In [1]:
```
#!/usr/bin/env python

import subprocess

subprocess.call("ifconfig eth0 down", shell=True)
subprocess.call("ifconfig eth0 hw ether 00:11:22:33:44:66", shell=True)
subprocess.call("ifconfig eth0 up", shell=True)
```

Out[1]:
```
1
```

The code above is neat, but it can be improved by setting the network interface and the new MAC address as variables rather than hard coded

In [1]:
```
#!/usr/bin/env python

import subprocess

interface = "eth0"
new_mac = "00:11:22:33:44:77"

print("[+] Changing MAC address for " + interface + " to " + new_mac)

subprocess.call("ifconfig " + interface + " down", shell=True)
subprocess.call("ifconfig " + interface + " hw ether "  + new_mac, shell=True)
subprocess.call("ifconfig " + interface + " up", shell=True)
```

Out[1]:
```
[+] Changing MAC address for eth0 to 00:11:22:33:44:77
1
```

Now, let us set the code to take user input for variables interface and new_mac

In [1]:
```
#!/usr/bin/env python

import subprocess

interface = input("interface > ")
new_mac = input("new MAC >")

print("[+] Changing MAC address for " + interface + " to " + new_mac)

subprocess.call("ifconfig " + interface + " down", shell=True)
subprocess.call("ifconfig " + interface + " hw ether "  + new_mac, shell=True)
subprocess.call("ifconfig " + interface + " up", shell=True)
```

Out[1]:
```
interface > eth0
new MAC >00:11:22:33:44
[+] Changing MAC address for eth0 to 00:11:22:33:44
1
```

bing bong

We now want to use parse interpreter and optparse which will allow the user input to come via arguments

In [1]:
```
#!/usr/bin/env python

import subprocess
import optparse
```

```
parser = optparse.OptionParser()

parser.add_option("-i", "--interface", dest="interface", help="Interface to change its MA
parser.add_option("-m", "--mac", dest="new_mac", help="New MAC address")

(options, arguements) = parser.parse_args()

interface = options.interface
new_mac = options.new_mac

print("[+] Changing MAC address for " + interface + " to " + new_mac)

#We now will implement input validation
subprocess.call(["ifconfig", interface, "down"])
subprocess.call(["ifconfig", interface, "hw", "ether", new_mac])
subprocess.call(["ifconfig", interface, "up"])
```

```
Usage: ipykernel_launcher.py [options]

ipykernel_launcher.py: error: no such option: -f
ERROR:root:Internal Python error in the inspect module.
Below is the traceback from this internal error.

Traceback (most recent call last):
  File "C:\Users\Keegz\anaconda3\lib\optparse.py", line 1387, in parse_args
    stop = self._process_args(largs, rargs, values)
  File "C:\Users\Keegz\anaconda3\lib\optparse.py", line 1431, in _process_args
    self._process_short_opts(rargs, values)
  File "C:\Users\Keegz\anaconda3\lib\optparse.py", line 1513, in _process_short_opts
    raise BadOptionError(opt)
optparse.BadOptionError: no such option: -f

During handling of the above exception, another exception occurred:

Traceback (most recent call last):
  File "C:\Users\Keegz\anaconda3\lib\site-packages\IPython\core\interactiveshell.py", line
3444, in run_code
    exec(code_obj, self.user_global_ns, self.user_ns)
  File "C:\Users\Keegz\AppData\Local\Temp/ipykernel_8328/3441666264.py", line 11, in <modu
le>
    (options, arguements) = parser.parse_args()
  File "C:\Users\Keegz\anaconda3\lib\optparse.py", line 1389, in parse_args
    self.error(str(err))
  File "C:\Users\Keegz\anaconda3\lib\optparse.py", line 1569, in error
    self.exit(2, "%s: error: %s\n" % (self.get_prog_name(), msg))
  File "C:\Users\Keegz\anaconda3\lib\optparse.py", line 1559, in exit
    sys.exit(status)
SystemExit: 2

During handling of the above exception, another exception occurred:

Traceback (most recent call last):
  File "C:\Users\Keegz\anaconda3\lib\site-packages\IPython\core\ultratb.py", line 1101, in
get_records
    return _fixed_getinnerframes(etb, number_of_lines_of_context, tb_offset)
  File "C:\Users\Keegz\anaconda3\lib\site-packages\IPython\core\ultratb.py", line 248, in
wrapped
    return f(*args, **kwargs)
  File "C:\Users\Keegz\anaconda3\lib\site-packages\IPython\core\ultratb.py", line 281, in
_fixed_getinnerframes
    records = fix_frame_records_filenames(inspect.getinnerframes(etb, context))
  File "C:\Users\Keegz\anaconda3\lib\inspect.py", line 1541, in getinnerframes
```

```
    frameinfo = (tb.tb_frame,) + getframeinfo(tb, context)
AttributeError: 'tuple' object has no attribute 'tb_frame'
```

```
---------------------------------------------------------------------------
BadOptionError                            Traceback (most recent call last)
~\anaconda3\lib\optparse.py in parse_args(self, args, values)
   1386         try:
-> 1387             stop = self._process_args(largs, rargs, values)
   1388         except (BadOptionError, OptionValueError) as err:

~\anaconda3\lib\optparse.py in _process_args(self, largs, rargs, values)
   1430                 # value(s) for the last one only)
-> 1431                 self._process_short_opts(rargs, values)
   1432             elif self.allow_interspersed_args:

~\anaconda3\lib\optparse.py in _process_short_opts(self, rargs, values)
   1512             if not option:
-> 1513                 raise BadOptionError(opt)
   1514             if option.takes_value():

BadOptionError: no such option: -f

During handling of the above exception, another exception occurred:

SystemExit                                Traceback (most recent call last)
    [... skipping hidden 1 frame]

~\AppData\Local\Temp/ipykernel_8328/3441666264.py in <module>
     10
---> 11 (options, arguements) = parser.parse_args()
     12

~\anaconda3\lib\optparse.py in parse_args(self, args, values)
   1388         except (BadOptionError, OptionValueError) as err:
-> 1389             self.error(str(err))
   1390

~\anaconda3\lib\optparse.py in error(self, msg)
   1568         self.print_usage(sys.stderr)
-> 1569         self.exit(2, "%s: error: %s\n" % (self.get_prog_name(), msg))
   1570

~\anaconda3\lib\optparse.py in exit(self, status, msg)
   1558             sys.stderr.write(msg)
-> 1559         sys.exit(status)
   1560

SystemExit: 2

During handling of the above exception, another exception occurred:

TypeError                                 Traceback (most recent call last)
    [... skipping hidden 1 frame]

~\anaconda3\lib\site-packages\IPython\core\interactiveshell.py in showtraceback(self, exc_
tuple, filename, tb_offset, exception_only, running_compiled_code)
   2055                     stb = ['An exception has occurred, use %tb to see '
   2056                            'the full traceback.\n']
-> 2057                     stb.extend(self.InteractiveTB.get_exception_only(etype,
   2058                                                                      value))
   2059                 else:

~\anaconda3\lib\site-packages\IPython\core\ultratb.py in get_exception_only(self, etype, v
alue)
    752         value : exception value
```

```
    753         """
--> 754         return ListTB.structured_traceback(self, etype, value)
    755 
    756     def show_exception_only(self, etype, evalue):

~\anaconda3\lib\site-packages\IPython\core\ultratb.py in structured_traceback(self, etype,
evalue, etb, tb_offset, context)
    627             chained_exceptions_tb_offset = 0
    628             out_list = (
--> 629                 self.structured_traceback(
    630                     etype, evalue, (etb, chained_exc_ids),
    631                     chained_exceptions_tb_offset, context)

~\anaconda3\lib\site-packages\IPython\core\ultratb.py in structured_traceback(self, etype,
value, tb, tb_offset, number_of_lines_of_context)
   1365         else:
   1366             self.tb = tb
-> 1367         return FormattedTB.structured_traceback(
   1368             self, etype, value, tb, tb_offset, number_of_lines_of_context)
   1369 

~\anaconda3\lib\site-packages\IPython\core\ultratb.py in structured_traceback(self, etype,
value, tb, tb_offset, number_of_lines_of_context)
   1265         if mode in self.verbose_modes:
   1266             # Verbose modes need a full traceback
-> 1267             return VerboseTB.structured_traceback(
   1268                 self, etype, value, tb, tb_offset, number_of_lines_of_context
   1269             )

~\anaconda3\lib\site-packages\IPython\core\ultratb.py in structured_traceback(self, etype,
evalue, etb, tb_offset, number_of_lines_of_context)
   1122         """Return a nice text document describing the traceback."""
   1123 
-> 1124         formatted_exception = self.format_exception_as_a_whole(etype, evalue, etb,
number_of_lines_of_context,
   1125                                                                tb_offset)
   1126 

~\anaconda3\lib\site-packages\IPython\core\ultratb.py in format_exception_as_a_whole(self,
etype, evalue, etb, number_of_lines_of_context, tb_offset)
   1080 
   1081 
-> 1082         last_unique, recursion_repeat = find_recursion(orig_etype, evalue, records
)
   1083 
   1084         frames = self.format_records(records, last_unique, recursion_repeat)

~\anaconda3\lib\site-packages\IPython\core\ultratb.py in find_recursion(etype, value, reco
rds)
    380     # first frame (from in to out) that looks different.
    381     if not is_recursion_error(etype, value, records):
--> 382         return len(records), 0
    383 
    384     # Select filename, lineno, func_name to track frames with

TypeError: object of type 'NoneType' has no len()
```

In [ ]: