УДК 004.056.53 Вестник УрФО № 2(48) / 2023, с. 67–73

Конев А.А., Коваленко А.С., Репкин В.С., Семёнов Г.Ю.

DOI: 10.14529/secur230207

УЯЗВИМОСТЬ «GITEA GIT FETCH REMOTE CODE EXECUTION»: АНАЛИЗ, ФОРМАЛИЗАЦИЯ АВТОМАТИЗИРОВАННОЙ ЭКСПЛУАТАЦИИ, МЕРЫ ЗАЩИТЫ

В данной научной статье проводится анализ уязвимости «Gitea Git Fetch Remote Code Execution (CVE-2022-30781)», которая была выявлена в программном обеспечении Gitea, предоставляющем возможность управления версиями разрабатываемого проекта. Описывается методика эксплуатации уязвимости и реализация автоматизированной атаки злоумышленником. В рамках исследования рассматриваются меры защиты и рекомендации по обеспечению безопасности систем, работающих с Gitea. Представленный материал является основой для создания уязвимых узлов в киберполигонах, а также поможет разработчикам, системным администраторам, специалистам по информационной безопасности и обычным пользователям лучше понять уязвимость «Gitea Git Fetch RCE», и, как следствие, принять соответствующие меры для предотвращения атак и обеспечения безопасности собственных систем.

Ключевые слова: *информационная безопасность, киберполигон, Gitea, уязвимость, эксплойт, Meta Attack Language, автоматизированная атака, Metasploit.*

Konev A.A., Kovalenko A.S., Repkin V.S., Semenov G.Yu.

VULNERABILITY «GITEA GIT FETCH REMOTE CODE EXECUTION»: ANALYSIS, AUTOMATED EXPLOITATION FORMALIZATION, AND MITIGATION MEASURES

This scientific article analyzes the vulnerability "Gitea Git Fetch Re-mote Code Execution (CVE-2022-30781)", which was identified in the Gitea software, which provides the ability to manage versions of the project under development. The technique of exploiting the vulnerabil-

ity and the implementation of an automated attack by an attacker is described. The study examines security measures and recommendations for ensuring the security of systems running with Gitea. The presented material is the basis for creating secure nodes in cyberpolygons, and will also help developers, system administrators, information security specialists and ordinary users to better understand the vulnerability of "Gitea Git Fetch RCE", and, as a result, take appropriate measures to prevent attacks and ensure the security of their own systems.

Keywords: *information security, cyberpolygon, Gitea, vulnerability, exploit, Meta Attack Language, automated attack, Metasploit.*

Введение

Сегодня важным вопросом является обучение специалистов по информационной безопасности, такие специалисты должны быть хорошо ознакомлены с существующими уязвимостями и угрозами, методами и технологиями, используемыми киберпреступниками. Для эффективной работы они должны непрерывно совершенствовать свои навыки и быть в курсе последних тенденций в области информационной безопасности.

Для получения специалистами по кибербезопасности практических навыков был разработан киберполигон Ampire, в основе которого находятся статические и динамические, состоящие из уязвимых узлов, сценарии, которые являются имитацией реальной компьютерной атаки. Однако для того, чтобы обучение с использованием данной платформы оставалось актуальным, необходимо регулярно обновлять базу сценариев кибератак и уязвимых узлов [1–3].

Gitea является популярной системой управления версиями Git, широко применяемой в различных проектах разработки программного обеспечения. Gitea является проектом с открытым исходным кодом, официальный репозиторий на платформе GitHub имеет 37 тысяч «звезд» и около полумиллиона наблюдателей [4].

Так как уязвимости типа Remote Code Execution считаются наиболее опасными и разрушительными уязвимостями в сфере информационной безопасности, из-за того, что успешная атака позволяет злоумышленнику выполнять удаленный код на целевой системе или устройстве без необходимости иметь прямой физический доступ к ним, а следовательно, предоставляет злоумышленнику полный контроль над целевой системой, уязвимость «Gitea Git Fetch RCE» может иметь серьезные последствия для множества организаций и проектов, использующих программное обеспечение Gitea. А значит, данная уязвимость является хорошим примером для обучения специалистов по информационной безопасности. Поэтому было принято решение подробно разобрать уязвимость и механизм ее работы для реализации уязвимого узла на основе «Gitea Git Fetch RCE».

Анализ уязвимости

Рассматриваемая уязвимость была обнаружена 16 апреля 2022 года и затрагивает все версии Gitea ниже 1.16.7, основанием для поиска данной уязвимости стала уязвимость CVE-2022-0415 системы Gogs, на которой было основано программное обеспечение Gitea.

Уязвимость «Gitea Git Fetch RCE» связана с ошибками в механизме миграции. Функция миграции (Migration) в Gitea состоит из двух частей: Downloader и Uploader, первая часть отвечает за загрузку информации о репозитории с удаленного сервиса, а вторая – за запись этой информации в Gitea.

В настоящее время Downloader поддерживает импорт кода из таких сервисов, как GitHub, GitLab, GitBucket, Gogs, Gitea и других. А реализация Uploader ограничена только одним сервисом – Gitea, поскольку окончательная миграция удаленного репозитория производится только на локальный экземпляр Gitea.

Возможность выполнения кода возникает в момент создания запросов на слияние локального репозитория из-за неправильной обработки входящих параметров [5].

Таким образом, злоумышленнику необходимо подделать экземпляр Gitea, настроив HTTP-сервер и необходимые маршруты для имитации удаленного сервиса, а именно:

/api/v1/version;

/api/v1/settings/api;

/api/v1/repos/<owner>/<repo>/;

/api/v1/repos/<owner>/<repo>/topics;

/api/v1/repos/<owner>/<repo>/pulls;

/api/v1/repos/<owner>/<repo>/issues/1/reactions;

/api/v1/repos/<owner>/<repo>/pulls/2/reviews.

В JSON-ответе маршрута «/api/v1/repos/<owner>/<repo>/pulls/2/reviews» необходимо изменить соответствующие поля, указав вместо <cmd> команду, которую нужно выполнить:

[0].head.ref: --upload-pack=bash -c '<cmd>';

[0].head.repo.clone_url: ./;

[0].head.owner.login: <username>.

Формальное описание эксплуатации уязвимости с помощью методологии моделирования Meta Attack Language показано на рис. 1 [6].

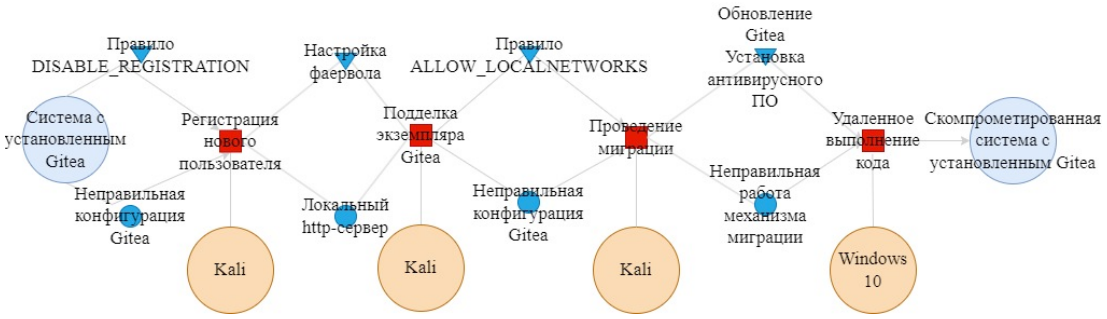

Рис. 1. Формальное описание эксплуатации уязвимости с помощью методологии моделирования Meta Attack Language

Первый эксплойт к описанной уязвимости был опубликован на платформе GitHub пользователем wuhan005, который и обнаружил уязвимость [7].

Автоматизированная эксплуатация уязвимости с помощью Metasploit

Для проведения атаки, эксплуатирующей уязвимость «Gitea Git Fetch RCE», будет использоваться готовый модуль консоли Metasploit Framework «exploit/multi/http/gitea_git_fetch_rce», в котором используется эксплойт, метод работы которого был ранее описан [8–10].

Для работы модуля необходимо заполнить следующие опции:

– PASSWORD – пароль пользователя, от чьего имени будет производиться миграция;

– RHOSTS – ip-адрес целевой системы;

– USERNAME – имя пользователя, от чьего имени будет производиться миграция;

– TARGET – цель эксплойта.

Также для модуля необходимо выбрать payload, то есть полезную нагрузку, которая будет выполнена на целевой машине после эксплуатации уязвимости. Для выбранного payload «windows/x64/shell_reverse_tcp» необходимо установить LHOST – ip-адрес, к которому целевая система отправит запрос на подключение после выполнения полезной нагрузки. Остальные опции имеют установленные значения по умолчанию и нужны для более гибкой настройки [11].

Консоль Metasploit Framework с заполненными опциями для выбранного модуля показана на рис. 2.

Для автоматизации действий атакующего будет использоваться скрипт на языке программирования Python, с подключенной библиотекой PyMetasploit от allfro, таким образом взаимодействие с Metasploit Framework будет производиться уже не через консоль, а через службу msfrpcd [12].

Для начала необходимо импортировать требующиеся библиотеки и подключиться к службе msfrpcd, пароль задается при инициализации, порт по умолчанию 55553.

```
from time import sleep
from typing import Union
from pymetasploit3.msfrpc import MsfRpcClient, ShellSession, MsfConsole
import requests
client = MsfRpcClient("PasswordToConnect", port=55553, ssl=True)
```

Далее в коде задаются глобальные переменные, соответствующие некоторым опциям, для более удобной работы.

```
RHOSTS = '192.168.0.1'
LHOST = '192.168.0.2'
USERNAME = 'Hacker'
PASSWORD = 'YourP@ssw0rd'
```

Затем в теле главной функции программы с помощью библиотеки requests производится регистрация пользователя злоумышленника, если код ответа равен 200, то выводится сообщение об успешной регистрации, в противном случае будет выведено сообщение о неудачной операции.

```
def main():
    url = 'http://'+ RHOSTS +':3000/user/sign_up'
```

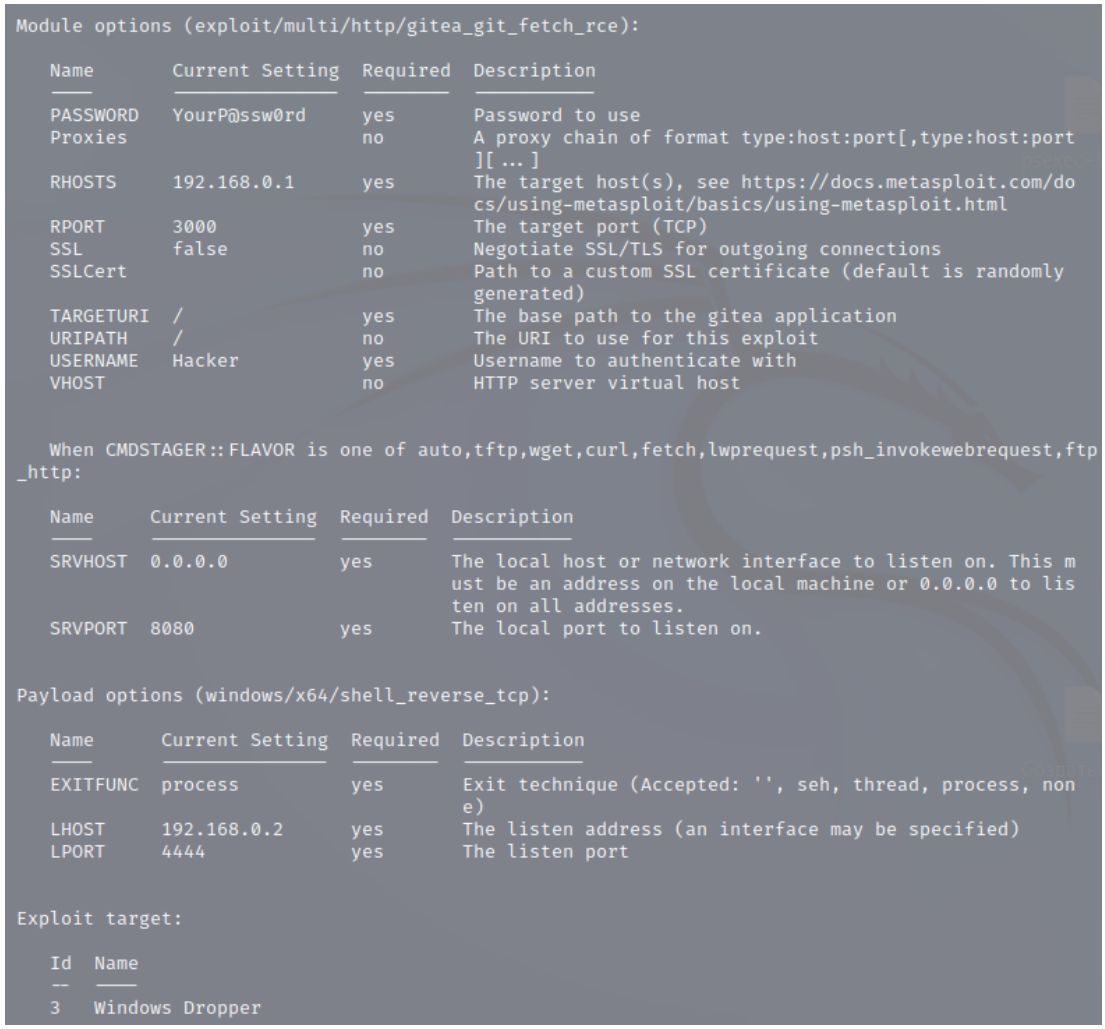

Рис. 2. Консоль Metasploit Framework с заполненными опциями для модуля «exploit/multi/http/gitea_git_fetch_rce»

```
    data = {
      'user_name': USERNAME,
      'email': 'hacker@tusur.com',
      'password': PASSWORD,
      'retype': PASSWORD}
    response = requests.post(url, data=data)
    if response.status_code == 200:
        print("Регистрация пользователя "+USERNAME+" прошла успешно")
    else:
        print(«Не удалось зарегистрировать пользователя «+USERNAME)
```

Далее так же в теле главной функции производится выбор используемого модуля и заполнение его опций.

```
    print(«Эксплуатация уязвимости Gitea git fetch RCE»)
    exploit = client.modules.use('exploit', 'multi/http/gitea_git_fetch_rce')
    exploit['RHOSTS'] = RHOSTS
    exploit['PASSWORD'] = PASSWORD
    exploit['USERNAME'] = USERNAME
    exploit.target = 3    # Windows Dropper
    payload = client.modules.use('payload',
        'windows/x64/shell_reverse_tcp')
    payload['LHOST'] = LHOST
```

После выбора модуля и заполнения опций производится эксплуатация уязвимости, во избежание каких-либо случайных неполадок, выполняется 5 попыток запуска модуля.

```
    for i in range(5):
        if 'created in the background.' in run_with_output(exploit, payload):
            print("Уязвимость Gitea git fetch RCE была успешно"
                "прэксплуатирована")
            return
    print("Не удалось проэксплуатировать уязвимость Gitea git fetch RCE"
        «спустя 5 попыток»)
```

В результате выполнения данного скрипта на языке программирования Python, злоу-

мышленник получит shell-сессию с обратным подключением на целевой системе, а следовательно полный контроль над атакованной системой. Далее действия злоумышленника будут ограничены только его фантазией и функциональными возможностями системы.

Меры защиты

Для предотвращения эксплуатации «Gitea Git Fetch RCE» на системе с установленным программным обеспечением от Gitea необходимо обновить Gitea до версии 1.16.7 и выше, также необходимо обновить Git до версии 2.30.2 и выше. В указанных версиях уязвимость была устранена.

При невозможности обновления программного обеспечения достаточно изменить конфигурацию работы Gitea, указав в конфигурационном файле следующие правила:

DISABLE_REGISTRATION = true (по умолчанию стоит false);

ALLOW_LOCALNETWORKS = false (по умолчанию не задано).

Первое правило отвечает за возможность регистрации новых пользователей, при значении true, пользователи могут беспрепятственно регистрировать новые аккаунты, при значении false, новые аккаунты могут быть зарегистрированы только администратором Gitea.

Второе правило отвечает за возможность миграции с частных ip-адресов, однако, стоит отметить, что правило будет игнорироваться при заданном правиле ALLOWED_DOMAINS.

Также, если настроено правило ALLOWED_DOMAINS, в нем необходимо указать только доверенные домены, с которых может быть произведена миграция git-репозиториев [13].

Заключение

В результате проведенного анализа была изучена уязвимость «Gitea Git Fetch Remote Code Execution (CVE-2022-30781)» и ее механизм работы, в рамках формализации автоматизированной атаки злоумышленника была построена модель эксплуатации уязвимости с помощью методологии моделирования Meta Attack Language.

Для интеграции уязвимого узла на базе уязвимости «Gitea Git Fetch RCE» в киберполигон был написан скрипт на языке программирования Python, позволяющий автоматизировать взаимодействие злоумышленника с модулем Metasploit Framework, содержащим эксплойт к уязвимости. Также написанная программа может быть использована для тестирования информационных систем на проникновение, так как позволяет сымитировать действия злоумышленника, направленные, на уязвимость в ПО Gitea.

В результате исследования были предложены необходимые и достаточные меры, позволяющие обеспечить защиту информационных систем, использующих программное обеспечение Gitea.

Литература

1. Моделирование сценариев кибератак на базе киберполигогна Ampire / Н. С. Егошин, Н. И. Сермавкин, В. С. Репкин, А. Д. Калякин // Электронные средства и системы управления. Материалы докладов Международной научно-практической конференции. – 2022. – № 1–2. – С. 69–72.

2. Разработка сценария кибератаки на веб-портал предприятия / А. Д. Калякин, А. С. Коваленко, В. С. Репкин, А. К. Новохрестов // Электронные средства и системы управления. Материалы докладов Международной научно-практической конференции. – 2022. – № 1-2. – С. 65–68.

3. Интеграция киберполигона Ampire в учебный процесс / Г. Ю. Семенов, Т. П. Лазарев, Н. И. Сермавкин, А. К. Новохрестов // Современное образование: интеграция образования, науки, бизнеса и власти. Трансформация образования, науки и производства – основа технологического прорыва: материалы международной научно-методической конференции. В 2 ч., Томск, 26–27 января 2023 года. Том Часть 1. – Томск: Томский государственный университет систем управления и радиоэлектроники, 2023. – С. 152–156.

4. GitHub: go-gitea / Gitea: Git with a cup of tea. [Электронный ресурс]. – URL: https://github.com/go-gitea/gitea (дата обращения 01.06.2023).

5. Tttang: CVE-2022-30781: Gitea RCE caused by a common Git command. [Электронный ресурс] – URL: https://tttang.com/archive/1607/ (дата обращения 02.06.2023).

6. Методы формализации описания сценариев кибератак / А. Ю. Якимук, С. А. Устинов, Т. П. Лазарев, А. С. Коваленко // Электронные средства и системы управления. Материалы докладов Международной научно-практической конференции. – 2022. – № 1–2. – С. 73–76.

7. GitHub: wuhan005 / CVE-2022-30781: Gitea repository migration remote command execution

exploit. [Электронный ресурс] – URL: https://github.com/wuhan005/CVE-2022-30781 (дата обращения 02.06.2023).

8. Kennedy, D., O'gorman, J., Kearns, D., & Aharoni, M. (2011). Metasploit: the penetration tester's guide. No Starch Press.

9. Holik F. et al. Effective penetration testing with Metasploit framework and methodologies //2014 IEEE 15th International Symposium on Computational Intelligence and Informatics (CINTI). – IEEE, 2014. – С. 237–242.

10. Maynor D. Metasploit toolkit for penetration testing, exploit development, and vulnerability research. – Elsevier, 2011.

11. InfosecMatter: Gitea Git Fetch RCE. [Электронный ресурс] – URL: https://www.infosecmatter.com/metasploit-module-library/?mm=exploit/multi/http/gitea_git_fetch_rce (дата обращения 02.06.2023).

12. GitHub: allfro / pymetasploit: A full-fledged msfrpc library for Metasploit framework. [Электронный ресурс] – URL: https://github.com/allfro/pymetasploit (дата обращения 03.06.2023).

13. Gitea Documentation: Configuration Cheat Sheet. [Электронный ресурс] – URL: https://docs.gitea.com/next/administration/config-cheat-sheet (дата обращения 04.06.2023).

Refernces

1. Modelirovanie stsenariev kiberatak na baze kiberpoligogna Ampire / N. S. Egoshin, N. I. Sermavkin, V. S. Repkin, A. D. Kalyakin // Elektronnye sredstva i sistemy upravleniya. Materialy dokladov Mezhdunarodnoy nauchno-prakticheskoy konferentsii. – 2022. – № 1–2. – S. 69–72.

2. Razrabotka stsenariya kiberataki na veb-portal predpriyatiya / A. D. Ka-lyakin, A. S. Kovalenko, V. S. Repkin, A. K. Novokhrestov // Elektronnye sredstva i sistemy upravleniya. Materialy dokladov Mezhdunarodnoy nauchno-prakticheskoy konferentsii. – 2022. – № 1–2. – S. 65–68.

3. Integratsiya kiberpoligona Ampire v uchebnyy protsess / G. Yu. Semenov, T. P. Lazarev, N. I. Sermavkin, A. K. Novokhrestov // Sovremennoe obrazovanie: integratsiya obrazovaniya, nauki, biznesa i vlasti. Transformatsiya obrazovaniya, nauki i proizvodstva – osnova tekhnologicheskogo proryva: materialy mezhdunarodnoy nauchno-metodicheskoy konferentsii. V 2 ch., Tomsk, 26–27 yanvarya 2023 goda. Tom Chast' 1. – Tomsk: Tomskiy gosudarstvennyy universitet sistem upravleniya i radioelektroniki, 2023. – S. 152–156.

4. GitHub: go-gitea / Gitea: Git with a cup of tea. [Elektronnyy resurs]. – URL: https://github.com/go-gitea/gitea (data obrashcheniya 01.06.2023).

5. Tttang: CVE-2022-30781: Gitea RCE caused by a common Git command. [Elektronnyy resurs] – URL: https://tttang.com/archive/1607/ (data obrashcheniya 02.06.2023).

6. Metody formalizatsii opisaniya stsenariev kiberatak / A. Yu. Yakimuk, S. A. Ustinov, T. P. Lazarev, A. S. Kovalenko // Elektronnye sredstva i sistemy upravleniya. Materialy dokladov Mezhdunarodnoy nauchno-prakticheskoy konferentsii. – 2022. – № 1–2. – S. 73–76.

7. GitHub: wuhan005 / CVE-2022-30781: Gitea repository migration remote command execution exploit. [Elektronnyy resurs] – URL: https://github.com/wuhan005/CVE-2022-30781 (data obrashcheniya 02.06.2023).

8. Kennedy, D., O'gorman, J., Kearns, D., & Aharoni, M. (2011). Metasploit: the penetration tester's guide. No Starch Press.

9. Holik F. et al. Effective penetration testing with Metasploit framework and methodologies //2014 IEEE 15th International Symposium on Computational Intelligence and Informatics (CINTI). – IEEE, 2014. – S. 237–242.

10. Maynor D. Metasploit toolkit for penetration testing, exploit development, and vulnerability research. – Elsevier, 2011.

11. InfosecMatter: Gitea Git Fetch RCE. [Elektronnyy resurs] – URL: https://www.infosecmatter.com/metasploit-module-library/?mm=exploit/multi/http/gitea_git_fetch_rce (data obrashcheniya 02.06.2023).

12. GitHub: allfro / pymetasploit: A full-fledged msfrpc library for Metasploit framework. [Elektronnyy resurs] – URL: https://github.com/allfro/pymetasploit (data obrashcheniya 03.06.2023).

13. Gitea Documentation: Configuration Cheat Sheet. [Elektronnyy resurs] – URL: https://docs.gitea.com/next/administration/config-cheat-sheet (data obrashcheniya 04.06.2023).

КОНЕВ Антон Александрович, кандидат технических наук, доцент кафедры комплексной информационной безопасности электронно-вычислительных систем федерального государственного бюджетного образовательного учреждения высшего образования «Томский государственный университет систем управления и радиоэлектроники». Россия, 634050, Томская область, г. Томск, пр. Ленина, 40. E-mail: kaa@fb.tusur.ru

КОВАЛЕНКО Александр Сергеевич, студент федерального государственного бюджетного образовательного учреждения высшего образования «Томский государственный университет систем управления и радиоэлектроники». Россия, 634050, Томская область, г. Томск, пр. Ленина, 40. E-mail: a.s.kovalenko@mail.ru

РЕПКИН Владимир Сергеевич, студент федерального государственного бюджетного образовательного учреждения высшего образования «Томский государственный университет систем управления и радиоэлектроники». Россия, 634050, Томская область, г. Томск, пр. Ленина, 40. E-mail: repkin_vova@mail.ru

СЕМЕНОВ Григорий Юрьевич, студент федерального государственного бюджетного образовательного учреждения высшего образования «Томский государственный университет систем управления и радиоэлектроники». Россия, 634050, Томская область, г. Томск, пр. Ленина, 40. E-mail: semenov.g.749-1@e.tusur.ru

KONEV Anton Alexandrovich, candidate of technical sciences, professor of the department of integrated information security of electronic computing systems of the federal state budgetary educational institution of higher education "Tomsk State University of Control Systems and Radioelectronics". Russia, 634050, Tomsk region, Tomsk, Lenin Avenue, 40. E-mail: kaa@fb.tusur.ru

KOVALENKO Alexandr Sergeevich, student of the federal state educational institution of higher education «Tomsk State University of Control Systems and Radioelectronics». Russia, 634050, Tomsk region, Tomsk, Lenin Avenue, 40. E-mail: a.s.kovalenko@mail.ru

REPKIN Vladimir Sergeevich, student of the federal state educational institution of higher education «Tomsk State University of Control Systems and Radioelectroics». Russia, 634050, Tomsk region, Tomsk, Lenin Avenue, 40. E-mail: repkin_vova@mail.ru

SEMENOV Grigoriy Yurievich, student of the federal state educational institution of higher education «Tomsk State University of Control Systems and Radioelectronics». Russia, 634050, Tomsk region, Tomsk, Lenin Avenue, 40. E-mail: semenov.g.749-1@e.tusur.ru