ҚАЗАҚСТАН РЕСПУБЛИКАСЫ ҒЫЛЫМ ЖӘНЕ ЖОҒАРЫ БІЛІМ МИНИСТРЛІГІ

«Л.Н. ГУМИЛЕВ АТЫНДАҒЫ ЕУРАЗИЯ ҰЛТТЫҚ УНИВЕРСИТЕТІ» КЕАҚ

Студенттер мен жас ғалымдардың
«ǴYLYM JÁNE BILIM - 2024»
XIX Халықаралық ғылыми конференциясының
БАЯНДАМАЛАР ЖИНАҒЫ

СБОРНИК МАТЕРИАЛОВ
XIX Международной научной конференции
студентов и молодых ученых
«ǴYLYM JÁNE BILIM - 2024»

PROCEEDINGS
of the XIX International Scientific Conference
for students and young scholars
«ǴYLYM JÁNE BILIM - 2024»

2024
Астана

УДК 001
ББК 72
G99

«ǴYLYM JÁNE BILIM – 2024» студенттер мен жас ғалымдардың XIX Халықаралық ғылыми конференциясы = XIX Международная научная конференция студентов и молодых ученых «ǴYLYM JÁNE BILIM – 2024» = The XIX International Scientific Conference for students and young scholars «ǴYLYM JÁNE BILIM – 2024». – Астана: – 7478 б. - қазақша, орысша, ағылшынша.

ISBN 978-601-7697-07-5

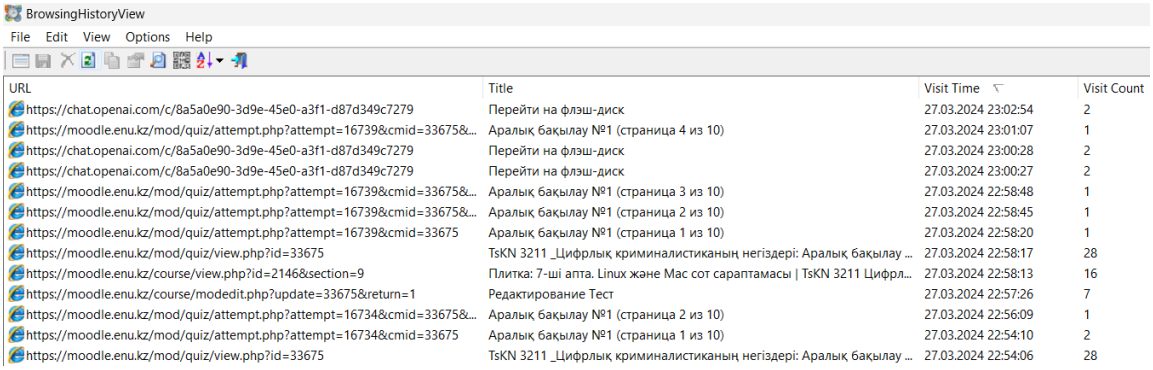
Жинаққа студенттердің, магистранттардың, докоранттардың және жас ғалымдардың жаратылыстану-техникалық және гуманитарлық ғылымдардың өзекті мәселелері бойынша баяндамалары енгізілген.

The proceedings are the papers of students, undergraduates, doctoral students and young researchers on topical issues of natural and technical sciences and humanities.

В сборник вошли доклады студентов, магистрантов, докторантов имолодых ученых по актуальным вопросам естественно-технических и гуманитарных наук.

УДК 001
ББК 72
G99

ISBN 978-601-7697-07-5


BrowsingHistoryView

File Edit View Options Help

URL	Title	Visit Time	Visit Count
https://chat.openai.com/c/8a5a0e90-3d9e-45e0-a3f1-d87d349c7279	Перейти на флэш-диск	27.03.2024 23:02:54	2
https://moodle.enu.kz/mod/quiz/attempt.php?attempt=16739&cmid=33675&...	Аралық бақылау №1 (страница 4 из 10)	27.03.2024 23:01:07	1
https://chat.openai.com/c/8a5a0e90-3d9e-45e0-a3f1-d87d349c7279	Перейти на флэш-диск	27.03.2024 23:00:28	2
https://chat.openai.com/c/8a5a0e90-3d9e-45e0-a3f1-d87d349c7279	Перейти на флэш-диск	27.03.2024 23:00:27	2
https://moodle.enu.kz/mod/quiz/attempt.php?attempt=16739&cmid=33675&...	Аралық бақылау №1 (страница 3 из 10)	27.03.2024 22:58:48	1
https://moodle.enu.kz/mod/quiz/attempt.php?attempt=16739&cmid=33675&...	Аралық бақылау №1 (страница 2 из 10)	27.03.2024 22:58:45	1
https://moodle.enu.kz/mod/quiz/attempt.php?attempt=16739&cmid=33675	Аралық бақылау №1 (страница 1 из 10)	27.03.2024 22:58:20	1
https://moodle.enu.kz/mod/quiz/view.php?id=33675	TsKN 3211 _Цифрлық криминалистиканың негіздері: Аралық бақылау ...	27.03.2024 22:58:17	28
https://moodle.enu.kz/course/view.php?id=2146&section=9	Плитка: 7-ші апта. Linux және Mac сот сараптамасы | TsKN 3211 Цифрл...	27.03.2024 22:58:13	16
https://moodle.enu.kz/course/modedit.php?update=33675&return=1	Редактирование Тест	27.03.2024 22:57:26	7
https://moodle.enu.kz/mod/quiz/attempt.php?attempt=16734&cmid=33675&...	Аралық бақылау №1 (страница 2 из 10)	27.03.2024 22:56:09	1
https://moodle.enu.kz/mod/quiz/attempt.php?attempt=16734&cmid=33675	Аралық бақылау №1 (страница 1 из 10)	27.03.2024 22:54:10	2
https://moodle.enu.kz/mod/quiz/view.php?id=33675	TsKN 3211 _Цифрлық криминалистиканың негіздері: Аралық бақылау ...	27.03.2024 22:54:06	28

Сурет 3. BrowsingHistoryView көмегімен Moodle.enu.kz және GPT чат (chat.openai.com) беттеріне бір уақытта кірудің дәлелі

Осылайша, академиялық адалдықтың бұзылуын дәлелдеу үшін криминалистикалық талдау құралдарын оңай пайдалануға болады.

Болашақта бұл зерттеуді академиялық адалдық қағидаттарының бұзылуы туралы ақпаратты автоматты түрде жинау үшін браузерге енгізілген жеке сценарийді әзірлеу арқылы жалғастыруға болады.

Қолданылған әдебиеттер тізімі

1. Zhang, Q., & Wang, L. (2022). Enhancing Online Testing Security: A Review of Proctoring Solutions. Journal of Information Technology in Education, 37(4), 321-335.
2. Garcia, A., & Rodriguez, P. (2019). Understanding the Effectiveness of Online Proctoring Services: A Case Study. Journal of Online Learning Research, 6(2), 87-102.
3. Johnson, T., & Davis, K. (2020). Leveraging Forensic Browser Tools in Academic Integrity Investigations. International Journal of Cyber Forensics and Advanced Threat Investigations, 8(3), 112-127.
4. Martinez, S., & Garcia, M. (2018). Forensic Analysis of Browser History and Cookies: Tools and Techniques. Journal of Digital Forensics, Security and Law, 13(2), 45-60.

УДК 004.056.53

АНАЛИЗ УЯЗВИМОСТИ ЧТЕНИЯ ПРОИЗВОЛЬНОГО ФАЙЛА В JENKINS (CVE-2024-23897)

Смайлов Адлет Зейноллаевич
adletsmailov@gmail.com
Магистрант 2-ого курса ЕНУ им.Л.Н.Гумилева, Нур-Султан, Казахстан
Научный руководитель – к.ф.-м.н, доцент Сауханова Ж.С.

Аннотация: Статья представляет анализ критической уязвимости, выявленной в январе 2024 года в системе Jenkins, затрагивающей версии 2.441 и предыдущие. Данная уязвимость, обозначенная как CVE-2024-23897, открывает возможность произвольного чтения файлов через интерфейс командной строки (CLI) Jenkins.

Ключевые слова: Jenkins, Arbitrary File Reading, CVE-2024–23897, Чтение произвольного файла, Анализ исходного кода, CI/CD.

Введение

В последние годы Jenkins стал краеугольным инструментом в области непрерывной интеграции и непрерывной доставки (CI/CD). Открытый исходный код и надежный набор функций сделали его лучшим выбором для автоматизации и оптимизации рабочих процессов DevOps. Однако, как и любая программная система, Jenkins не застрахован от уязвимостей. Одной из таких недавно обнаруженных критических уязвимостей является уязвимость чтения произвольных файлов, которая позволяет получить несанкционированный доступ к

конфиденциальной информации, хранящейся в системе. Понимая тонкости этой уязвимости и реализуя рекомендуемые меры безопасности, организации могут защитить свои данные и сохранить доверие заинтересованных сторон.

1. Описание уязвимости

Чтение произвольного файла (Arbitrary File Reading) — это уязвимость в веб-приложениях, которая позволяет злоумышленникам получить доступ к файлам на сервере без необходимости аутентификации. Уязвимость этого типа возникает, когда приложение позволяет пользователю в контролируемом им вводе указать путь к файлу на сервере, без должной проверки или очистки этого ввода со стороны самого приложения. В результате злоумышленники могут манипулировать этими входными параметрами для перемещения по каталогам и чтения конфиденциальных файлов, таких как файлы конфигурации, файлы базы данных или даже файлы исходного кода. В определённых случаях, данная уязвимость может привести к полному контролю над сервером.

В январе 2024 года была выявлена критическая уязвимость в системе CI/CD Jenkins, оценённая по CVSS на уровне 9.8/10 [1][2]. Эта уязвимость представляет критическую угрозу для безопасности информации в её всех аспектах: конфиденциальности, целостности и доступности.

2. Эксплуатация уязвимости

С целью исследования уязвимости предварительно была настроена лабораторная среда на Kali Linux с уязвимой версией Jenkins 2.4.441. Это позволит нам продемонстрировать, как работает данная уязвимость на практике перед её анализом.

Чтобы поднять и настроить лабораторию, необходимо выполнить следующие шаги:

1. Выполнить следующие команды, чтобы установить Jenkins [3]:

sudo apt update

sudo apt install -y docker.io

sudo systemctl enable docker –now

sudo docker pull jenkins/jenkins:2.441-jdk17

2. Выполнить следующую команду, чтобы запустить Jenkins. Сохранить пароль, который появится во время установки:

docker run -p 8080:8080 jenkins/jenkins:2.441-jdk17

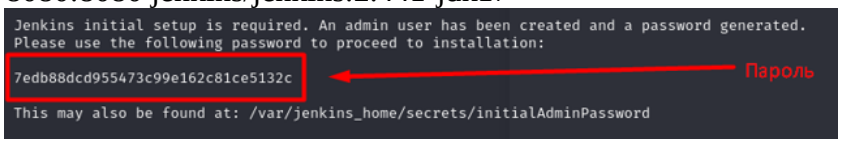


Рисунок 1. Автогенерируемый пароль при установке Jenkins

3. Веб-интерфейс Jenkins доступен по следующему адресу: http://127.0.0.1:8080. При первом входе потребуется ввести сохраненный пароль для авторизации в качестве администратора.

Дальше необходим интерфейс командной строки Jenkins CLI, который можно скачать по указанному пути: http://<Jenkins-URL>/jnlpJars/jenkins-cli.jar.

Для эксплуатации данной уязвимости нужно выполнить следующие команды:

1) В случае, если в настройках Jenkins отключен анонимный доступ, вы сможете прочитать только 1 строчку файла:

java -jar jenkins-cli.jar -s "http://127.0.0.1:8080/" -http help 1 "@/var/jenkins_home/secrets/initialAdminPassword"

Рисунок 2. Эксплуатация уязвимости при отключенном анонимном доступе

В данном случае, в качестве примера был получен доступ к паролю администратора.

2) В случае, если в настройках Jenkins включен анонимный доступ, можно прочитать файл целиком:

java -jar jenkins-cli.jar -s http://127.0.0.1:8080/ -http connect-node 1 "@/etc/hosts"

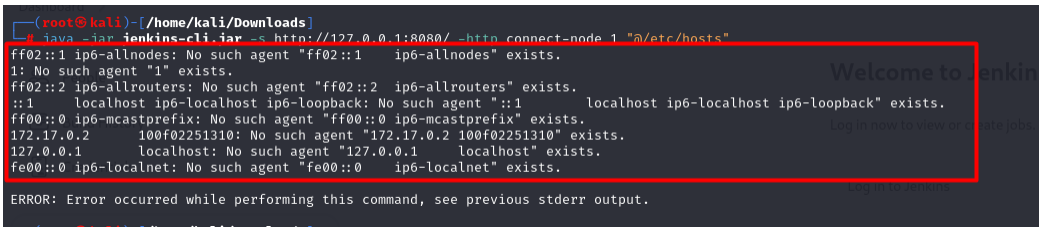


Рисунок 3. Эксплуатация уязвимости при включенном анонимном доступе

В данном случае, в качестве примера был получен доступ к системному файлу /etc/hosts.

3. Анализ исходного кода

Для проведения анализа уязвимости в Jenkins версии 2.441, связанной с Jenkins CLI, необходимо начать с загрузки исходного кода данной версии. Исходный код доступен по следующей ссылке: https://github.com/jenkinsci/jenkins/archive/refs/tags/jenkins-2.441.zip.

После загрузки архива исходного кода необходимо обратить внимание на файл **CLICommand.java**, ответственный за функциональность работы с Jenkins CLI. Этот файл расположен по следующему пути в распакованном архиве:

\jenkins-jenkins-2.441\core\src\main\java\hudson\cli\CLICommand.java

Анализ содержимого файла CLICommand.java позволит выявить и понять проблемные участки кода. Это важный шаг для последующего исправления и обеспечения безопасности системы Jenkins.

Для изучения механизма анализа аргументов и параметров при обработке команд Jenkins CLI, необходимо обратить внимание на библиотеку args4j, используемую в CLICommand.java.

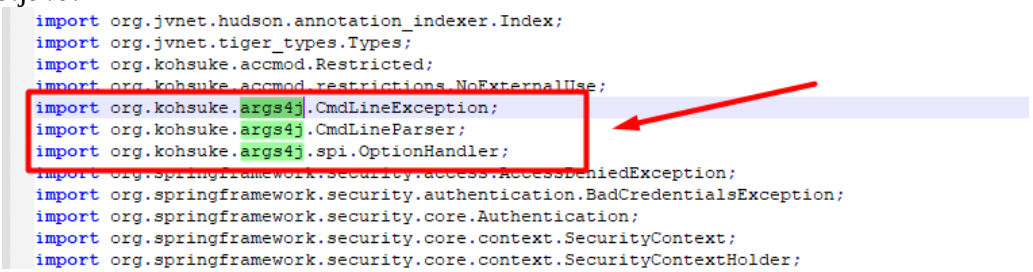


Рисунок 4. Библиотека args4j в Jenkins CLI

Для этого необходимо загрузить исходный код библиотеки args4j доступный по следующей ссылке: https://github.com/kohsuke/args4j/ [4].

После загрузки исходного кода библиотеки args4j, необходимо обратиться к файлу **CmdLineParser.java**, ответственному за анализ аргументов и параметров при работе с Jenkins CLI. Данный файл находится по следующему пути в скачанном репозитории:

\args4j-master\args4j\src\org\kohsuke\args4j\CmdLineParser.java

Анализ содержимого файла **CmdLineParser.java** позволит понять принципы работы библиотеки args4j и ее взаимодействие с командами Jenkins CLI.

Рассмотрим метод **parseArgument()** из файла **CmdLineParser.java**. Данный метод принимает аргументы командной строки в виде строки массива (String[]), иными словами метод обрабатывает аргументы, которые пользователь вводит в Jenkins CLI.

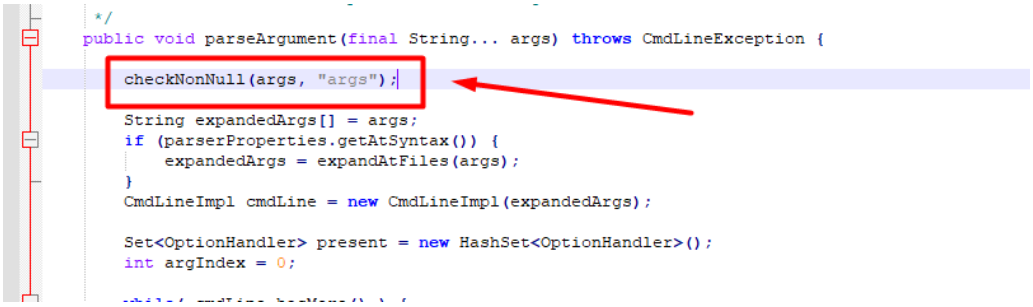


Рисунок 5. Вызов метода checkNonNull () в CmdLineParser.java

Первым шагом в методе parseArgument() идет вызов метода checkNonNull(args, "args"), который находится в файле **Utilities.java**. Этот метод используется для проверки аргументов на null, если аргументы пустые, то возвращается исключение с сообщением «args is null».

Код метода **checkNonNull()** можно найти в файле **Utilities.java** по следующему пути в скачанном репозитории:

\args4j-master\args4j\src\org\kohsuke\args4j\Utilities.java

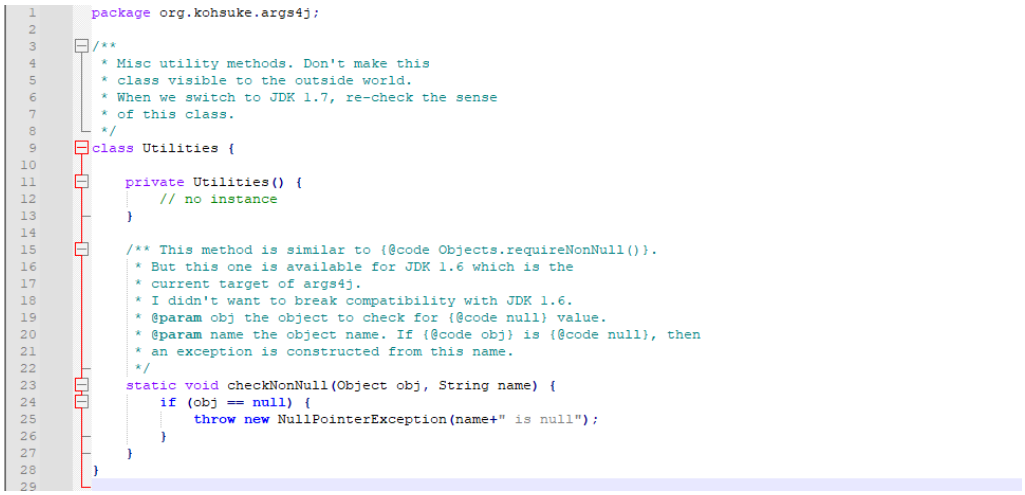


Рисунок 6. Метод checkNonNull () в Utilities.java

Затем создается массив expandedArgs[], который инициализируется переданным массивом args, содержащим аргументы командной строки. После этого осуществляется проверка свойства atSyntax объекта parserProperties.

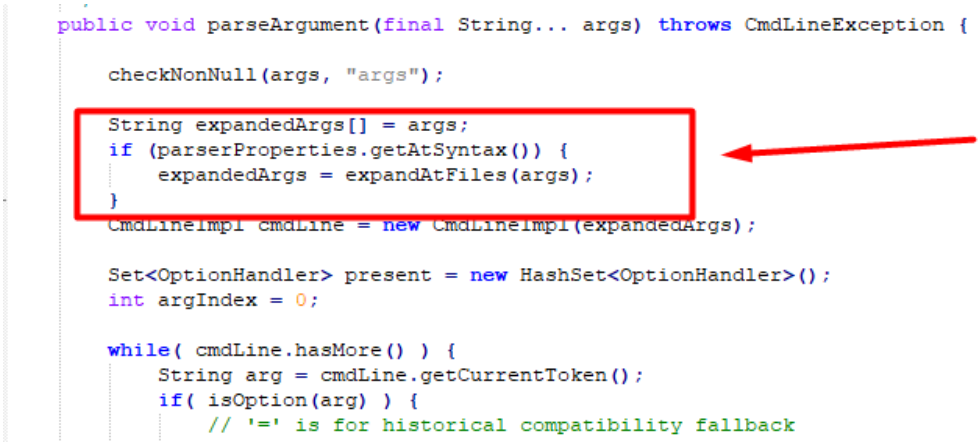


Рисунок 7. Проверка свойства atSyntax

Проверяется условие, если указанное свойство установлено в значении true, то выполнится метод expandAtFiles(args). Разберем метод expandAtFiles() более подробно:

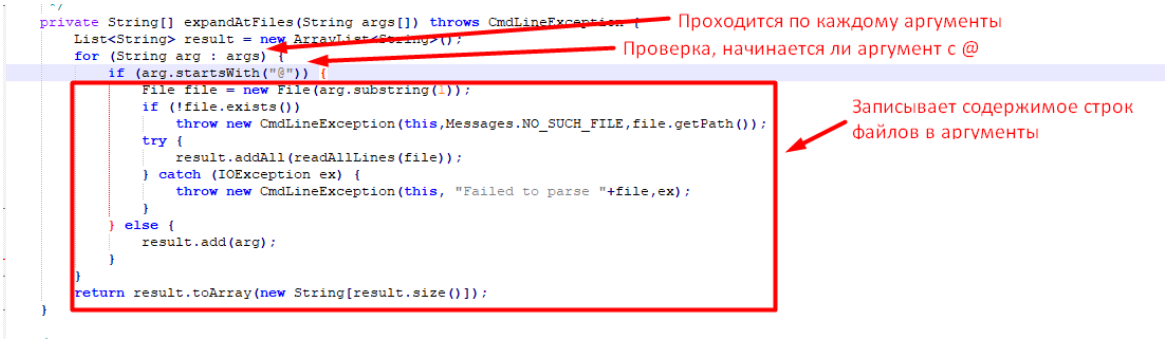


Рисунок 8. Метод expandAtFiles()

Метод **expandAtFiles**(String args[]) проходит по всем элементам массива args. Если элемент начинается с символа '**@**', то метод пытается прочитать файл, указанный в этом элементе, и заменить элемент массива на строки из этого файла, тем самым выводя содержимое произвольного файла, неаутентифицированному пользователю, что и было продемонстрировано в примерах эксплуатации выше.

Метод expandAtFiles() является небезопасным и представляет главную причину возникновения уязвимости произвольного чтения файлов в Jenkins CVE-2024-23897.

Рекомендации по устранению

Обновите Jenkins до версии, начиная с Jenkins 2.442, LTS 2.426.3 и LTS 2.440.1. В этих версиях функция синтаксического анализатора команд отключена, что предотвращает замену символа @ на содержимое файла для команд CLI в аргументах, и тем самым устраняет уязвимость произвольного чтения файлов CVE-2024-23897.

Вывод

Актуальный анализ уязвимости CVE-2024-23897 подчеркивает важность регулярной проверки и обновления зависимостей третьих сторон в информационных системах. Уязвимость, обусловленная использованием устаревшей библиотеки (args4j), подтверждает, что сторонние приложения могут стать ключевым источником слабых мест и угроз в системах. Этот инцидент подчеркивает необходимость активного мониторинга обновлений безопасности в используемых библиотеках и приложениях. Регулярное обновление зависимостей и внимательное отслеживание обновлений помогут предотвратить подобные уязвимости и улучшить общий уровень кибербезопасности.

Список использованных источников

1. National Vulnerability Database (NVD): CVE-2024-23897 Detail. – URL: https://nvd.nist.gov/vuln/detail/CVE-2024-23897 (дата обращения: 16.03.2024)

2. Common Vulnerabilities and Exposures (CVE): CVE-2024-23897. – URL: https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2024-23897 (дата обращения: 16.03.2024)

3. Jenkins Documentation. – URL: https://www.jenkins.io/doc/ (дата обращения: 16.03.2024)

4. Args4j: Java Command-Line Parser Library by Kohsuke Kawaguchi. – URL: https://args4j.kohsuke.org/ (дата обращения: 16.03.2024)

УДК 512.647

МОБИЛЬДІ ҚҰРЫЛҒЫЛАРҒА АРНАЛҒАН ҚҰПИЯ СӨЗ ҚОСЫМШАЛАРЫН ӘЗІРЛЕУДЕ АҚПАРАТТЫҚ ҚАУІПСІЗДІК БОЙЫНША ҰСЫНЫСТАР

Сүлеймен Әлихан Қайратұлы
asuleimenov.21@gmail.com