Рекомендации по устранение уязвимостей в среде разработки программного обеспечения

1. Уязвимость механизма маршрутизации модуля Spring для продвижения бизнес-логики с помощью функции Spring Cloud Function (BDU:2022-01628, уровень опасности по CVSS 2.0 – средний, по CVSS 3.0 – средний), связанная с недостатками процедуры нейтрализации особых элементов в выходных данных, используемых входящим компонентом. Эксплуатация уязвимости может позволить нарушителю, действующему удаленно, получить несанкционированный доступ к локальным ресурсам или вызвать отказ в обслуживании с помощью специально созданного SpEL-выражения.

В целях предотвращения возможности эксплуатации указанной уязвимости необходимо использовать средства межсетевого экранирования прикладного уровня в режиме блокировки для фильтрации HTTP-запросов.

2. Уязвимость программной платформы Spring Framework (BDU:2022-01627, уровень опасности по CVSS 2.0 – средний, по CVSS 3.0 – средний), связанная с неограниченным распределением ресурсов. Эксплуатация уязвимости может позволить нарушителю, действующему удаленно, вызвать отказ в обслуживании с помощью специально созданного SpEL-выражения.

В целях предотвращения возможности эксплуатации указанной уязвимости необходимо использовать средства межсетевого экранирования прикладного уровня в режиме блокировки для фильтрации HTTP-запросов.

3. Уязвимость «нулевого дня» модуля Spring Core программной платформы Spring Framework (BDU:2022-01631, уровень опасности по CVSS 2.0 – критический, по CVSS 3.0 – высокий), связана с применением входных данных с внешним управлением для выбора классов. Эксплуатация уязвимости может позволить нарушителю, действующему удаленно, выполнить произвольный код.

В целях предотвращения возможности эксплуатации указанной уязвимости необходимо:

- осуществить настройку средств межсетевого экранирования прикладного

уровня, позволяющую фильтровать строки, содержащие значения: «class.», «Class.», «.class.» и «.Class.»;

- модифицировать контроллер Spring Framework в части аннотации @InitBinder путем дополнения вызова метода dataBinder.setDisallowedFields строками: «class.», «Class.», «.class.» и «.Class.»;

- добавить в проект глобального класса, обеспечивающего вызов метода dataBinder.setDisallowedFields для обновления в «черный список» строк: «class.», «Class.», «.class.» и «.Class.»;

- использовать JDK версии 8 или более ранние версии.

4. Уязвимость обработчика JavaScript-сценариев V8 браузера Google Chrome (BDU:2022-01822, уровень опасности по CVSS 2.0 – критический, по CVSS 3.0 – критический), связанной со смешением типов данных;

уязвимость (BDU:2022-02336, уровень опасности по CVSS 2.0 – критический, по CVSS 3.0 – высокий), связанная со смешением типов данных;

уязвимость (BDU:2022-03225, уровень опасности по CVSS 2.0 – критический, по CVSS 3.0 – высокий), связанная с возможностью использования памяти после освобождения.

В целях предотвращения возможности эксплуатации указанной уязвимости необходимо принять меры:

-использование средства антивирусной защиты с функцией контроля доступа к веб-ресурсам;

-контролировать доступ в сеть «Интернет» за счет регламентации разрешенных сетевых ресурсов и соединений;

-организовать запуск веб-браузера от имени пользователя минимальными возможными привилегиями в операционной системе;

-использовать альтернативные веб-браузеры;

-применять системы обнаружения вторжений.

5. Уязвимость реализации демона LDAP-auth HTTP-сервера nginx (BDU:2022-02111, уровень опасности по CVSS 2.0 – критический, по CVSS 3.0 – критический), связанная с ошибками в коде. Эксплуатация уязвимости может позволить

нарушителю, действующему удаленно, выполнить произвольный код в уязвимой системе. В целях предотвращения возможности эксплуатации указанной уязвимости необходимо принять меры:

-использовать альтернативные средства аутентификации;

-добавить следующий блок в файле конфигурации nginx-ldap-auth.conf:

location=/auth-proxy {
…
proxy_pass_request_headers off;
proxy_set_header Authorization $http_authorization; # If using Basic auth
…
};

-проверить и удалить демоном LDAP-auth всех специальных символов из поля имени пользователя;

-отключить свойства ldapDaemon.enabled.

6. Уязвимость платформы для управления облачными хранилищами VMware vCloudDirector (BDU:2022-02335, уровень опасности по CVSS 2.0 – высокий, по CVSS 3.0 – критический), связанная с ошибками при обработке входных данных. Эксплуатация указанной уязвимости может позволить нарушителю, действующему удаленно, выполнить произвольный код. В целях предотвращения возможности эксплуатации указанной уязвимости необходимо принять меры:

-использовать средства межсетевого экранирования для предотвращения доступа к платформе управления;

-использовать средства антивирусной защиты для контроля обрабатываемых файлов.

Для указанной версии VMware vCloudDirector 9.7, 10.0, 10.1, 10.2 и 10.3 выполнить следующие шаги:

1.Обратиться по SSH к любой ячейки в группе серверов.

2.Загрузить скрипт WA_CVE-2022-22966.sh в каталог /tmp (файл доступен по ссылке:https://kb.vmware.com/sfc/servlet.shepherd/version/download/0685G00000jt8WpQAI

3.Изменить права доступа к файлу, чтобы разрешить выполнение:

chown root: vcloud /tmp/WA_CVE-2022-22966.sh

chmod 740 /tmp/ WA_CVE-2022-22966.sh

4. Перейти в каталог /tmp ячейки:

cd /tmp

5. Выполнить скрипт:

./WA_CVE-2022-22966.sh

6. Убедиться, что службы в текущей ячейке Cloud Director перезапущены, прежде чем продолжить выполнение сценария в последующих ячейках, выполнить следующую команду:

tail -f /opt/vmware/vcloud-director/logs/cell.log

7. Чтобы убедиться, что исправление было правильно применено, проверить существующую конфигурацию, а также параметры среды выполнения Cloud Director Cell. Для проверки конфигурации выполнить следующую команду:

grep trustSerialData/opt/vmware/vcloud-director/bin/vmware-ved-cell-common

Для проверки настройки среды выполнения подключиться к порту Cells JMX (8999) с помощью jConsole или jmxterm.

1) Чтобы использовать jConsole, необходимо открыть клиент jConsole и подключиться к ячейке.

Открыть вкладку MBeans на верхней панели навигации и следовать дереву навигации: java.lang > Runtime > Attributes > SystemProperties.

Открыть javax.management.openmbean.TabularDataSupport

Найти com.sun.jndi.ldap.object.trustSerialData, а также там должна присутствовать пара ключ-значение, причем значение должно быть ложным (false).

2) Чтобы использовать jConsole, скачайте его по следующей ссылке:

https://docs.cyclopsgroup.org/jmxterm

Запустить следующую команду в командной строке jmxterm, указав правильные параметры:

echo "get --domain java.lang -b type=Runtime SystemProperties" | java -jar –n -l :8999 -u -p " | /bin/grep -A2 com.sun. jndi.ldap.object.trustSerialData

Как и в случаи с jConsole, найти пару ключ-значение, значение должно быть ложным (false).

Повторный запуск сценария после первого выполнения сообщит, что ячейка «Защищена».

8. Последовательно повторить шаги, указанные в пунктах 3-5, пока все ячейки в группе серверов не будут исправлены.

9. Уязвимость интерпретатора языка программирования Scala (BDU:2023-00169, уровень опасности по CVSS 2.0 – критический, по CVSS 3.0 – критический), связанная с ошибками при десериализации данных. Эксплуатация указанной уязвимости может позволить нарушителю, действующему удаленно, выполнить произвольный код.

В целях предотвращения возможности эксплуатации указанной уязвимости рекомендуется установить обновление программного обеспечения в соответствии с Методикой оценки уровня критичности программных, программно-аппаратных средств, утвержденной ФСТЭК России от 28 октября 2022 года.

В случае невозможности установки обновления ПО необходимо принять следующие меры:

-использовать средства межсетевого экранирования для ограничения возможности удаленного доступа;

-использовать антивирусное программное обеспечение для ограничения возможности загрузки нежелательных файлов.