EvilPostman

Graficzny modyfikator pakietów On the fly

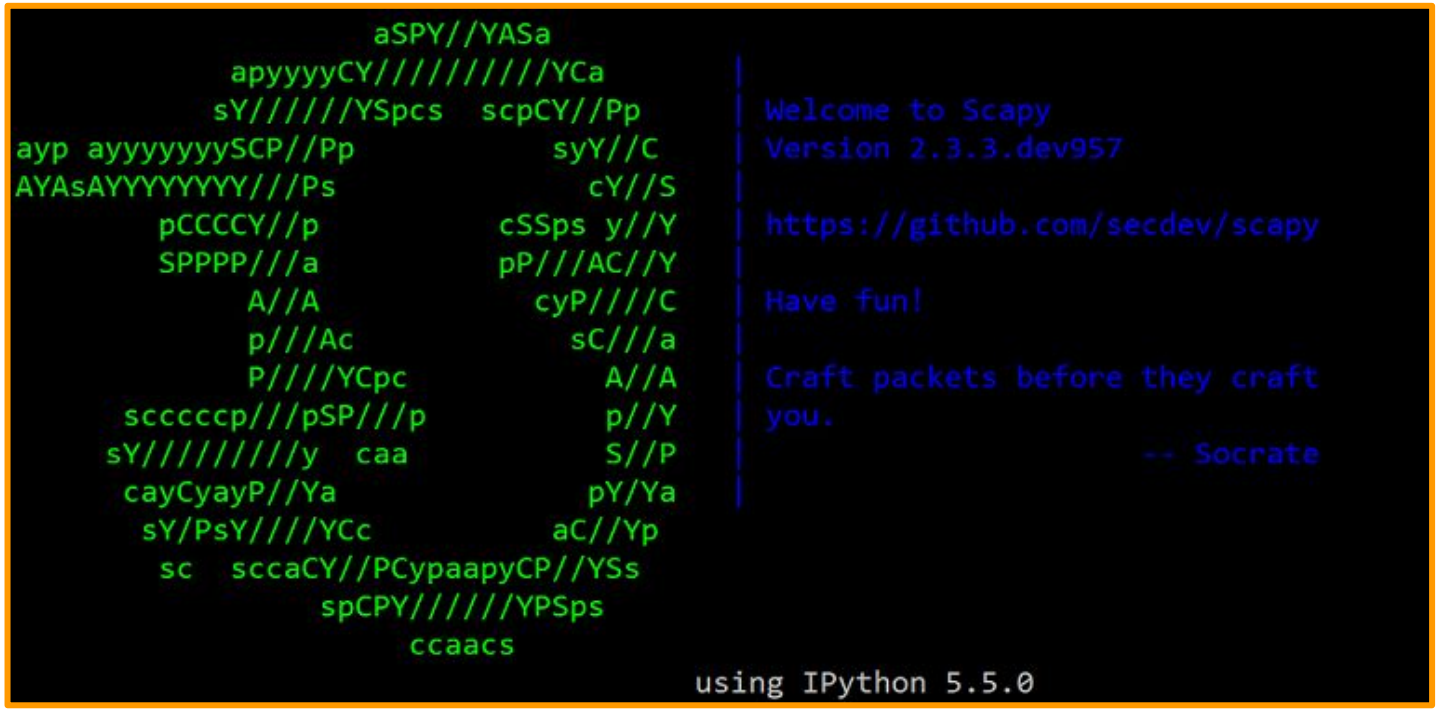
Zespół

- Jarosław Wieczorek
- Krzysztof Orczyk
- Dominika Pawlaczyk

Prowadzący

- Mgr inż. Przemysław Walkowiak

Scapy


aSPY//YASa
apyyyyCY//////////YCa
sY//////YSpcs scpCY//Pp
ayp ayyyyyyySCP//Pp syY//C
AYAsAYYYYYYYY///Ps cY//S
pCCCCY//p cSSps y//Y
SPPPP///a pP///AC//Y
A//A cyP////C
p///Ac sC///a
P////YCpc A//A
scccccp///pSP///p p//Y
sY/////////y caa S//P
cayCyayP//Ya pY/Ya
sY/PsY////YCc aC//Yp
sc sccaCY//PCypaapyCP//YSs
spCPY//////YPSps
ccaacs
Welcome to Scapy
Version 2.3.3.dev957
https://github.com/secdev/scapy
Have fun!
Craft packets before they craft you.
-- Socrate
using IPython 5.5.0

Generowanie pakietów Scapy

Tworzenie nowego pakietu

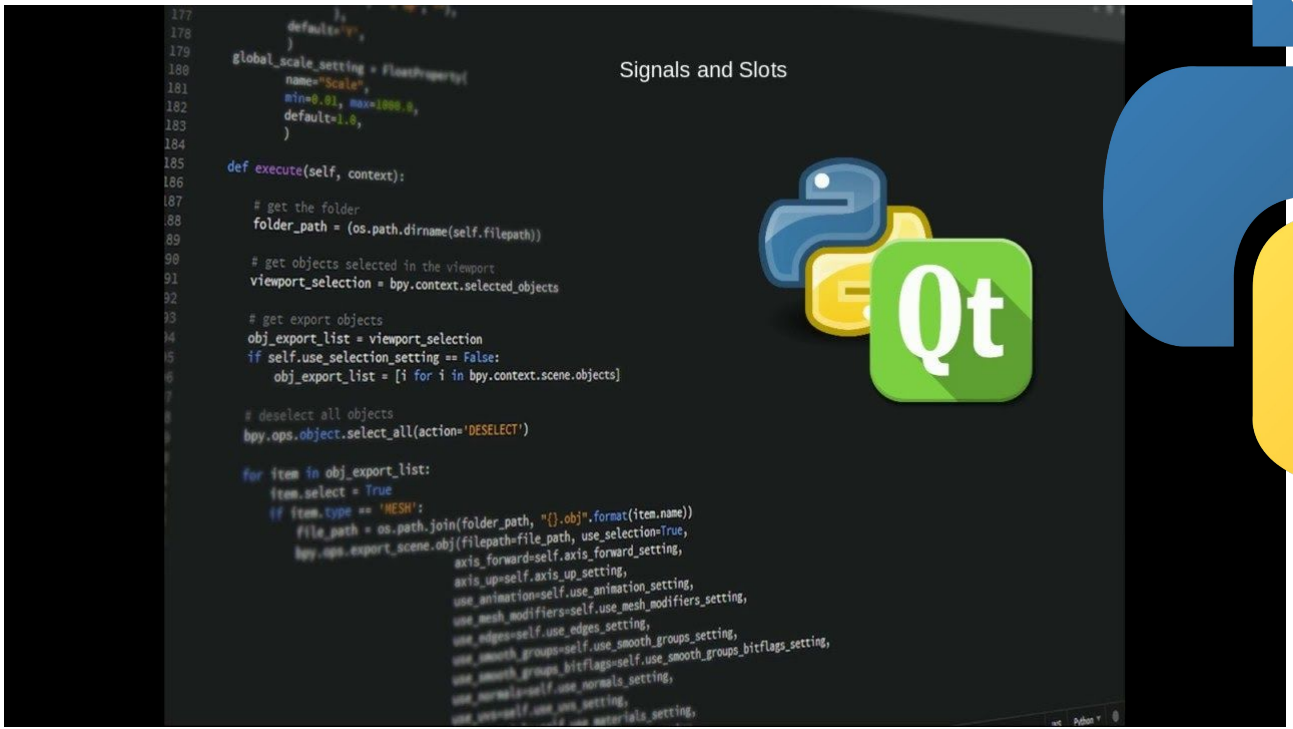
W scapy jest to prosta operacja: https://sekurak.pl/generator-pakietow-scapy/

```
>>> p = ICMP()
```

Wyświetlenie szczegółów pakietu

```
>>> p.show()
###[ ICMP ]###
  type= echo-request
  code= 0
  chksum= None
  id= 0x0
  seq= 0x0
```

PyQT5 Gui



Funkcjonalności gui

Wyświetlanie informacji o odebranych pakietach

Możliwość edycji pakietów

Ułatwienie przetwarzania danych

Podgląd “Raw data”

Odczyt pakietów

Packet # 1019

0000 FFFFFFFFFFFF000000000000008004500
.....E.

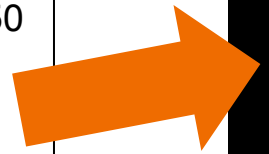
0010 001C00010000040017CDE7F0000017F00
.....@.|.....

0020 00010800F7FF00000000 .....

Krótki pokaz przesyłania danych

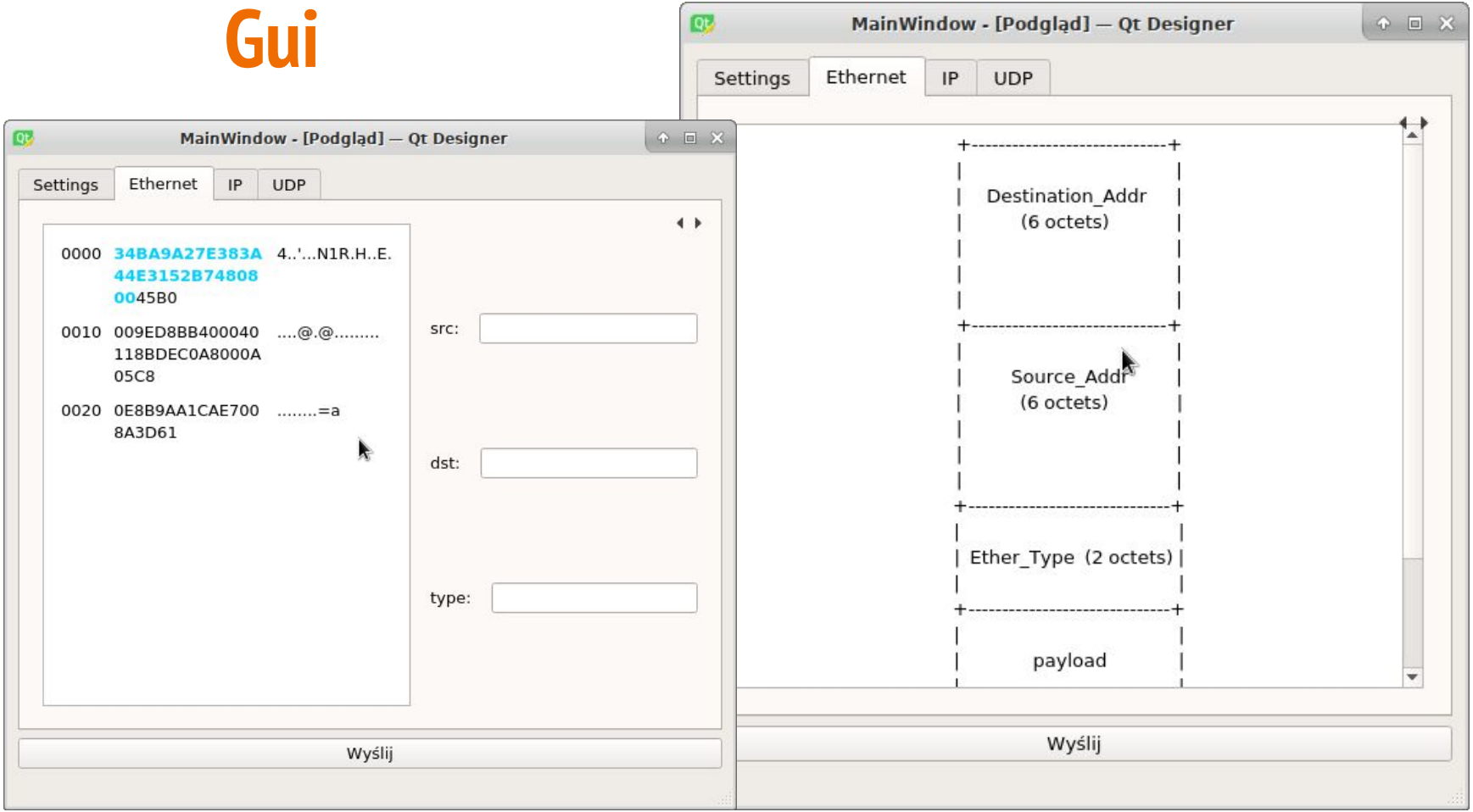
Dekodowanie pakietów w Scapy

```
0000
FFFFFFFFFFFF000000000000080045
0 ..............E.

0010
001C0001000040017CDE7F0000017F0
0 ......@.|.......

0020  00010800F7FF00000000
..........
```



```
Packet # 1019
###[ Ethernet ]###
  dst= ff:ff:ff:ff:ff:ff
  src= 00:00:00:00:00:00
  type= 0x800
###[ IP ]###
     version= 4
     ihl= 5
     tos= 0x0
     len= 28
     id= 1
     flags=
     frag= 0
     ttl= 64
     proto= icmp
     chksum= 0x7cde
     src= 127.0.0.1
     dst= 127.0.0.1
     \options\
###[ ICMP ]###
        type= echo-request
        code= 0
        chksum= 0xf7ff
        id= 0x0
        seq= 0x0
```

Gui


MainWindow - [Podgląd] — Qt Designer
Settings
Ethernet
IP
UDP
0000 34BA9A27E383A 4..'...N1R.H..E.
44E3152B74808
00045B0
0010 009ED8BB400040 ....@.@.....
118BDEC0A8000A
05C8
0020 0E8B9AA1CAE700 .....=a
8A3D61
src:
dst:
type:
Wyślij
MainWindow - [Podgląd] — Qt Designer
Settings
Ethernet
IP
UDP
Destination_Addr
(6 octets)
Source_Addr
(6 octets)
Ether_Type (2 octets)
payload
Wyślij

Użyte Technologie

Python 3.x

PyQt5

Scapy

Spyder 3

Sublime Text Editor 3

Ubuntu 16.04

Przykładowy proces modyfikacji pakietu

1. Uruchomienie programu
2. Wybór trybu działania

    a. Wybranie pakietu z przechwyconych, ręczna modyfikacja parametrów
    Modyfikowanie przychodzących pakietów na podstawie utworzonego filtru

    b. Modyfikacja wybranych danych.

3. Wysyłanie pakietów

Tworzenie struktury projektu w celu usprawnienia pracy z interfejsami PyQt5

```
$ mkdir -p evilpostman/gui

$ touch setup.py myapp/{__init__,__main__}.py evilpostman/gui/__init__.py

$ mv mainwindow.ui ./evilpostman/gui/mainwindow.ui
```

Generate json

```
$ pyuicfg -g --pyqt5

      pyuic.json generated

$ cat pyuic.json
```

```
}
    "files": [],
    "hooks": [],
    "pyrcc": "pyrcc5",
    "pyrcc_options": "",
    "pyuic": "pyuic5",
    "pyuic_options":
"--from-import"
}
```

Przygotowanie struktury pliku JSON i wykorzystanie

evilpostman główny katalog z projektem

evilpostman/gui katalog zawierający interfejsy z rozszerzeniem ***.ui**

evilpostman/gui/styles
katalog zawierający pliki z rozszerzeniem ***.qss**

evilpostman/gui/resources
katalog zawierający pliki z rozszerzeniem ***.qrc**

```
{
"files":
[
 [
        "evilpostman/gui/*.ui",
        "evilpostman/gui"
 ],
 [      "evilpostman/gui/styles/*.qss",
        "evilpostman/gui/styles"
 ],
 [      "evilpostman/gui/resources/*.qrc",
        "evilpostman/gui/resources"
 ]
],
        "hooks": [],
        "pyrcc": "pyrcc5",
        "pyrcc_options": "",
        "pyuic": "pyuic5",
        "pyuic_options": "--from-import"
}
```

Utworzenie narzędzia do automatycznej budowy interfejsów

$ cat setup.py

```
from setuptools import setup

try:
    from pyqt_distutils.build_ui import build_ui
    cmdclass = {"build_ui": build_ui}
except ImportError:
    cmdclass = {}

setup(
    name="evilpostman",
    version="0.1",
    packages=["evilpostman"],
    cmdclass=cmdclass,
)
```

Budowa aplikacji

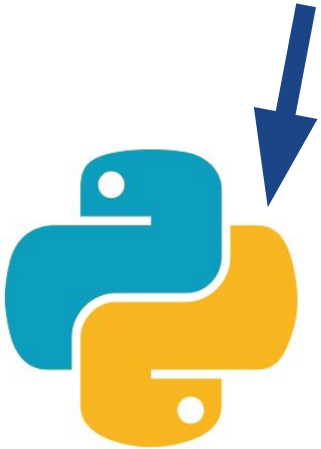
$ python setup.py build_ui

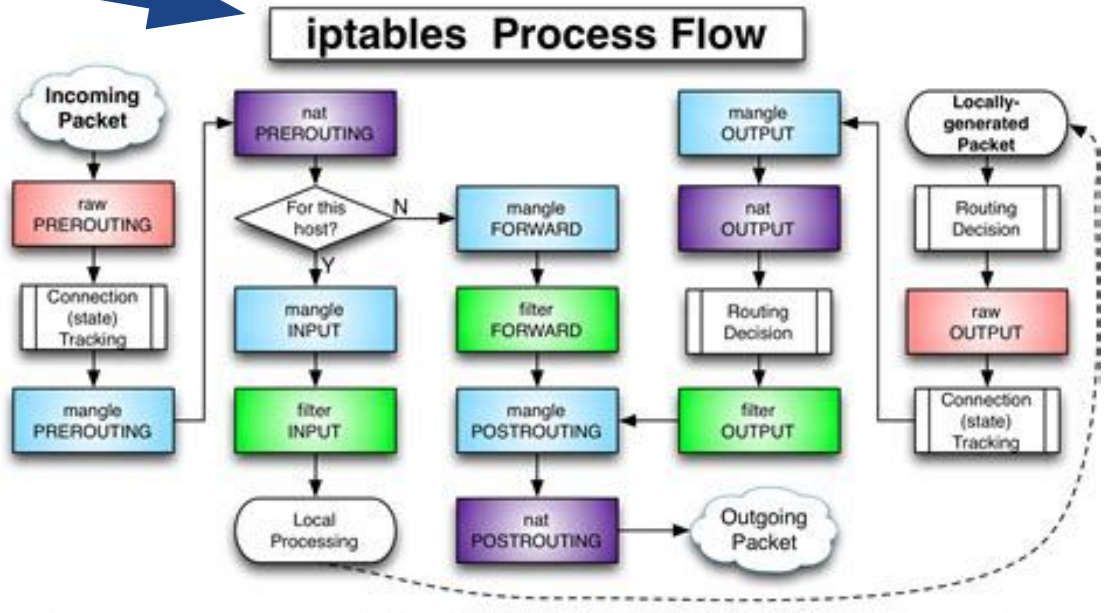
```
(env) [redacted]:~/Modifing_Packets_On-The-Fly_In_SCAPY$ python3 setup.py build_ui
running build_ui
skipping [redacted]/Modifing_Packets_On-The-Fly_In_SCAPY/evilpostman/gui/dialog.ui, up to date
pyuic5 --from-import [redacted]/Modifing_Packets_On-The-Fly_In_SCAPY/evilpostman/gui/mainwindow.ui
-o [redacted]/Modifing_Packets_On-The-Fly_In_SCAPY/evilpostman/gui/mainwindow_ui.py
(env) [redacted]:~/Modifing_Packets_On-The-Fly_In_SCAPY$
```

Implementacja przechwytywania i wyświetlania pakietów

netfilter
firewalling, NAT, and packet mangling for linux

nfqueue





NFQUEUE

$ iptables -A INPUT -j NFQUEUE --queue-num 1

```
sudo iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source      destination
NFQUEUE    all  --  anywhere    anywhere      NFQUEUE num 1
```

Python nfqueue

Python nfqueue bindings: [python-netfilterqueue]

```
def start_capture(self):
    nfqueue = NetfilterQueue()
    nfqueue.bind(1, self.modify, mode=COPY_PACKET)
    try:
        print("Begining capture.")
        nfqueue.run()
        #self.accept_all()
    except KeyboardInterrupt:
        pass
    print("Worked.")
```

Tryby działania aplikacji

Ręczna edycja wybranego pakietu z listy przechwyconych

Automatyczne filtrowanie przechwyconych pakietów

Filtry

Pozwolą na modyfikacje wybranych pól/pakietów przy spełnieniu określonych warunków:

Wybrane pola ==/!= wartość

Domyślnie, każdy pakiet wybranego rodzaju będzie modyfikowalny

Bibliografia

https://sekurak.pl/generator-pakietow-scapy/

https://helion.pl/ksiazki/black-hat-python-jezyk-python-dla-hakerow-i-pentesterow-justin-seitz,blahap.htm#format/d

http://scapy.readthedocs.io/en/latest/usage.html

https://theitgeekchronicles.files.wordpress.com/2012/05/scapyguide1.pdf

https://helion.pl/ksiazki/zapory-sieciowe-w-systemie-linux-kompendium-wiedzy-o-nftables-wydanie-iv-steve-suehring,zasili.htm#format/d