поиска уже известных вредоносных программ используется несколько антивирусов, которые поставляются вместе с продуктом. Поведенческий анализ предусматривает запуск файла в виртуальной среде и глубокое изучение его действий. Продукт не только анализирует сам файл и связанные с ним артефакты, но и проверяет генерируемый им трафик, выявляя вредоносную сетевую активность.

Функциональные возможности PT Sandbox

Продукт имеет возможность использовать два типа режима для анализа объекта: пассивный и блокирующий. В зависимости от типа анализа PT Sandbox будет или блокировать опасные файлы и письма, или отслеживать их. В пассивном режиме файлы и письма отправляются на проверку одновременно с их дальнейшей передачей. Все дальнейшие решения о распространении файла и их реализации на возникающие угрозы принимают операторы безопасности по результатам анализа.

В блокирующем режиме на время проверки продукт останавливает дальнейшую передачу файла или письма до получения результата. Если в задании на проверку выявлен хотя бы один опасный файл, то блокируются все файлы из этого задания. В этом режиме могут работать следующие источники: палка-шлюз, почтовый сервер с установленным агентом и почтовый сервер в режиме фильтрации.

Выявленные особенности «песочницы» после изучения функциональности и документации по продукту заключаются в следующем.

1. Обнаружение целевых атак (в том числе и атаки нулевого дня) на конкретное ПО. Продукт поддерживает глубокую настройку виртуализации для анализа и загрузки в них того ПО, которое реально используется в компании и может быть мишенью для злоумышленников.

2. Выявление угроз в сетевом трафике, включая шифрованный. Продукт анализирует весь трафик, который генерируется в процессе анализа файла, в том числе скрытый под TLS. Это дает возможность выявлять опасную сетевую активность, которая внешне может быть не связана с конкретным файлом.

3. Выявление скрытых в инфраструктурах ранее неизвестных угроз. Этого удается достичь с помощью регулярного ретроспективного анализа. После обновления баз знаний продукт выполняет автоматическую перепроверку уже обработанных файлов и находит угрозы, которые не детектировались на момент предыдущего исследования. Взаимодействие с продуктом происходит через веб-интерфейс, который делится на главное меню, расположенное в верхней части страницы, и рабочую область.

При работе PT Sandbox уделяется особое внимание качеству детектирования угроз. Помимо стандартной настройки для «песочниц» функциональности по анализу файлов, в продукте реализована возможность настройки виртуальных сред, проверка генерируемого файлом сетевого трафика (в том числе шифрованного) и автоматический ретроспективный анализ.

PT Sandbox позволяет осуществлять проверку сжатых файлов и архивов. По умолчанию декомпрессия перед проверкой отключена (табл. 2).

Таблица 2

Table 2

Поддерживаемые методы сжатия файлов PT Sandbox

Supported PT Sandbox File Compression Methods

Вид компрессии	Расширения
Gzip	.gz
Compress	.z
Bzip2	.bz2
LZMA	.lz, .lzma
LZMA2	.xz

Достоинства PT Sandbox:

- большое число источников файлов и писем для проверки (почтовые серверы, ICAP-сервер, папки-шлюзы, общие папки и др.), включая проверку по требованию пользователя через веб-интерфейс и с помощью почтовой службы;
- комплексная проверка файлов, сочетающая в себе статический и динамический виды анализа и обеспечивающая высокий уровень детектирования благодаря уникальным правилам PT ESC;

- поддержка анализа сетевого трафика и наличие ретроспективного анализа;
- подробный обзор действий файла, связанных с ним артефактов и сетевой активности, детальная визуализация операций в виртуальной среде (граф, видеозапись);
- удобный графический интерфейс для мониторинга и настройки заданий на проверку;
- интеграция с продуктами Positive Technologies и антивирусными средствами сторонних производителей;
- поддержка горизонтального масштабирования.

После анализа двух «песочниц» от разных компаний можно сказать, что «песочницы» отличаются незначительно, так как основные функции реализованы единообразно.

ЗАКЛЮЧЕНИЕ

Мировой рынок традиционно представлен большим числом крупных производителей, предлагающих продукты разного уровня – как по стоимости, так и по качеству защиты. Выбор конкретного решения существенно зависит от того, какие средства уже используются в инфраструктуре. Так, например, если в ней установлены продукты Check Point, Fortinet, Palo Alto или McAfee, то весьма логичным будет внедрение «песочницы», FortiSandbox. Это обеспечит бесшовную интеграцию продуктов. Если же целью является построение платформенно независимого комплекса, то, возможно, имеет смысл присмотреться к предложениям Positive Technologies. Учитывая ниспадающий объем целенаправленных атак на инфраструктуры организаций, можно утверждать, что сетевые «песочницы» актуальны для отечественного заказчика, и дальнейший спрос на них будет только расти. Это, несомненно, должно положительно сказаться на развитии российского рынка таких продуктов.

СПИСОК ЛИТЕРАТУРЫ

1. ГОСТ Р ИСО/МЭК 56938–2016. Защита информации при использовании технологий виртуализации: введ. 2017–06–01. – М.: Изд-во стандартов, 2017. – 30 с.

2. Песочница (Sandbox) / Лаборатория Касперского. – URL: https://www.kaspersky.ru/enterprise-security/wiki-section/products/sandbox (дата обращения: 01.09.2023).

3. *Казыханов А.А., Попов К.Г.* Анализ использования песочниц для работы с зловредами // Символ науки. – 2016. – № 7-2. – С. 63–64.

4. Что такое песочница? Как работает облачная песочница? – URL: https://www.avast.ru/business/resources/what-is-sandboxing#pc (дата обращения: 04.09.2023).

5. *Солопов М.И., Чиркин Е.С.* Анализ степени изолированности защищенных сред средств обеспечения информационной безопасности // Гаудеамус. – 2012. – № 20. – С. 157–159.

6. Red Hat Enterprise Virtualization. Обзор продукта / Бюро Соломатина. – М., 2010. – URL: http://www.pcweek.ru/upload/iblock/eba/bureausolomatina-4.pdf (дата обращения: 04.09.2023).

7. *Баранов А.В., Николаев Д.С.* Использование контейнерной виртуализации в организации высокопроизводительных вычислений // Программные системы: теория и приложения. – 2016. – № 1 (28). – С. 117–134.

8. *Метельков А.Н.* Моделирование сценариев кибератак в киберполигонах // Вестник Санкт-Петербургского университета ГПС МЧС России. – 2023. – № 2. – С. 161–176.

9. *Метельков А.Н.* Киберучения: зарубежный опыт защиты критической инфраструктуры // Правовая информатика. – 2022. – № 1. – С. 51–60.

10. The current state of the art and future of European cyber range ecosystem / C. Virág, J. Cegan, T. Lieskovan, M. Merialdo // 2021 IEEE International Conference on Cyber Security and Resilience (CSR). – Rhodes, Greece, 2021. – P. 390–395.

11. *Davies J., Margat S.* A survey of cyber ranges and testbeds. DSTO-GD-0771 / Cyber Electronic Warfare Division, Defense Science and Technology Organization DSTO. – Edinburgh, Australia, 2013.

12. *Brilingaite A., Bukauskas L., Kutka E.* Development of an educational platform for cyber defence training // Proceedings of the 16th European Conference on Cyber Warfare and Security (ECCWS’17). – Dublin, Ireland, 2017. – P. 73–81.

Архипова Анастасия Борисовна, доцент кафедры защиты информации Новосибирского государственного технического университета. Основное направление научных исследований – математическое моделирование в информационной безопасности, оценка качества социально значимой деятельности. E-mail: arhipova@corp.nstu.ru

Бережной Антон Сергеевич, инженер кафедры защиты информации Новосибирского государственного технического университета. Область научных интересов – информационная безопасность. E-mail: kaf_zi@corp.nstu.ru

DOI: 10.17212/2782-2230-2023-3-40-53

Selection and evaluation of functionality of closed program execution environments (sandboxes) for testing and detection of potentially dangerous files and programs*

A.B. Arkhipova[1], A.S. Bereznoy[2]

[1] *Novosibirsk State Technical University, 20 Karl Marx Prospekt, Novosibirsk, 630073, Russian Federation, Associate Professor of the Department of Information Security. E-mail: arhipova@corp.nstu.ru*
[2] *Novosibirsk State Technical University, 20 Karl Marx Prospekt, Novosibirsk, 630073, Russian Federation, engineer of the Department of Information Security. E-mail: kaf_zi@corp.nstu.ru*

Sandbox technologies provide the most effective mechanisms to protect against targeted and zero-day attacks. The principle of the sandbox is that suspicious software runs in a specially prepared environment for it, isolated from the rest of the infrastructure. The work analyzed the methods for implementing sandboxes to assess and select the functionality of closed environments. We considered local sandboxes, which are part of many antiviruses. They implement isolation based on partial file system virtualization and registry. Analyzed network sandboxes, which have fewer restrictions than local ones, since they do not reduce the performance of the user's computer and allow you to check potential threats on various operating systems. To select and evaluate closed environments, two companies were selected in the work that provide sandboxes both in software and hardware form and using ready-made images for virtualization environments: Fortinet's FortiSandBox and Positive Technologies' PT Sandbox. Product testing took place in demo mode, in a VMware workstation virtual environment.

Keywords: information security, confined environment, sandbox, product testing, virtualization, containerization, functionality, hypervisor

REFERENCES

1. GOST R ISO/MEK 56938–2016. *Zashchita informatsii pri ispol'zovanii tekhnologii virtualizatsii* [State standard R ISO/IEC 56938–2016. Protect information with virtualization technologies]. Moscow, Standards Publ., 2017. 30 p.

2. *Pesochnitsa* [Sandbox]. Kaspersky Lab. (In Russian). Available at: https://www.kaspersky.ru/enterprise-security/wiki-section/products/sandbox (accessed 01.09.2023).

* *Received 10 June 2023.*

3. Kazykhanov A.A., Popov K.G. Analiz ispol'zovaniya pesochnits dlya raboty s zlovredami [Analysis of the use of sandboxes to work with malware]. *Simvol nauki = Symbol of science*, 2016, no. 7-2, pp. 63–64.

4. *Chto takoe pesochnitsa? Kak rabotaet oblachnaya pesochnitsa?* [What is a sandbox? How does the cloud sandbox work?]. Available at: https://www.avast.ru/business/resources/what-is-sandboxing#pc (accessed 04.09.2023).

5. Solopov M.I., Chirkin E.S. Analiz stepeni izolirovannosti zashchishchennykh sred sredstv obespecheniya informatsionnoi bezopasnosti [Analysis of the degree of isolation of protected environments of information security tools]. *Gaudeamus*, 2012, no. 20, pp. 157–159. (In Russian).

6. Byreau Solomatina. *Red Hat Enterprise Virtualization. Obzor produkta* [Red Hat Enterprise Virtualization. Product overview]. Moscow, 2010. Available at: http://www.pcweek.ru/upload/iblock/eba/bureausolomatina-4.pdf (accessed 04.09.2023).

7. Baranov A.V., Nikolaev D.S. Ispol'zovanie konteinernoi virtualizatsii v organizatsii vysokoproizvoditel'nykh vychislenii [The use of container virtualization in the organization of high-performance computing]. *Programmnye sistemy: teoriya i prilozheniya = Software and Systems*, 2016, no. 1 (28), pp. 117–134.

8. Metelkov A.N. Modelirovanie stsenariev kiberatak v kiberpoligonakh [Modeling cyberattack scenarios in cyberpolygons]. *Vestnik Sankt-Peterburgskogo universiteta GPS MChS Rossii = Vestnik Saint-Petersburg university of State fire service of EMERCOM of Russia*, 2023, no. 2, pp. 161–176.

9. Metel'kov A.N. Kiberucheniya: zarubezhnyi opyt zashchity kriticheskoi infrastruktury [Cyber exercises: foreign experience in protecting critical infrastructure]. *Pravovaya informatika = Legal Informatics*, 2022, no. 1, pp. 51–60.

10. Virág C., Cegan J., Lieskovan T., Merialdo M. The current state of the art and future of European cyber range ecosystem. *2021 IEEE International Conference on Cyber Security and Resilience (CSR)*, Rhodes, Greece, 2021, pp. 390–395.

11. Davies J., Margat S. *A survey of cyber ranges and testbeds*. DSTO-GD-0771. Cyber Electronic Warfare Division, Defense Science and Technology Organization DSTO, Edinburgh, Australia, 2013.

12. Brilingaite A., Bukauskas L., Kutka E. Development of an educational platform for cyber defence training. *Proceedings of the 16th European Conference on Cyber Warfare and Security (ECCWS'17)*, Dublin, Ireland, 2017, pp. 73–81.

Для цитирования:

Архипова А.Б., *Бережной А.С.* Выбор и оценка функциональности замкнутых сред выполнения программ («песочниц») для тестирования и детектирования потенциально опасных файлов и программ // Безопасность цифровых технологий. – 2023. – № 3 (110). – С. 40–53. – DOI: 10.17212/2782-2230-2023-3-40-53.

For citation:

Arkhipova A.B., Bereznoy A.S. Vybor i otsenka funktsional'nosti zamknutykh sred vypolneniya programm («pesochnits») dlya testirovaniya i detektirovaniya potentsial'no opasnykh failov i programm [Selection and evaluation of functionality of closed program execution environments (sandboxes) for testing and detection of potentially dangerous files and programs]. *Bezopasnost' tsifrovykh tekhnologii = Digital Technology Security*, 2023, no. 3 (110), pp. 40–53. DOI: 10.17212/2782-2230-2023-3-40-53.

БЕЗОПАСНОСТЬ ЦИФРОВЫХ ТЕХНОЛОГИЙ. – 2023. – № 3 (110). – 54–66

МЕТОДЫ И СИСТЕМЫ ЗАЩИТЫ ИНФОРМАЦИИ, ИНФОРМАЦИОННАЯ БЕЗОПАСНОСТЬ

УДК 004.056.5 DOI: 10.17212/2782-2230-2023-3-54-66

РАЗРАБОТКА СИСТЕМЫ КОНТРОЛЯ ИНЦИДЕНТОВ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ*

Е.Ю. СОЛДАТОВ[1], В.В. СЕЛИФАНОВ[2], М.А. КУВШИНОВ[3]

[1] *630108, РФ, г. Новосибирск, ул. Плахотного, 10, Сибирский государственный университет геосистем и технологий, лаборант. E-mail: wilgieforz@mail.ru*

[2] *630073, РФ, г. Новосибирск, пр. Карла Маркса, 20, Новосибирский государственный технический университет, старший преподаватель. E-mail: sfo1@mail.ru,* ORCID ID: 0000-0002-6691-5647

[3] *630073, РФ, г. Новосибирск, пр. Карла Маркса, 20, Новосибирский государственный технический университет, ассистент кафедры защиты информации. E-mail: kuvshinovma@gmail.com*

Целью настоящей работы является разработка системы контроля инцидентов информационной безопасности, отвечающей требованиям к регистрируемой информации и инцидентам информационной безопасности. В статье поднимается вопрос необходимости создания системы, которая позволит контролировать инциденты информационной безопасности, проводить оценку существующих решений на рынке, формировать требования для системы, выбор и обоснование технологий при разработке. После регистрации инцидента есть возможность взаимодействия с другими узлами сети, блокировки IP-адреса источника на МЭ веб-сервера (iptables), закрытия сетевого порта, блокировки доменов на прокси-сервере. Также реализован функционал просмотра данной информации в веб-интерфейс системы. В статье описана и обоснована необходимость создания и внедрения такой системы в информационную сеть. Для достижения поставленной цели был проведен анализ рынка аналогичных систем, а также проблем при их сопровождении. Исходя из анализа было разработано техническое задание с последующей реализацией программного кода, проведена апробация системы и реализовано несколько сценариев. В работе был проведен анализ методических документов, связанных с инцидентами информационной безопасности, разработано техническое задание, реализован программный код, проведена апробация. В результате было разработано программное обеспечение «Система контроля инцидентов информационной безопасности».

Ключевые слова: информационная безопасность, инцидент информационной безопасности, расследование, кибербезопасность, система обнаружения вторжений, система предотвращения вторжений, межсетевой экран, прокси-сервер

* Статья получена 03 июля 2023 г.

ВВЕДЕНИЕ

В связи с тем, что атаки на информационные системы организаций с каждым годом становятся всё чаще, масштабнее и серьезнее, возрастают масштабы негативных последствий, возникает потребность своевременно реагировать и регистрировать инциденты информационной безопасности, направленные на информационную систему. Для реализации этой задачи используется специальное программное обеспечение – система контроля инцидентов информационной безопасности IMS (Incident Management Software). На рынке представлено множество популярных решений для реализации данной задачи, но практически все из них – реализации иностранных государств.

Президентом Российской Федерации в 2022 году было утверждено два указа:

– Указ Президента РФ от 30.03.2022 № 166 о том, что с 31 марта 2022 года были введены ограничения на приобретение иностранного оборудования и программного обеспечения для субъектов КИИ, а также услуги по использованию такого ПО без согласования с уполномоченным органом [1];

– Указ Президента РФ от 01.05.2022 № 250 о том, что с 1 января 2025 года организациям запрещается использовать средства защиты информации, произведенные в недружественных государствах [2].

Также с 13 февраля 2023 года вступил в силу приказ ФСБ России № 77, утверждающий порядок взаимодействия операторов с ГосСОПКА на информационных ресурсах РФ, включая информирование ФСБ России о компьютерных инцидентах, повлекших неправомерную передачу (предоставление, распространение, доступ) персональных данных [3].

Таким образом, можно утверждать, что обеспечение отечественной системой контроля инцидентов информационной безопасности в защищаемую информационную систему – задача, актуальная для каждой организации.

Целью настоящей работы является создание системы контроля инцидентов информационной безопасности, которая должна решить проблемы, упомянутые выше, и будет проста во внедрении и сопровождении.

Для достижения данной цели были поставлены следующие задачи:

1) разработка требований к создаваемой системе;

2) разработка системы контроля инцидентов информационной безопасности;

3) внедрение системы и ее апробация.

1. РАЗРАБОТКА ТРЕБОВАНИЙ К СИСТЕМЕ КОНТРОЛЯ ИНЦИДЕНТОВ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

Управление инцидентами кибербезопасности не является линейным процессом. Это цикл, состоящий из подготовки, обнаружения, сдерживания, ликвидации и восстановления. Заключительный этап состоит из извлечения уроков из инцидента с целью улучшения процесса и подготовки к будущим возможным сценариям. После каждого инцидента следует организовать обзорное собрание с участием команды специалистов отдела ИБ, руководства организации и каждого отдельного сотрудника, чтобы сделать соответствующие выводы и проанализировать эффективность плана реагирования на инциденты и стратегии на каждом его этапе. Жизненный цикл инцидента информационной безопасности представлен на рис. 1.

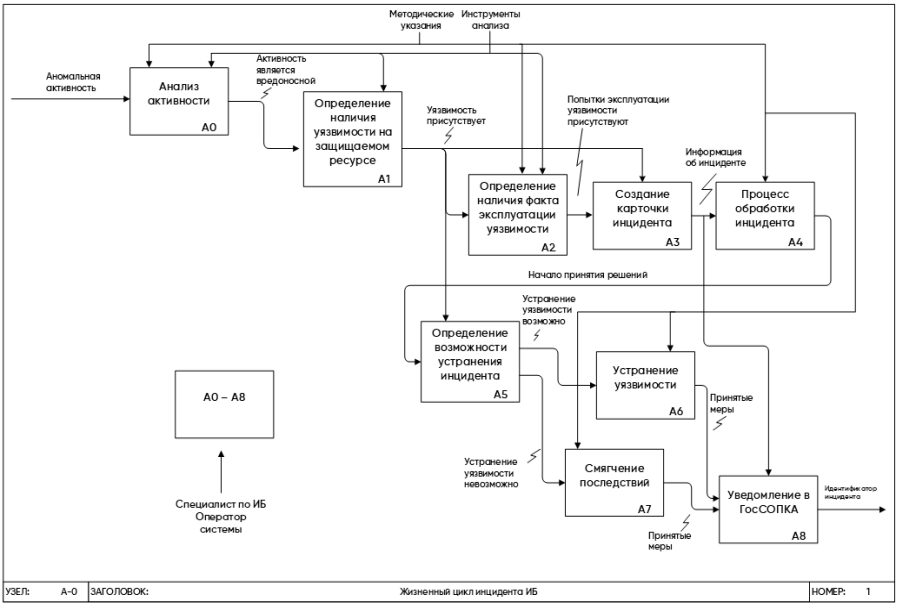


Рис. 1. Жизненный цикл инцидента ИБ

Fig. 1. Life cycle of an information security incident

Современные системы контроля инцидентов информационной безопасности, как ни странно, выполняют функции регистрации инцидентов ИБ и

позволяют удобно отслеживать, в каком состоянии они находятся. Инцидент после отработки сохраняется в базе данных для возможного дальнейшего извлечения уроков из него.

Современный рынок характеризуется сравнительно небольшим выбором систем – как коммерческих, так и с открытым исходным кодом. Первые отличаются высокой стоимостью, оправданной наличием сертификатов соответствия техническим требованиям, предъявляемым регуляторами в сфере информационной безопасности: если в сети обрабатывается информация, подлежащая обязательной защите в соответствии с действующим законодательством, то допускается использование только сертифицированных средств защиты, прошедших процедуру оценки ФСБ России и ФСТЭК России. Второй тип решений, состоящих из приложений с открытым исходным кодом, не менее распространен ввиду своей доступности. Такие системы зачастую используются в средствах защиты информации, к которым не предъявляются строгие требования регуляторов (в частности, в отношении сертификации средств защиты информации), или служат основой для создания коммерческих решений, разрабатываемых частными компаниями. Главной проблемой рассматриваемых систем с открытым исходным кодом является то, что они не специализируются на инцидентах информационной безопасности: практически все из них – реализация для Service Desk (техническая поддержка).

Ввиду невозможности использования дорогостоящих коммерческих систем с целью разбора функционала для реализации задач данной работы далее будут рассмотрены некоторые популярные коммерческие решения из обзорных статей на сайтах производителей.

Security Vision Incident Response Platform (IRP / SOAR) – российское программное обеспечение для автоматизации действий по реагированию на инциденты кибербезопасности. Этот программный продукт позволяет автоматически выполнять дежурные назначения в режиме реального времени. Модуль IRP позволяет автоматически реагировать на инциденты кибербезопасности (попытки внедрения ВПО, попытки эксплуатации уязвимостей, активность ВПО в сети, нарушение политик и др.), благодаря чему снижается риск человеческого фактора и ошибок операторов, ответственных за реагирование на инциденты информационной безопасности [4].

Security Vision Security Operation Center (SOC) – российский программный продукт, предназначенный для создания собственного глобального центра мониторинга информационной безопасности в масштабах организации, города, страны или мира. Это программное обеспечение обладает полным функционалом для построения и визуализации процессов информационной без-

опасности в режиме реального времени на масштабируемой карте. Благодаря этому операторы SOC получают полную информацию и аналитику в режиме online, а значит, могут оперативно реагировать на инциденты любой сложности. В данном программном продукте есть функция обмена информацией об инцидентах с государственными и коммерческими центрами мониторинга, такими как ГосСОПКА, ФинЦЕРТ и другие [5].

R-Vision SOAR (ранее R-Vision IRP) – это программный продукт для автоматизации деятельности по мониторингу, регистрации и реагированию на инциденты информационной безопасности. Позволяет получать данные об инцидентах с SIEM, СЗИ и других источников. Собственные правила и источники Threat Intelligence позволяют корректно среагировать на событие информационной безопасности, зарегистрировать инцидент и подробно его описать.

Благодаря функционалу R-Vision SOAR специалисты центра мониторинга имеют возможность удобно взаимодействовать и обмениваться информацией с ГосСОПКА, ФинЦЕРТ и MSS-провайдерами об инцидентах в информационной сети [6].

TheHive – это масштабируемая IRP, тесно интегрированная с MISP (платформой для обмена информацией о вредоносных программах), предназначена для упрощения работы SOC, CSIRT, CERT и любых специалистов по информационной безопасности, имеющих дело с инцидентами кибербезопасности, которые необходимо расследовать и оперативно на них реагировать. Считается лидером среди программ с открытым исходным кодом.

TheHive позиционируется как продукт 4-in-1 и содержит в себе:

– ядро системы, в котором происходит основной рабочий процесс;

– интегрированная система поиска Cortex, благодаря которой осуществляется анализ событий и механизм активного реагирования;

– агрегатор каналов узлов Hipposampe, объединяющий индикаторы компрометации из множества открытых источников в кластере Elacticsearch;

– REST API клиент TheHive4Py для написания скриптов на Python под любые нужды.

Программный код TheHive реализован на языке Scala и на текущий момент поддерживает ELK 5/6 для хранения логов. Клиентская часть системы реализована на JavaScript с использованием платформы для разработки веб-приложений AngularJS в паре с набором инструментов Bootstrap [7].

Данная система проста как в установке, так и в использовании с теми возможностями «из коробки», которых достаточно для выполнения большинства базовых задач.

2. РАЗРАБОТКА СИСТЕМЫ КОНТРОЛЯ ИНЦИДЕНТОВ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

Что такое стек технологий для веб-разработки? Стек веб-разработки относится к комбинации инструментов и технологий, используемых для создания веб-приложения. Стеки технологий веб-разработки включают в себя все языки программирования, фреймворки, библиотеки, серверы, программное обеспечение, используемые веб-разработчиками для написания проекта. Хотя веб-разработчики могут свободно создавать стеки в соответствии со своими потребностями, некоторые технологические стеки настолько хорошо работают вместе, что стали стандартами в индустрии веб-разработки. Они отлично себя зарекомендовали, так как позволяют работать более эффективно, устранять лишние ошибки и ускорять процесс разработки.

Для реализации программного кода и архитектуры приложения за основу был взят стек технологий PERN (PostgreSQL, Express, React, Node.js) [8]. LAMP – программное обеспечение с открытым исходным кодом, которое обычно устанавливается на сервер для отображения динамических веб-сайтов и веб-приложений. Обозначает операционную систему Linux с установленным веб-сервером Apache, базой данных MySQL для хранения информации сайта и его пользователей и PHP для обработки динамического контента.

Ruby on Rails – фреймворк для построения веб приложений на языке Ruby, использующий реляционные и NoSQL БД (MySQL, MariaDB, PostgreSQL и MondoDB).

Для реализации полноценного веб-приложения технологический стек PERN подходит лучше всего, так как язык, на котором реализуется программный код – JavaScript. Также у данного стека отличная масштабируемость, низкие системные требования для серверного оборудования и высокая производительность благодаря асинхронному выполнению кода, что позволяет обрабатывать запросы от тысячи пользователей.

Само веб-приложение будет построено на клиент-серверной архитектуре (рис. 2). Это означает, что приложение разделено на два звена – клиент и сервер. Клиент и сервер можно считать отдельным программным обеспечением. Поскольку серверная часть приложения должна выполнять множество запросов от различных клиентов, то ее необходимо размещать на выделенном сервере с высокой производительностью и пропускной способностью.

В качестве операционной системы на серверной части веб-приложения будет использоваться отечественная операционная система специального назначения Astra Linux Special Edition (Релиз «Смоленск») [9], в качестве БД выступает PostgreSQL.

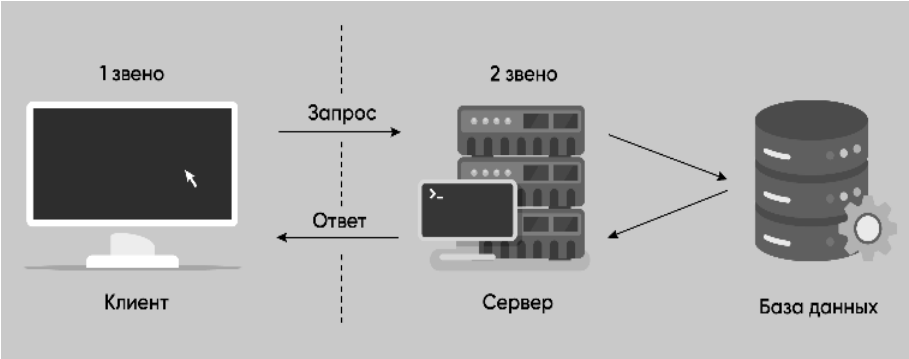


Рис. 2. Клиент-серверная архитектура

Fig. 2. Client-server architecture

Для реализации программного кода и архитектуры приложения за основу был взят стек технологий PERN (PostgreSQL, Express, React, Node.js) (рис. 3).

Рис. 3. Стек технологий PERN

Fig. 3. PERN technology stack

В результате дашборд и карточка инцидента выглядят следующим образом (рис. 4 и 5).

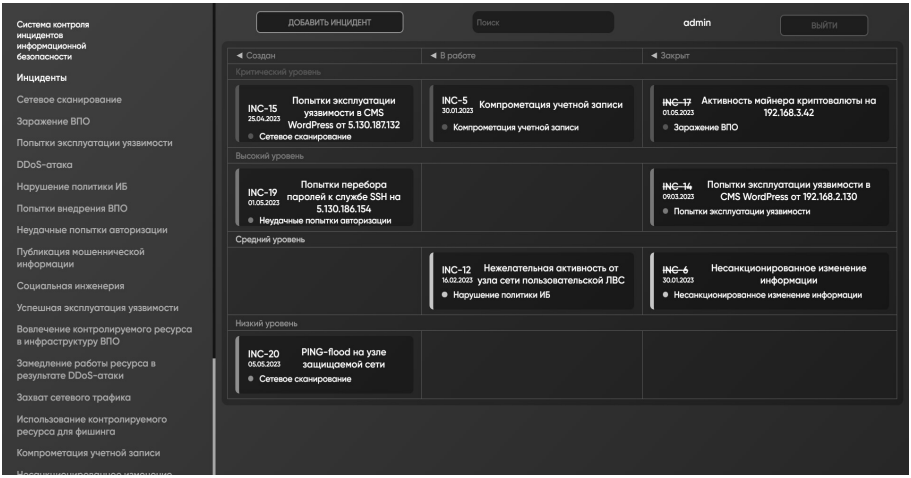


Рис. 4. Графическая панель с инцидентами

Fig. 4. Dashboard with incidents

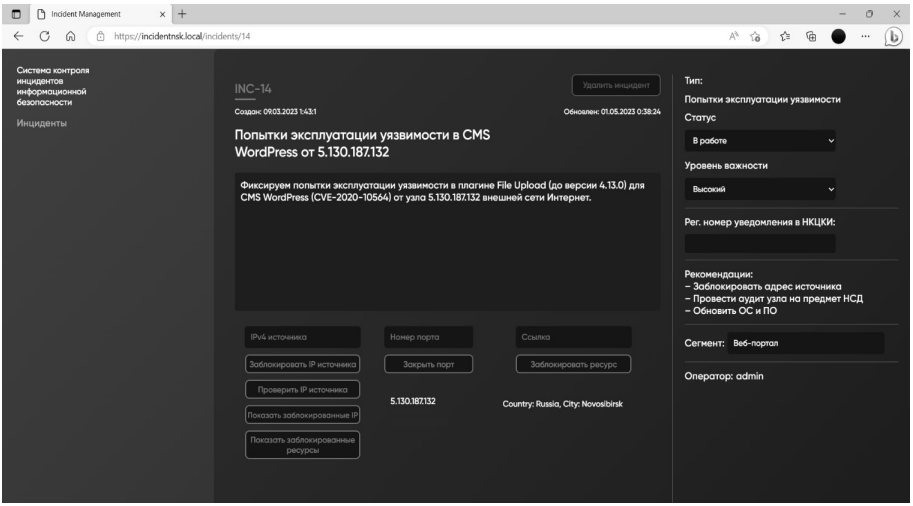


Рис. 5. Карточка инцидента ИБ

Fig. 5. Information security incident card

3. ВНЕДРЕНИЕ В КОМПЬЮТЕРНУЮ СЕТЬ

Информационная система была внедрена в компьютерную сеть с такими СЗИ, как ViPNet IDS NS, xFirewall и PT AF. Также вместе с ней был внедрен кэширующий проксисервер Squid для реализации функции блокировки URI сегмента Office Users (рис. 6).

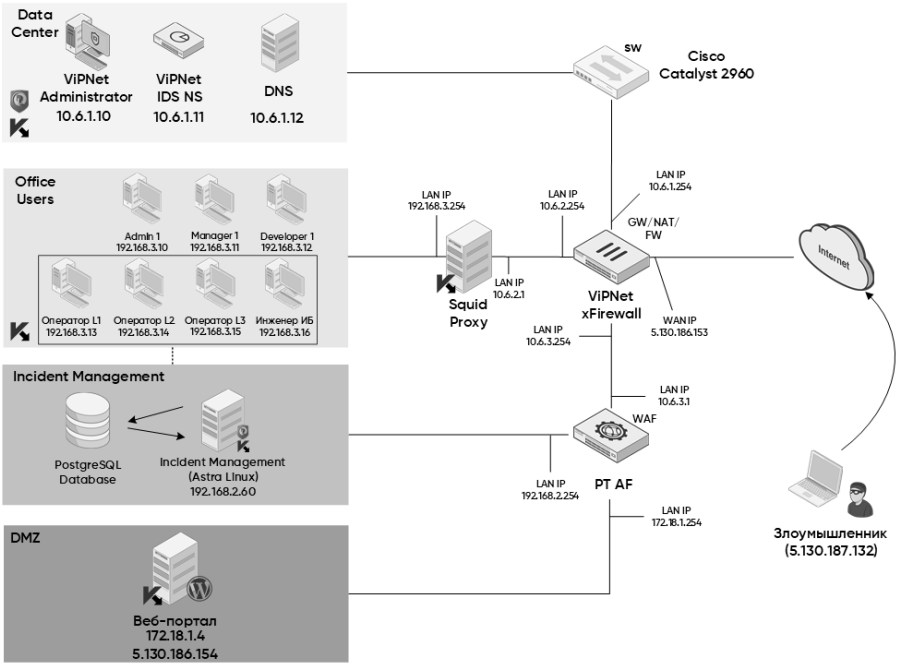


Рис. 6. Логическая схема компьютерной сети

Fig. 6. Logic diagram of a computer network

ЗАКЛЮЧЕНИЕ

Разработанная система контроля инцидентов информационной безопасности решает упомянутые ранее проблемы и позволяет эффективно регистрировать инциденты, собирает информацию в одном месте, передает его в ГосСОПКА (НКЦКИ) [10] с помощью электронной почты, а также имеет модуль управления инцидентом, позволяющий блокировать нежелательные IP, ресур-

сы и закрывать сетевые порты на встроенном межсетевом экране дистрибутива Linux iptables.

СПИСОК ЛИТЕРАТУРЫ

1. Указ Президента РФ от 30.03.2022 № 166 «О мерах по обеспечению технологической независимости и безопасности критической информационной инфраструктуры Российской Федерации» // Собрание законодательства РФ. – 2022. – № 14. – Ст. 2242.

2. Указ Президента РФ от 01.05.2022 № 250 «О дополнительных мерах по обеспечению информационной безопасности Российской Федерации» // Собрание законодательства РФ. – 2022. – № 18. – Ст. 3058.

3. Приказ ФСБ РФ от 13.02.2023 № 77 «Об утверждении порядка взаимодействия операторов с ГосСОПКА на информационные ресурсы, включая информирование ФСБ России о компьютерных инцидентах, повлекших неправомерную передачу (представление, распространение, доступ) персональных данных» // Официальный интернет-портал правовой информации (ФСБ России). – Опубл. 20.02.2023. – № 14.

4. Security Vision IRP: сайт. – URL: https://www.securityvision.ru/products/irp/ (дата обращения: 04.09.2023).

5. Security Vision SOC: сайт. – URL: https://www.securityvision.ru/products/soc/ (дата обращения: 04.09.2023).

6. R-Vision SOAR: сайт. – URL: https://rvision.ru/products/soar (дата обращения: 04.09.2023).

7. TheHive Project: сайт. – URL: https://thehive-project.org/ (дата обращения: 04.09.2023).

8. *Сафин А.М., Кадыров К.А.* Стек разработки приложений PERN // Актуальные вопросы общества, науки и образования: сборник статей Международной научно-практической конференции. – Пенза, 2022. – С. 95–97.

9. Astra Linux: сайт. – URL: https://astralinux.ru/ (дата обращения: 04.09.2023).

10. Приказ ФСБ РФ от 19.06.2019 № 282 «Об утверждении Порядка информирования ФСБ России о компьютерных инцидентах, реагирования на них, принятия мер по ликвидации последствий компьютерных атак, проведенных в отношении значимых объектов критической информационной инфраструктуры Российской Федерации» // Официальный интернет-портал правовой информации (ФСБ России). – Опубл. 17.07.2019. – № 49.

Солдатов Егор Юрьевич, лаборант кафедры защиты информации Сибирского государственного университета геосистем и технологий. В настоящее время специализируется в области информационной безопасности. E-mail: wilgieforz@mail.ru

Селифанов Валентин Валерьевич, старший преподаватель кафедры защиты информации Новосибирского государственного технического университета. В настоящее время специализируется в области информационной безопасности. E-mail: sfo1@mail.ru

Кувшинов Максим Алексеевич, ассистент кафедры защиты информации Новосибирского государственного технического университета. В настоящее время специализируется в области информационной безопасности. E-mail: kuvshinovma@gmail.com

DOI: 10.17212/2782-2230-2023-3-54-66

Development of the information security incident control system*

E.Yu. Soldatov[1], V.V. Selifanov[2], M.A. Kuvshinov[3]

[1] *Siberian State University of Geosystems and Technologies, 10 K. Plakhotnogo, Novosibirsk, 630108, Russian Federation, laboratory assistant of the Department of Information Security. E-mail: wilgieforz@mail.ru*
[2] *Novosibirsk State Technical University, 20 Karl Marx Prospekt, Novosibirsk, 630073, Russian Federation, senior lecturer of the Department of Information Security. E-mail: sfo1@mail.ru*
[3] *Novosibirsk State Technical University, 20 Karl Marx Prospekt, Novosibirsk, 630073, Russian Federation, assistant of the Department of Information Security. E-mail: kuvshinovma@gmail.com*

The purpose of this work is to develop an information security incident control system that meets the requirements for recorded information and information security incidents. the article raises the question of the need to create a system that will allow you to control information security incidents. Evaluation of existing solutions on the market, formation of requirements for the system. Selection and justification of technologies during development. After registering an incident, it is possible to interact with other network nodes, block the source IP address on the ME web server (iptables), close the network port, block domains on the proxy server. The functionality of viewing this information in the system web interface is also implemented. The article describes and substantiates the need to create and implement such a system in the information network. To achieve this goal, an analysis of the market for similar systems, as well as problems in their maintenance, was carried out. Based on the analysis, a technical task was developed with the subsequent implementation of the program code. The system was then tested and several work scenarios were implemented. In the work, an analysis of methodologi-

* *Received 03 July 2023.*

cal documents related to information security incidents was made, a technical task was developed, a program code was implemented, and testing was carried out. As a result, the software "Information Security Incident Control System" was developed.

Keywords: information security, information security incident, investigation, cybersecurity, intrusion detection system, intrusion prevention system, firewall, proxy server

REFERENCES

1. Ukaz Prezidenta RF ot 30.03.2022 № 166 «O merakh po obespecheniyu tekhnologi-cheskoi nezavisimosti i bezopasnosti kriticheskoi informatsionnoi infrastruktury Rossiiskoi Federatsii» [Decree of the President of the Russian Federation of March 30, 2022 No. 166 "On measures to ensure the technological independence and security of the critical information infrastructure of the Russian Federation"]. *Sobranie zakonodatel'stva Rossiiskoi Federatsii = Collection of the legislation of the Russian Federation*, 2022, no. 14, art. 2242.

2. Ukaz Prezidenta RF ot 01.05.2022 № 250 «O dopolnitel'nykh merakh po obespecheniyu informatsionnoi bezopasnosti Rossiiskoi Federatsii» [Decree of the President of the Russian Federation of 01.05.2022 Nn. 250 "On additional measures to ensure information security of the Russian Federation"]. *Sobranie zakonodatel'stva Rossiiskoi Federatsii = Collection of the legislation of the Russian Federation*, 2022, no. 18, art. 3058.

3. [Order of the Federal Security Service of the Russian Federation dated February 13, 2023 No. 77 "On approval of the procedure for the interaction of operators with NCCCI on information resources, including informing the FSB of Russia about computer incidents that resulted in the unlawful transfer (presentation, distribution, access) of personal data"]. *Ofitsial'nyi internet-portal pravovoi informatsii (FSB Rossii)* [Official Internet portal of legal information (FSB of Russia)], 2023, no. 14. (In Russian).

4. *Security Vision IRP*. Website. (In Russian). Available at: https://www.securityvision.ru/products/irp/ (accessed 04.09.2023).

5. *Security Vision SOC*. Website. (In Russian). Available at: https://www.securityvision.ru/products/soc/ (accessed 04.09.2023).

6. *R-Vision SOAR*. Website. (In Russian). Available at: https://rvision.ru/products/soar (accessed 04.09.2023).

7. *TheHive Project*. Website. Available at: https://thehive-project.org/ (accessed 04.09.2023).

8. Safin A.M., Kadyrov K.A. [PERN application development stack]. *Aktual'nye voprosy obshchestva, nauki i obrazovaniya* [Actual issues of society, science and education]. Collection of materials of the International scientific and practical conference, Penza, 2022, pp. 95–97. (In Russian).

9. *Astra Linux*. Website. (In Russian). Available at: https://astralinux.ru/ (accessed 09.04.2023).

10. [Order of the Federal Security Service of the Russian Federation of June 19, 2019 No. 282 "On approval of the Procedure for informing the FSB of Russia about computer incidents, responding to them, taking measures to eliminate the consequences of computer attacks carried out in relation to significant objects of the critical information infrastructure of the Russian Federation"]. *Ofitsial'nyi internet-portal pravovoi informatsii (FSB Rossii)* [Official Internet portal of legal information (FSB of Russia)], 2019, no. 49. (In Russian).

Для цитирования:

Солдатов Е.Ю., Селифанов В.В., Кувшинов М.А. Разработка системы контроля инцидентов информационной безопасности // Безопасность цифровых технологий. – 2023. – № 3 (110). – С. 54–66. – DOI: 10.17212/2782-2230-2023-3-54-66.

For citation:

Soldatov E.Yu., Selifanov V.V., Kuvshinov M.A. Razrabotka sistemy kontrolya intsidentov informatsionnoi bezopasnosti [Development of the information security incident control system]. *Bezopasnost' tsifrovykh tekhnologii = Digital Technology Security*, 2023, no. 3 (110), pp. 54–66. DOI: 10.17212/2782-2230-2023-3-54-66.

БЕЗОПАСНОСТЬ ЦИФРОВЫХ ТЕХНОЛОГИЙ. – 2023. – № 3 (110). – 67–82

МЕТОДЫ И СИСТЕМЫ ЗАЩИТЫ ИНФОРМАЦИИ, ИНФОРМАЦИОННАЯ БЕЗОПАСНОСТЬ

УДК 004.056.5 DOI: 10.17212/2782-2230-2023-3-67-82

ВОПРОСЫ АУДИТА ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ*

А.В. СИТСКАЯ[1], В.В. СЕЛИФАНОВ[2], П.А. ЗВЯГИНЦЕВА[3]

[1] *630108, РФ, 195112, г. Санкт-Петербург, БЦ «Золотая долина», площадь Карла Фаберже, 8, лит. Б, 3 эт., оф. 302, менеджер по продажам, департамент информационной безопасности. E-mail: Anastasiya.Sitskaya@softline.com*
[2] *630073, РФ, г. Новосибирск, пр. Карла Маркса, 20, Новосибирский государственный технический университет, доцент. E-mail: sfo1@mail.ru*
[3] *630108, РФ, г. Новосибирск, ул. Плахотного, 10, Сибирский государственный университет геосистем и технологий, доцент кафедры информационной безопасности. E-mail: polinasgugit@mail.ru*

Сегодня информационная безопасность властвует абсолютно во всех сферах деятельности человека. Нарушение безопасности всегда влечет за собой последствия, но в зависимости от структуры они могут быть незаметными или же, наоборот, могут остановить деятельность организации или даже страны. Вопрос обеспечения информационной безопасности объектов критической информационной инфраструктуры сегодня стал наиболее обсуждаем. Государственные регуляторы разработали множество нормативно правовых документов, которые содержат в себе требования к системе защиты объектов критической информационной инфраструктуры различных уровней значимости. Однако никто так и не создал ни одной методики, которая бы показала реальное состояние информационной безопасности объекта критической информационной инфраструктуры. Как следствие, как организация, так и регуляторы видят только общую картину состояния системы защиты, что, в свою очередь, создает множество уязвимых мест, которыми может воспользоваться злоумышленник для осуществления атаки. Вследствие атаки объекта критической информационной инфраструктуры вред может быть нанесен не только организации, но и людям, могут быть нарушены рабочие процессы всего государства. Именно поэтому вопрос создания такой методики становится не просто актуальным, но и необходимым как для внутреннего контроля самой организации, так и для автоматизации работы регуляторов при проведении очередных проверок. Методика покажет не только качественную оценку состояния системы защиты, но и возможные уязвимые места организации, которые необходимо устранить для повышения эффективности всей защиты.

* Статья получена 20 июля 2023 г.

Ключевые слова: информационная безопасность, критическая информационная инфраструктура, объект критической информационной инфраструктуры, аудит, аудит информационной безопасности, система защиты информации, разработка методики оценки, защита информационных систем

ВВЕДЕНИЕ

Объекты критической информационной инфраструктуры (далее – объект КИИ), определяемые согласно Федеральному закону от 26 июля 2017 г. № 187 «О безопасности критической информационной инфраструктуры Российской Федерации» (далее – Федеральный закон № 187) [1] как «информационные системы, автоматизированные системы управления субъектов критической информационной инфраструктуры», подлежат защите. Однако приведенное определение нельзя считать полным, так как оно требует дополнительного пояснения. Таким пояснением становится определение субъекта критической информационной инфраструктуры. Согласно Федеральному закону № 187 субъект КИИ – это «государственные органы, государственные учреждения, российские юридические лица и (или) индивидуальные предприниматели, которым на праве собственности, аренды или на ином законном основании принадлежат информационные системы, информационно-телекоммуникационные сети, автоматизированные системы управления, функционирующие в сфере здравоохранения, науки, транспорта, связи, энергетики, государственной регистрации прав на недвижимое имущество и сделок с ним, банковской сфере и иных сферах финансового рынка, топливно-энергетического комплекса, в области атомной энергии, оборонной, ракетно-космической, горнодобывающей, металлургической и химической промышленности, российские юридические лица и (или) индивидуальные предприниматели, которые обеспечивают взаимодействие указанных систем или сетей». В совокупности эти два определения дают достаточно полное понимание составляющих КИИ.

НОРМАТИВНО-ПРАВОВАЯ БАЗА

Исходя из определений видно, что данной области защиты информации (далее – ЗИ) должно быть уделено особое внимание. При нарушении требований ЗИ может быть нарушено функционирование одной из областей КИИ, а как следствие, полная остановка и угроза как сфере деятельности государства, так и жизни людей отдельных регионов страны. Именно серьезные по-

следствия при нарушении деятельности КИИ стали причиной для разработки ряда законодательных и нормативных правовых актов (далее – НПА), обязательных для соблюдения объектами КИИ.

Для чего же на самом деле нужна методика оценки? Она нужна, чтобы не просто в качественном или количественном виде увидеть состояние защиты системы, но и точно увидеть все уязвимости системы, которые может использовать злоумышленник для реализации эффективной атаки на информационные системы (далее – ИС) объекта КИИ. Понятно, что в зависимости от потенциала нарушителя последствия его атаки будут различными.

Если при нарушении деятельности объекта КИИ главные потери понесет сам объект (например, денежные потери), то внимание к такому объекту будет минимально и, собственно, защита своей инфраструктуры будет его проблемой. Однако если простой объекта КИИ в теории может вызвать серьезные последствия для государства, то защита такого предприятия приобретает высокую значимость. А за нарушения будут отвечать сотрудники предприятия согласно Уголовному и Административному кодексам Российской Федерации. Естественно, предприятие становится заинтересованным в реализации сильной и эффективной системы ЗИ.

Однако, казалось бы, система ЗИ построена и работает. Но как это может понять сторонний наблюдатель? Как наиболее наглядно показать, не углубляясь в техническую часть работы системы ЗИ, что она работает, и работает на необходимом уровне? Для решения данного вопроса нам необходимо в первую очередь правильно провести категорирование.

Категорирование объектов КИИ проводится согласно Постановлению Правительства Российской Федерации от 8 февраля 2018 г. № 127 «Об утверждении Правил категорирования объектов критической информационной инфраструктуры Российской Федерации, а также перечня показателей критериев значимости объектов критической информационной инфраструктуры РФ и их значений» (далее – Постановление Правительства) [2]. Согласно Постановлению Правительства категорирование объектов КИИ в обязательном порядке включает в себя следующие пункты [2]:

а) определение процессов, указанных в пункте 3 настоящих Правил, в рамках выполнения функций (полномочий) или осуществления видов деятельности субъекта критической информационной инфраструктуры;

б) выявление управленческих, технологических, производственных, финансово-экономических и (или) иных процессов в рамках выполнения функций (полномочий) или осуществления видов деятельности субъектов критической информационной инфраструктуры, нарушение и (или) прекращение которых может привести к негативным социальным, политическим, экономиче-

ским, экологическим последствиям, последствиям для обеспечения обороны страны, безопасности государства и правопорядка (далее – критические процессы);

в) определение объектов критической информационной инфраструктуры, которые обрабатывают информацию, необходимую для обеспечения критических процессов, и (или) осуществляют управление, контроль или мониторинг критических процессов;

г) формирование перечня объектов критической информационной инфраструктуры, подлежащих категорированию (далее – перечень объектов);

д) оценка в соответствии с перечнем показателей критериев значимости масштаба возможных последствий в случае возникновения компьютерных инцидентов на объектах критической информационной инфраструктуры;

е) присвоение каждому из объектов критической информационной инфраструктуры одной из категорий значимости либо принятие решения об отсутствии необходимости присвоения им одной из категорий значимости.

После того как объекты КИИ получили категории и их одобрил регулятор, актуальным становится вопрос: как защитить? Мы помним, что основную роль в формировании требований к технической защите организации КИИ взял на себя такой регулятор, как ФСТЭК России. Главным его НПА в части технической защиты стал приказ ФСТЭК России от 25 декабря 2017 г. № 239 «Об утверждении Требований по обеспечению безопасности значимых объектов критической информационной инфраструктуры Российской Федерации» (далее – приказ ФСТЭК № 239) [3], содержащий в себе 17 групп мер, которые по своей сути обязательны к использованию, однако приказ предусматривает варианты, когда при наличии обоснования их можно исключить.

АУДИТ ОБЪЕКТА КИИ

Как же должен строиться аудит информационной безопасности (далее – ИБ), чтобы организация могла увидеть реальную картину состояний системы ЗИ? Проанализировав литературу [4–10], можно выделить три основных этапа аудита, исполнение которых отражает реальное состояние КИИ:

1) проверка полноты и достаточности существующей документации в организации по требованиям НПА в области защиты информации;

2) оценка качества документации в области ЗИ;

3) проверка реализации технической составляющей системы ЗИ согласно существующей НПА в области защиты информации.

Далее рассмотрим нюансы каждого этапа аудита.

Первый этап аудита заключается в проверке наличия всех существующих документов в области ИБ. На данном этапе должное внимание уделяется не качеству документа, а лишь факту его существования. То есть главная задача аудитора – составить актуальный перечень существующей документации организации. Для этого аудитор руководствуется документами [1–3, 11–15].

Так как у аудитора в наличии только НПА, которые регламентируют отдельные области ЗИ, при этом в большей степени не говорят конкретно, в каких документах и что должно содержаться, ему необходимо разработать перечень необходимых документов для стабильного функционирования организации. На базе такого перечня создается чек-лист для аудитора, который наглядно покажет наличие или отсутствие ряда документов.

На первом этапе формируется:

1) актуальный чек-лист необходимых документов организации;
2) количественная оценка наличия ряда документов в области ЗИ.

Поскольку чек-лист по своей сути создается индивидуально для каждого предприятия, то и количественная оценка должна напрямую зависеть от чек-листа, а точнее, от количества существующих в организации документов и от количества необходимых документов. Естественно, чек-лист имеет градацию оценок: 1 – документ в наличии, 0 – документ отсутствует. Из вышесказанного получаем следующую формулу:

$$E_{ch1} = \frac{\sum_{i=1}^{n} d_i}{n}, \qquad (1)$$

где E_{ch1} – оценка наличия необходимого перечня документов; i – номер документа чек-листа; n – количество документов в чек-листе; d_i – оценка наличия i-го документа.

На втором этапе аудитор уделяет внимание качеству документа. Для этого он должен не только знать НПА в области ЗИ, но углубиться в особенности построенной КИИ организации, чтобы оценить как актуальность документа, так и его достаточность. Пункты чек-листа второго этапа полностью идентичны чек-листу первого этапа, однако есть существенная разница. В данном чек-листе градация оценок состоит из трех уровней:

- 0 – если документ неактуален или отсутствует;
- 0,5 – если документ требует доработки;

- 1 – если актуален и полностью соответствует реальному функционированию системы ЗИ.

Такая градация необходима, чтобы как можно подробнее оценить состояние системы ЗИ.

Результаты второго этапа будут заключены в следующих пунктах:

1) заполненный чек-лист актуальности и полноты документов организации;

2) количественная оценка достаточности документов в области ЗИ.

После заполнения чек-листа второго этапа для получения объективной оценки состояния документации необходимо обратиться к формуле

$$E_{ch2} = \frac{\sum_{i=1}^{n} dp_i}{n}, \qquad (2)$$

где E_{ch2} – оценка полноты содержания необходимого перечня документов; i – номер документа чек-листа; n – количество документов в чек-листе; dp_i – полнота i-го документа.

Третий этап аудита заключается в проведении технической проверки состояния системы ЗИ КИИ. На данном этапе аудитор должен достоверно убедиться в наличии и работоспособности технических средств ЗИ, которые используются организацией в построенной системе ЗИ.

Порядок проведения третьего пункта аудита подробно описан в статьях [16, 17], в которых представлен алгоритм построения графа, показанного на рис. 1. Граф подробно демонстрирует ход третьего этапа аудита и разбиение его на три параллели, построен на основе анализа и систематизации рейтинга SANS CIS Controls 8 и практического опыта ФСТЭК России. Каждая вершина графа сопоставлена с группами мер в соответствии с приказом ФСТЭК России № 239.

При оценивании каждой группы мер аудитор должен помнить, что не все меры равноценны между собой, а значит, использование мер разных уровней злоумышленником нанесет разный уровень последствий. Именно поэтому меры имеют разную градацию оценок.

Первая градация будет применима для критичных мер, которые составляют первые два уровня графа. Вторая градация будет включать в себя третий и четвертый уровни графа.

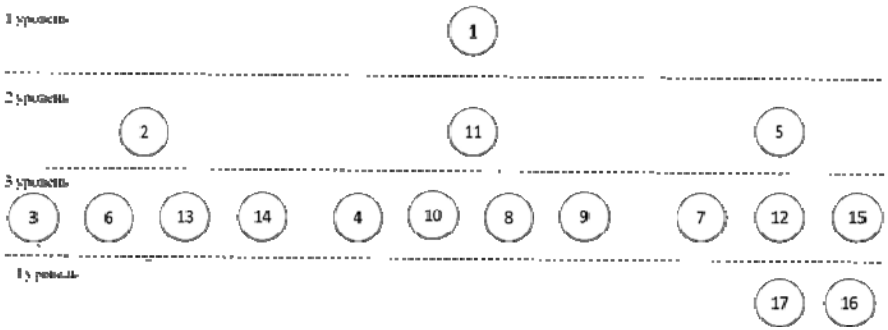


Рис. 1. Граф аудита ИБ

Fig. 1. Graph of IB audits

Менее критичные меры графа имеют следующую градацию:

– оценка 1 ставится, если мера реализована полностью;

– оценка 0 ставится, если мера не реализована или реализована частично.

Наиболее критичные меры графа имеют следующую градацию:

– оценка 1 ставится, если мера реализована полностью;

– оценка 0,5 ставится, если мера реализована не в полном объеме;

– оценка 0 ставится, если мера не реализована.

После получения результатов по достаточности реализации мер ЗИ необходимо получить конечные оценки, которые наглядно покажут состояние технической ЗИ. Для этого необходим эталон. Эталоном становится идеализированная система ЗИ, построенная в проверяемой организации. Согласно статье [16] применим следующий алгоритм вычисления оценки для третьего этапа аудита.

В первую очередь рассчитаем значения оценок для каждой из групп мер:

$$E_k = \frac{\sum_{i=1}^{n} M_i}{N}, \qquad (3)$$

где E_k – численное значение промежуточной оценки выбранных организацией групп мер; M_i – оценка *i*-й меры; N – число мер, необходимых в системе защиты; n – количество реализуемых мер.

Далее рассчитаем промежуточные оценки реализуемых групп мер. Для этого необходимо вычислить промежуточные значения по каждой ветви графа:

$$E_{bk} = \sum_{i=1}^{n} E_i \ , \tag{4}$$

где E_{bk} – числовое значение промежуточной оценки выбранной ветви мер; E_i – промежуточная оценка *i*-й группы меры, касающейся идентификации и аутентификации; *n* – число групп мер в ветви.

Для того чтобы получить взвешенную оценку эффективности защиты ИС, используем весовые коэффициенты. В системе защиты четыре ветви, поэтому, чтобы оценка была объективной и цельной, необходимо просчитать среднеарифметическое всех полученных оценок. Исходя из вышесказанного получаем следующую формулу:

$$E_{ch3} = \frac{0,5E_{B1} + 0,2E_{B2} + 0,2E_{B3} + 0,1E_{B4}}{4} \ . \tag{5}$$

где E_{ch3} – взвешенная оценка выбранных организацией мер; E_{B1} – промежуточная оценка групп мер первой ветви; E_{B2} – промежуточная оценка групп мер второй ветви; E_{B3} – промежуточная оценка групп мер третьей ветви; E_{B4} – промежуточная оценка групп мер четвертой ветви.

Последним пунктом аудита становится расчет оценки, которая содержит результаты всех этапов аудита, полученных по приведенным ранее формулам. Конечная оценка состояния системы ЗИ объекта КИИ должна учитывать результаты работы всех трех этапов. Если все три этапа аудита равноценны, будет применяться формула

$$E = \frac{E_{ch1} + E_{ch2} + E_{ch3}}{3} \ ,$$

где *E* – средневзвешенная оценка всех трех этапов аудита.

При получении итоговой оценки становится необходимым ее применение, т. е. сравнение с диапазоном допустимых значений в соответствии с уровнями объекта КИИ. Ни один из приказов не содержит в себе необходимых диапазонов, однако схожие диапазоны разработал Центральный банк Российской Федерации для обеспечения собственной безопасности. Такие

диапазоны прописаны в ГОСТ Р 57580.2–2018. Следует помнить, что финансовые организации также являются объектами КИИ, а значит, требования к системе защиты, разработанные ЦБ РФ и ФСТЭК, имеют схожую базу. Поэтому целесообразно использовать наиболее строгую систему оценивания, разработанную ЦБ РФ. Опираясь на вышеупомянутый ГОСТ, целесообразно использовать следующую градацию оценок:

– третий уровень ЗО КИИ соответствует диапазону $0{,}7 < E \leq 0{,}85$;
– второй уровень ЗО КИИ соответствует диапазону $0{,}85 < E \leq 0{,}9$;
– первый уровень ЗО КИИ соответствует диапазону $0{,}9 < E \leq 1$.

Здесь 1 – полное соответствие СЗИ эталонной СЗИ. Такие жесткие и высокие диапазоны оценок обоснованы тем, что нарушение функционирования ЗО КИИ в зависимости от его уровня может нанести непоправимый урон одной из пяти сфер согласно Постановлению Правительства [2].

ПРАКТИЧЕСКОЕ ПРИМЕНЕНИЕ

Для наглядного демонстрирования работы с разработанной методикой рассмотрим ее применение на практическом примере. Рассмотрим существующую систему ЗИ химическо-промышленной компании *N*. Система защиты информации имеет структуру, представленную на рис. 2.

Компания *N* провела категорирование ИС и определила, что ИС соответствует третьей категории. Согласно приказу ФСТЭК № 239 компания *N* обязана реализовать перечень мер, соответствующих третьей категории ЗО КИИ. Однако применение вслепую абсолютно всех мер нецелесообразно, так как если обратить внимание на структуру ИС, становится очевидно, что такие меры, как идентификация и аутентификация внешних пользователей, реализация защищенного удаленного доступа, управление перемещением виртуальных машин (контейнеров) и обрабатываемых на них данных, защита информации при использовании мобильных устройств, являются излишними.

Для первого этапа аудита необходима реализация нулевых мер каждой из групп мер, а также разработка документов, которая необходима согласно перечисленным ранее НПА. Для второго же этапа аудита важно, чтобы документ не только был разработан, но и содержал в себе всю необходимую информацию. Так, например, если мы обратимся к 16-й группе мер, то в разработанном документе, регламентирующем правила и процедуры обеспечения действий в нештатных ситуациях, должны содержаться комментарии и перечень действий по каждой из групп мер ДНС. Таким образом, результатом двух этапов будут две оценки, отображающие полноту наличия документации и оценку ее качества.

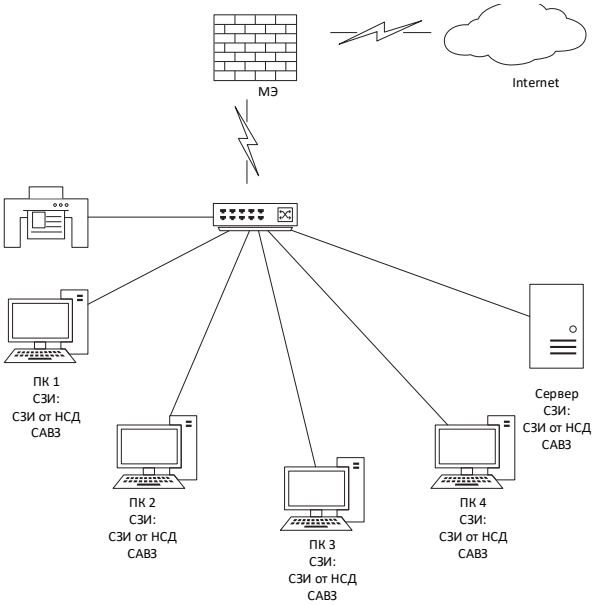


Рис. 2. Система защиты информации компании

Fig. 2. Company Information Security System

Конечными оценками СЗИ компании *N* стали:

- $E_{ch1} = 0,85$ – оценка достаточности разработанных документов;
- $E_{ch2} = 0,67$ – оценка полноты разработанных документов;
- $E_{ch3} = 0,71$ – оценка реализации мер приказа ФСТЭК № 239.

Получив оценки по каждому этапу, возможно рассчитать конечную оценку *E*:

$$E = \frac{E_{ch1} + E_{ch2} + E_{ch3}}{3} = 0,74 .$$

Из полученного значения оценки мы можем сделать вывод, что разработанная система соответствует нижней грани третьего уровня ЗО КИИ. При этом из приведенных результатов видно, что у компании есть проблемы не столько с наличием документов в области ЗИ, сколько с их актуальностью, что, в свою очередь, отражается как на реализации мер (оценка E_{ch3}), так и в целом на всей системе ЗИ.

ВЫВОД

Предложенная последовательность проведения аудита включает разбиение его на три этапа: аудит наличия документов, аудит качества документов, аудит реализации мер ЗИ в соответствии с законодательством. Такой подход покажет наиболее уязвимые места СЗИ в целом, а методика может точно показать места, которые необходимо закрыть организации, гибкая система оценивания покажет уровень защищенности всей СЗИ

СПИСОК ЛИТЕРАТУРЫ

1. Федеральный закон от 26 июля 2017 г. № 187-ФЗ «О безопасности критической информационной инфраструктуры Российской Федерации».

2. Постановление Правительства РФ от 8 февраля 2018 г. № 127 «Об утверждении Правил категорирования объектов критической информационной инфраструктуры Российской Федерации, а также перечня показателей критериев значимости объектов критической информационной инфраструктуры РФ и их значений».

3. Приказ ФСТЭК России от 25 декабря 2017 г. № 239 «Об утверждении Требований по обеспечению безопасности значимых объектов критической информационной инфраструктуры Российской Федерации».

4. ГОСТ Р ИСО 19011–2021. Оценка соответствия. Руководящие указания по проведению аудита систем менеджмента качества. – М.: Стандартинформ, 2021. – 35 с.

5. *Geldiyev H., Churiyev M., Mahmudov R.* Issues regarding cybersecurity in modern world // Digitalization and industry 4.0: Economic and societal development. – Wiesbaden: Springer Gabler, 2020. – P. 3–14. – DOI: 10.1007/978-3-658-27110-7_1.

6. *Ан В.Р., Табакаева В.А.* Разработка алгоритма проведения аудита кибербезопасности // МНСК-2021. Информационные технологии: материалы 59-й Международной научной студенческой конференции, Новосибирск, 12–23 апреля 2021 г. – Новосибирск, 2021. – С. 5. – EDN CAYHXE.

7. Разработка методики аудита кибербезопасности ГИС, относящихся к объектам критической информационной инфраструктуры Российской Федерации / В.Р. Ан, В.В. Селифанов, В.А. Табакаева, С.А. Буларга, А.С. Ворожцов // Сборник научных трудов НГТУ. – 2019. – № 3–4 (96). – С. 84–95. – DOI: 10.17212/2307-6879-2019-3-4-84-95. – EDN BOGOJY.

8. *Милославская Н.Г., Толстой А.И.* Проверка деятельности по управлению информационной безопасностью. – М.: Горячая линия – Телеком, 2022. – 172 с.

9. *Leksin V.N.* Effectiveness and efficiency of the activities of regional and municipal governments: Purpose and possibility of a correct assessment // Regional Research of Russia. – 2013. – Vol. 3. – P. 277–290. – DOI: 10.1134/S2079970513030076.

10. *Степашин С.В.* Государственный аудит в Российской Федерации // Государственный аудит. Право. Экономика. – 2009. – № 1. – С. 4–9. – EDN PCGDAD.

11. Постановление Правительства РФ от 17 февраля 2018 г. № 162 «Об утверждении Правил осуществления государственного контроля в области обеспечения безопасности значимых объектов критической информационной инфраструктуры Российской Федерации».

12. Приказ ФСТЭК России от 21 декабря 2017 г. № 235 «Об утверждении Требований к созданию систем безопасности значимых объектов критической информационной инфраструктуры Российской Федерации и обеспечению их функционирования».

13. Приказ ФСБ России от 19 июня 2019 г. № 282 «Об утверждении Порядка информирования ФСБ России о компьютерных инцидентах, реагирования на них, принятия мер по ликвидации последствий компьютерных атак, проведенных в отношении значимых объектов критической информационной инфраструктуры Российской Федерации».

14. Приказ ФСТЭК России от 29 апреля 2021 г. № 77 «Об утверждении порядка организации и проведения работ по аттестации объектов информатизации на соответствие требованиям о защите информации ограниченного доступа, не составляющей государственную тайну».

15. Приказ ФСБ России от 24 июля 2018 г. № 367 «Об утверждении Перечня информации, представляемой в государственную систему обнаружения, предупреждения и ликвидации последствий компьютерных атак на информационные ресурсы Российской Федерации и Порядка представления информации в государственную систему обнаружения, предупреждения и ликвидации последствий компьютерных атак на информационные ресурсы Российской Федерации».

16. *Sitskaya A., Selifanov V., Purgina M.* Development of weighting coefficients for assessing the quality of security measures implementation for significant objects of critical information infrastructure // Advances in Automation IV. RusAutoCon 2022. – Cham, Springer, 2023. – P. 411–420. – (Lecture Notes in Electrical Engineering; vol 986). – DOI: 10.1007/978-3-031-22311-2_39.

17. *Ситская А.В., Селифанов В.В.* Ранжирование мер обеспечения безопасности значимых объектов критической информационной инфраструктуры // Интерэкспо Гео-Сибирь. – 2022. – Т. 6. – С. 240–249. – DOI: 10.33764/2618-981X-2022-6-240-249. – EDN GFUGGU.

Ситская Анастасия Вадимовна, менеджер по продажам департамента информационной безопасности БЦ «Золотая долина». В настоящее время специализируется в области информационной безопасности. E-mail: Anastasiya.Sitskaya@softline.com

Селифанов Валентин Валерьевич, старший преподаватель кафедры защиты информации Новосибирского государственного технического университета. В настоящее время специализируется в области информационной безопасности. E-mail: sfo1@mail.ru

Звягинцева Полина Александровна, старший преподаватель кафедры информационной безопасности Сибирского государственного университета геосистем и технологи. Область научных интересов – защита информации. E-mail: polinasgugit@mail.ru

DOI: 10.17212/2782-2230-2023-3-67-82

Information security audit issues*

A.V. Sitskaya[1], V.V. Selifanov[2], P.A. Zvyagintseva[3]

[1] *Siberian State University of Geosystems and Technologies, 10 K. Plakhotnogo, Novosibirsk, 630108, Russian Federation, laboratory assistant of the Department of Information Security. E-mail: Anastasiya.Sitskaya@softline.com*
[2] *Novosibirsk State Technical University, 20 Karl Marx Prospekt, Novosibirsk, 630073, Russian Federation, senior lecturer of the Department of Information Security. E-mail: sfo1@mail.ru*
[3] *Siberian State University of Geosystems and Technologies, 10 K. Plakhotnogo, Novosibirsk, 630108, Russian Federation, associate professor of the Department of Information Security. E-mail: polinasgugit@mail.ru*

Today, information security reigns in absolutely all areas of human activity. Security breaches always have consequences, but depending on the structure, they can be invisible, or vice versa, stop the activities of an organization or even a country. The issue of ensuring information security of critical information infrastructure facilities has become the most discussed today. The state and its regulators to develop many regulatory legal documents that contain require-

* *Received 20 July 2023.*

ments for the system of protection of critical information infrastructure facilities of various levels of significance. However, no one has ever created a single technique that would show the real state of information security of a critical information infrastructure object. As a result, both the organization and regulators see only a general picture of the state of the defense system, which in turn creates many vulnerabilities that an attacker can use to carry out an attack. As a result of the attack on the object of critical information infrastructure, harm can be done not only to organizations, but also to people, the work processes of the entire state can be violated. That is why the issue of creating such a methodology becomes not only relevant, but also necessary both for the internal control of the organization itself and for automating the work of regulators during the next inspections. The methodology will show not only a high-quality assessment of the state of the protection system, but also show the possible vulnerabilities of the organization that need to be closed to improve the effectiveness of all protection.

Keywords: information security, critical information infrastructure, critical information infrastructure facility, audit, information security audit, information security system, development of evaluation methodology, protection of information systems

REFERENCES

1. Federal Law of July 26, 2017 No. 187-FZ "On the security of the critical information infrastructure of the Russian Federation". (In Russian).

2. Decree of the Government of the Russian Federation of February 8, 2018 No. 127 "On approval of the rules for categorizing critical information infrastructure facilities of the Russian Federation, as well as a list of indicators of criteria for the significance of critical information infrastructure facilities of the Russian Federation and their values". (In Russian).

3. Order of the FSTEC of Russia dated December 25, 2017 No. 239 "On approval of requirements for ensuring the security of significant objects of the critical information infrastructure of the Russian Federation". (In Russian).

4. State standard R ISO 19011–2021. *Conformity assessment. Guidelines for auditing management systems*. Moscow, Standartinform Publ., 2021. 35 p. (In Russian).

5. Geldiyev H., Churiyev M., Mahmudov R. Issues regarding cybersecurity in modern world. *Digitalization and industry 4.0: Economic and societal development*. Wiesbaden, Springer Gabler, 2020, pp. 3–14. DOI: 10.1007/978-3-658-27110-7_1.

6. An V.R., Tabakaeva V.A. [Development of an algorithm for conducting a cybersecurity audit]. *MNSK-2021. Informatsionnye tekhnologii* [MNSK-2021. Information technology]. Proceedings of the 59th International Students Scientific Conference, April 12−23, 2021. Novosibirsk, 2021, p. 5. (In Russian).

7. An V.R., Selifanov V.V., Tabakaeva V.A., Bularga S.A., Vorozhtsov A.S. Razrabotka metodiki audita kiberbezopasnosti GIS, otnosyashchikhsya k ob"ektam kriticheskoi informatsionnoi infrastruktury Rossiiskoi Federatsii [Development of a GIS cybersecurity audit methodology related to critical information infrastructure

facilities of the Russian Federation]. *Sbornik nauchnykh trudov Novosibirskogo gosudarstvennogo tekhnicheskogo universiteta = Transaction of scientific papers of the Novosibirsk state technical university*, 2019, no. 3–4 (96), pp. 84–95. DOI: 10.17212/2307-6879-2019-3-4-84-95.

8. Miloslavskaya N.G., Tolstoi A.I. *Proverka deyatel'nosti po upravleniyu informatsionnoi bezopasnost'yu* [Verification of information security management activities]. Moscow, Goryachaya liniya – Telekom Publ., 2022. 172 p.

9. Leksin V.N. Effectiveness and efficiency of the activities of regional and municipal governments: Purpose and possibility of a correct assessment. *Regional Research of Russia*, 2013, vol. 3, pp. 277–290. DOI: 10.1134/S2079970513030076.

10. Stepashin S.V. Gosudarstvennyi audit v Rossiiskoi Federatsii [State audit in the Russian Federation]. *Gosudarstvennyi audit. Pravo. Ekonomika = State audit. Law. Economics*, 2009, no. 1, pp. 4–9.

11. Decree of the Government of the Russian Federation of February 17, 2018 No. 162 "On approval of the Rules for the implementation of state control in the field of ensuring the security of significant objects of the critical information infrastructure of the Russian Federation". (In Russian).

12. Order of the FSTEC of Russia dated December 21, 2017 No. 235 "On approval of Requirements for the creation of security systems for significant objects of the critical information infrastructure of the Russian Federation and ensuring their functioning". (In Russian).

13. Order of the FSB of Russia dated June 19, 2019 No. 282 "On approval of the Procedure for informing the FSB of Russia about computer incidents, responding to them, taking measures to eliminate the consequences of computer attacks carried out on significant objects of the critical information infrastructure of the Russian Federation". (In Russian).

14. Order of the FSTEC of Russia dated April 29, 2021 No. 77 "On approval of the Procedure for organizing and conducting work on the certification of informatization facilities for compliance with the requirements for the protection of limited access information that does not constitute a state secret". (In Russian).

15. Order of the FSB of Russia dated July 24, 2018 No. 367 "On approval of the List of information submitted to the state system for detecting, preventing and eliminating the consequences of computer attacks on information resources of the Russian Federation and the Procedure for submitting information to the state system for detecting, preventing and eliminating the consequences of computer attacks on information resources of the Russian Federation". (In Russian).

16. Sitskaya A., Selifanov V., Purgina M. Development of weighting coefficients for assessing the quality of security measures implementation for significant objects of critical information infrastructure. *Advances in Automation IV. RusAuto-*

Con 2022. Cham, Springer, 2023, pp. 411–420. DOI: 10.1007/978-3-031-22311-2_39.

17. Sitskaya A.V., Selifanov V.V. Ranzhirovanie mer obespecheniya bezopasnosti znachimykh ob"ektov kriticheskoi informatsionnoi infrastruktury [Ranking of security measures of significant objects of critical information infrastructure]. *Interekspo Geo-Sibir' = Interexpo GEO-Siberia*, 2022, vol. 6, pp. 240–249. DOI: 10.33764/2618-981X-2022-6-240-249.

Для цитирования:

Ситская А.В., Селифанов В.В., Звягинцева П.А. Вопросы аудита информационной безопасности // Безопасность цифровых технологий. – 2023. – № 3 (110). – С. 67–82. – DOI: 10.17212/2782-2230-2023-3-67-82.

For citation:

Sitskaya A.V., Selifanov V.V., Zvyagintseva P.A. Voprosy audita informatsionnoi bezopasnosti [Information security audit issues]. *Bezopasnost' tsifrovykh tekhnologii = Digital Technology Security*, 2023, no. 3 (110), pp. 67–82. DOI: 10.17212/2782-2230-2023-3-67-82.

ПРАВИЛА ДЛЯ АВТОРОВ

УСЛОВИЯ ПРИЕМА СТАТЕЙ

Все статьи и сопровождающие их материалы в журнал подаются через сайт журнала в электронном виде после регистрации всех авторов статьи. Регистрация обязывает каждого автора иметь международный идентификационный номер ORCID. Иные варианты подачи материалов не рассматриваются.

Автор (один из соавторов) в своем личном кабинете выбирает в меню пункт «Подать статью» и вводит все необходимые данные. Своих соавторов при этом он выбирает из списка зарегистрированных пользователей.

Рукопись статьи готовится в соответствии с правилами оформления в редакторе MS Word и прикрепляется в формате *.doc, *.docx.

Сканированные лицензионный договор с подписями авторов и экспертное заключение (цветной режим сканирования, разрешение не менее 600 dpi) необходимо также разместить на сайте журнала в разделе «Подать статью» в формате *.pdf, *.jpg, *.jpeg.

По окончании всех работ обязательно нажать кнопку «Отправить в редакцию».

В редакцию журнала предоставляются следующие материалы.

1. **Статья**, подготовленная в соответствии с правилами оформления, – печатная версия, 2 экземпляра, подписанных авторами.

2. **Контактная информация** (телефоны рабочий и сотовый, адреса электронной почты, место работы, адрес места работы, должность, ученая степень, ученое звание автора) – печатная версия, 2 экземпляра.

3. **Описание статьи для базы данных «Российский индекс научного цитирования (РИНЦ)»**, подготовленное в соответствии с правилами оформления, – печатная версия, один экземпляр.

4. **Лицензионный договор**, заполненный и подписанный.

5. **Электронная версия статьи, контактной информации, описания статьи для базы данных РИНЦ, сканированный лицензионный договор и экспертное заключение о возможности опубликования (в отдельных файлах на адрес редакции)**.

6. **Согласие на публикацию, обработку и распространение персональных данных авторов статей**.

7. **Экспертное заключение о возможности опубликования**.

Редакцией рассматриваются только те материалы авторов, которые полностью соответствуют вышеобозначенным требованиям. Неполный пакет материалов редакцией не рассматривается.

Подготовленные материалы направляются на почтовый адрес редакции: 630073, г. Новосибирск, пр. Карла Маркса, 20, Новосибирский государственный технический университет (НГТУ), корп. 7, ком. 606, в редакцию журнала «Безопасность цифровых технологий».

Все рукописи рецензируются, по результатам рецензирования редколлегия принимает решение о целесообразности опубликования материалов.

ВНИМАНИЕ!

Авторы несут ответственность за оформление, содержание и сам факт публикации статьи. Редакция журнала не несет ответственности за возможный ущерб, вызванный публикацией статьи. При наличии существенных недостатков в оформлении и содержании статьи редакция принимает решение об отклонении статьи без приведения полного перечня ошибок автора.

Ранее опубликованные материалы, а также материалы, представленные для публикации в других журналах, к рассмотрению не принимаются.

ПРАВИЛА ОФОРМЛЕНИЯ

При подготовке документов для отправки в редакцию журнала авторам рекомендуется внимательно прочитать правила и посмотреть примеры оформления статей и всех необходимых сопутствующих документов. Редакция рассматривает статьи, подготовленные как на русском, так и на английском языке. Для опубликования статьи на английском языке необходимо дополнительно предоставить ее русскоязычный вариант, оформленный по правилам журнала (кроме зарубежных авторов).

Перед отправкой рукописи в редакцию авторам необходимо проверить свою статью с помощью системы «Антиплагиат». Принятый редакционной коллегией уровень оригинальности статей должен составлять не менее 85 %.

Чтобы статья была направлена на рецензирование, необходимо подготовить следующее:

1) **статью** в соответствии с правилами оформления;

2) **контактную информацию** в одном файле предоставить по каждому автору: ФИО полностью, ученая степень, ученое звание автора, должность, место работы, адрес места работы, телефон рабочий и мобильный, адрес электронной почты;

3) **описание статьи для базы данных «Российский индекс научного цитирования (РИНЦ)»**;

4) **лицензионный договор** заполнить, бланк лицензионного договора должен быть подписан только авторами (он доступен авторам также в личном кабинете). Если авторов несколько, то необходимо добавить поля на всех авторов и подписать каждому из них;

5) **экспертное заключение** о возможности опубликования, принятое в вашей организации;

6) согласие на публикацию, обработку и распространение персональных данных авторов статей;

7) авторы, не являющиеся сотрудниками НГТУ, предоставляют **сопроводительное письмо** на имя проректора по научной работе НГТУ (ссылка на страницу сайта НГТУ). Письмо нужно подготовить на бланке организации с подписью и печатью руководителя.

ОСНОВНЫЕ РАЗДЕЛЫ ЖУРНАЛА

Автоматизация и управление технологическими процессами и производствами.

Управление в социальных и экономических системах.

Методы и системы защиты информации, информационная безопасность.

RULES FOR AUTHORS

CONDITIONS FOR ACCEPTANCE OF ARTICLES

All articles and their accompanying materials are submitted to the magazine through the magazine's website in electronic form after registration of all the authors of the article. Registration obliges each author to have an international ORCID. No other material supply options are considered.

The author (one of the co-authors) in his personal account selects the item "Submit article" in the menu and enters all the necessary data. At the same time, he selects his co-authors from the list of registered users.

The manuscript of the article is prepared in accordance with the design rules in the MS Word editor and attached in the format * .doc, * .docx.

Scanned license agreement with signatures of authors and expert opinion (color mode scanning, resolution not less than 600 dpi) can also be attached on the website of the magazine in the section "Submit article" in the format * .pdf, * .jpg, * .jpeg.

At the end of all works, be sure to click the "Send to Design" button.

The following materials are provided to the journal editor:

1. **The article**, prepared in accordance with the rules of design, is a private version, 2 copies signed by the authors.
2. **Contact information** (working and cellular phones, e-mail addresses, place of work, address of the place of work, position, scientific degree, academic title of the author) – printed version, 2 copies.
3. **The description of the article** for the database "**Russian Scientific Citation Index (RSCI)**", prepared in accordance with the rules of form-making, is a printed version, one copy.
4. **License agreement** completed and signed.
5. **Electronic version of the article**, contact information, description of the article for the RSCI database, scanned license agreement and expert opinion on the possibility of publication (in separate files to the editorial address).
6. **Consent to the publication, processing and dissemination of the personal data** of the authors of the articles.
7. **Expert opinion** on the possibility of publication.

The editors consider only those materials of the authors that fully meet the above requirements. Incomplete package of materials is not considered by the revision.

The prepared materials are sent to the postal address of the editorial office: 630073, Novosibirsk, Karl Marx Prospekt, 20, Novosibirsk State Technical University (NSTU), building 7, office 606, to the editors of the journal "Digital Technology Security".

All manuscripts were reviewed, and according to the results of the review, the editorial board decided on the appropriateness of publishing the materials.

ATTENTION!

The authors are responsible for the design, content and the fact of publication of the article. The editorial board of the journal is not responsible for possible damage caused by the publication of the article. If there are significant shortcomings in the design and content of the article, the editorial board decides to reject the article without giving a full list of the author's mistakes.

Previously published materials, as well as materials submitted for publication in other journals, are not accepted for consideration.

FORMATTING RULES

When preparing documents for submission to the journal editor, authors are advised to carefully read the rules and see examples of the design of articles and all necessary related documents. The Drafting Committee considered articles prepared in both Russian and English. To publish the article in English, it is necessary to additionally provide its Russian-language version, drawn up according to the rules of the magazine (except for foreign authors).

Before sending the manuscript to the editorial office, authors must check their article using the Antiplagiarism system. The level of originality of articles adopted by the Editorial Board should be at least 85 %.

For the article to be aimed at peer review, you need to prepare the following:

1) **the article** in accordance with the rules of design (volume from 7 to 30 pages);

2) **provide contact information** in one file for each author: full name, degree, academic title of the author, position, place of work, address of the place of work, telephone number of the worker and mobile, e-mail address;

3) **description of the article** for the database "Russian Scientific Citation Index (RSCI)";

4) fill out the **license agreement**, the form of the license agreement must be signed only by the authors (it is also available to the authors in the personal office), if there are several authors, then it is necessary to add fields on all authors and sign each of them;

5) **expert opinion** on the possibility of publication, adopted in your organization;

6) consent to the publication, processing and dissemination of the personal data of the authors of the articles;

7) authors who are not employees of the NSTU provide a **companion letter** addressed to the vice-rector for scientific work of the NSTU (link to the page of the NSTU website). The letter should be prepared on the form of the organization with the signature and seal of the manager.

JOURNAL SECTION

Automation and control of technological processes and productions.
Governance in social and economic systems.
Methods and systems of information protection, information security.