Scapy : les commandes de base

Utilisation de Scapy en interactif

Pour pouvoir manipuler les paquets réseaux, il est nécessaire d'être root pour une majorité de tâches lors du lancement de Scapy

- Lancement de l'interpréteur Python avec Scapy :

```
root@debian:~# scapy
INFO: Please, report issues to https://github.com/phaethon/scapy
WARNING: IPython not available. Using standard Python shell instead.
Welcome to Scapy (3.0.0)
>>>
```

Utilisation de Scapy dans un script Python

Pour pouvoir **utiliser** Scapy dans un **script Python**, il faut inclure la bibliothèque Scapy avec l'instruction :

```
from scapy.all import *
```

Les commandes de base

- Connaître la **liste des protocoles** supportés par Scapy :

```
>>> ls()
AH          : AH
ARP         : ARP
ASN1_Packet : None
BOOTP       : BOOTP
CAN         : CAN
CookedLinux : cooked linux
DHCP        : DHCP options
DHCP6       : DHCPv6 Generic Message)
...
```

Plus de 150 protocoles réseaux supportés dont Ethernet, IP, IPv6, TCP, UDP, DNS, ICMP, DHCP, ARP, BOOTP, NetBIOS, NTP, Radius, SNMP, TFTP, etc.

Pour **visualiser** les informations contenues dans un objet de protocole (avec les valeurs par défaut) :

```
>>> ls(ARP)
hwtype     : XShortField           = (1)
ptype      : XShortEnumField       = (2048)
hwlen      : ByteField             = (6)
plen       : ByteField             = (4)
op         : ShortEnumField        = (1)
hwsrc      : ARPSourceMACField     = (None)
psrc       : SourceIPField         = (None)
hwdst      : MACField              = ('00:00:00:00:00:00')
pdst       : IPField               = ('0.0.0.0')
>>>
```

- connaître les **fonctions de base** de Scapy (environ une vingtaine):

```
>>> lsc()
arpcachepoison      : Poison target's cache with (your MAC,victim's IP)
couple
arping              : Send ARP who-has requests to determine which hosts are
up
bind_layers         : Bind 2 layers on some specific fields' values
bridge_and_sniff    : Forward traffic between two interfaces and sniff
packets exchanged
corrupt_bits        : Flip a given percentage or number of bits from bytes
...
```

- se documenter sur une fonction de Scapy. Exemple pour la fonction **send** :

```
>>> help(send)
Help on function send in module scapy.sendrecv:

send(x, inter=0, loop=0, count=None, verbose=None, realtime=None, *args,
**kargs)
    Send packets at layer 3
    send(packets, [inter=0], [loop=0], [verbose=conf.verb]) -> None
(END)
```

- se documenter sur une méthode proposée par un protocole supporté par Scapy. Exemple pour le protocole **IP** :

```
>>> dir(IP)
['__bool__', '__bytes__', '__class__', '__contains__', '__delattr__',
'__delitem__', '__dict__', '__dir__', '__div__', '__doc__', '__eq__',
'__format__', '__ge__', '__getattr__', '__getattribute__', '__getitem__',
'__gt__', '__hash__', '__init__', '__iter__', '__le__', '__len__', '__lt__',
'__module__', '__mul__', '__ne__', '__new__', '__rdiv__', '__reduce__',
'__reduce_ex__', '__repr__', '__rmul__', '__rtruediv__', '__setattr__',
'__setitem__', '__sizeof__', '__str__', '__subclasshook__', '__truediv__',
'__weakref__', '_do_summary', 'add_payload', 'add_underlayer', 'aliastypes',
'answers', 'build', 'build_done', 'build_padding', 'build_ps',
```

```
'canvas_dump', 'clone_with', 'command', 'copy', 'decode_payload_as',
'default_payload_class', 'delfieldval', 'display', 'dissect',
'dissection_done', 'do_build', 'do_build_payload', 'do_build_ps',
'do_dissect', 'do_dissect_payload', 'do_init_fields', 'explicit',
'extract_padding', 'fields_desc', 'firstlayer', 'fragment', 'from_hexcap',
'get_field', 'getbyteval', 'getdictval', 'getfield_and_val', 'getfieldval',
'getlayer', 'getstrval', 'guess_payload_class', 'hashret', 'haslayer',
'hide_defaults', 'hops', 'init_fields', 'initialized', 'is_priv_addr',
'lastlayer', 'libnet', 'lower_bonds', 'mysummary', 'name', 'ottl',
'overload_fields', 'payload_guess', 'pdfdump', 'post_build', 'post_dissect',
'post_dissection', 'pre_dissect', 'psdump', 'raw_packet_cache',
'remove_payload', 'remove_underlayer', 'route', 'self_build', 'send',
'sent_time', 'setfieldval', 'show', 'show2', 'show_indent', 'sprintf',
'summary', 'underlayer', 'upper_bonds', 'whois']
```

Retour à Python : la bibliothèque Scapy ...

- Python : la bibliothèque Scapy pour manipuler les paquets réseau