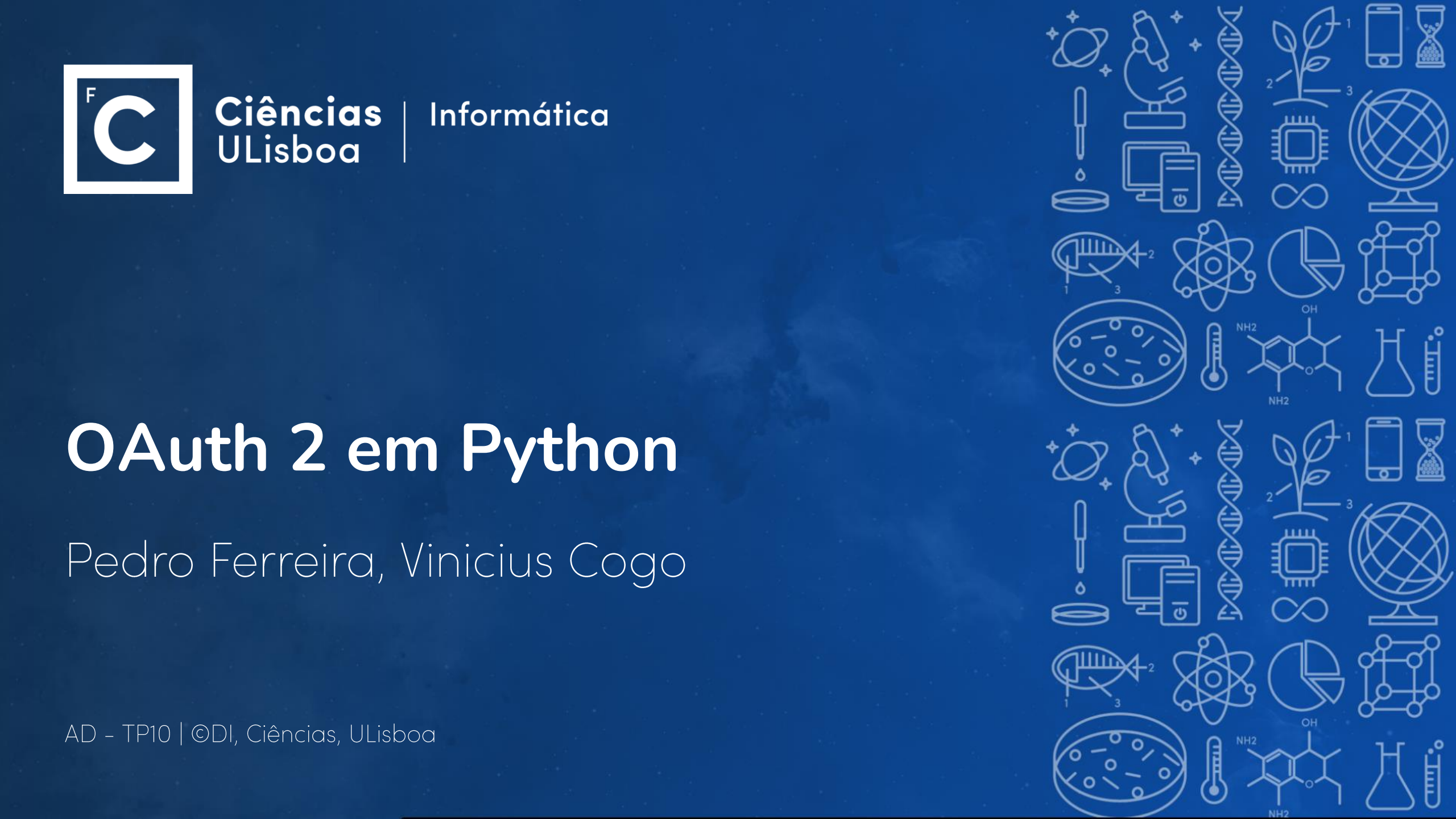


OAuth 2 em Python

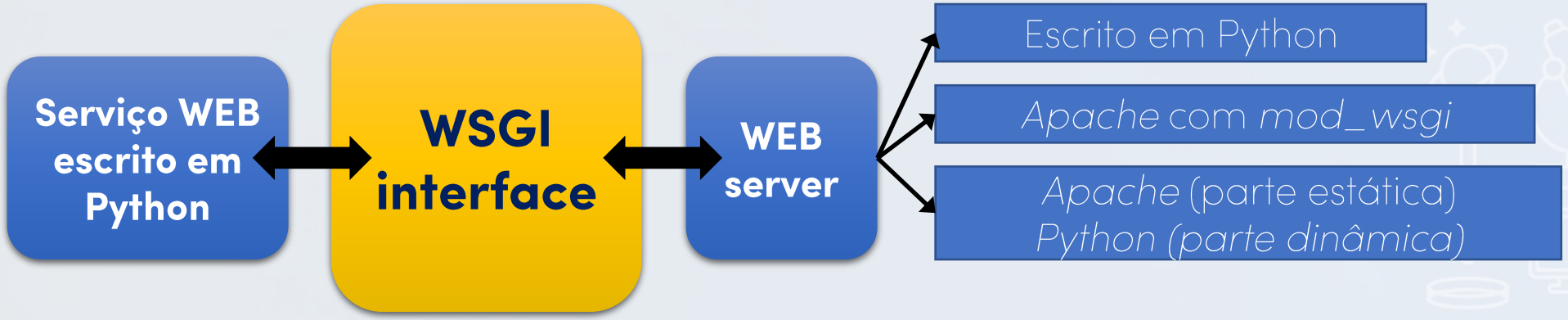
Pedro Ferreira, Vinicius Cogo

Tópicos

1. Servidores de HTTP(S) em Python
2. Server-side web application flow
3. Aplicação Básica com Oauth
4. Aplicação Flask com OAuth

1. Servidores de HTTP em Python

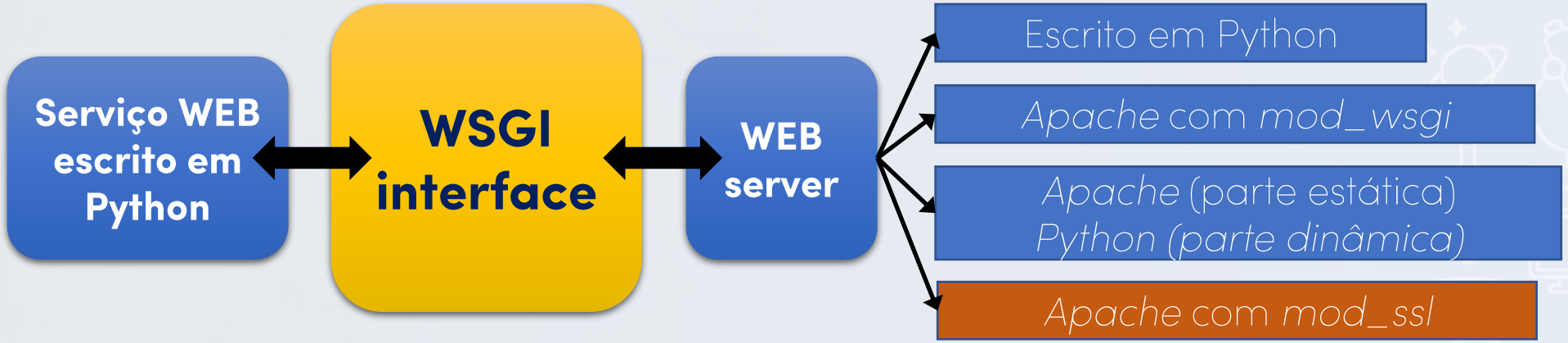
- Opções clássicas dependem de mecanismos diferentes de interface aplicação/servidor
- Forma padronizada introduzida no PEP[1] 3333: *Web Server Gateway Interface* (WSGI)



1: *PEP: Python Enhancement Proposal*

1. Servidores de HTTPS em Python

- Conexão com TLS/SSL
- Utilização de OAuth 2: fornece uma interface em Python para a conexão de clientes OAuth



2. Server-side web application flow

Características:

- Protocolo em dois passos:
  requer um código de autorização e um token de acesso
- Suporta a renovação de tokens durante a ligação
- O código de autorização é usado para
  a obtenção do token de acesso e a renovação deste

Implementação:

- Registar a aplicação cliente num serviço de fornecimento de OAuth
  (p.ex., GitHub, Facebook, Google, Twitter) com o URI de redireccionamento,
  para receber um ID e um segredo para a obtenção de autorização neste
- Configurar a Aplicação (p. ex., Flask) com estas credenciais,
  tendo em conta o fornecedor do OAuth.

3. Aplicação Básica com OAuth

Aplicação básica ligada à API OAuth 2 do Github

Código mínimo

```
from requests_oauthlib import OAuth2Session

client_id = '<the id you get from github>'
client_secret = '<the secret you get from github>'

authorization_base_url = 'https://github.com/login/oauth/authorize'
token_url = 'https://github.com/login/oauth/access_token'

github = OAuth2Session(client_id)
authorization_url, state = github.authorization_url(authorization_base_url)
print('Please go here and authorize: ', authorization_url)
redirect_response = raw_input('Paste the full redirect URL here:')
github.fetch_token(token_url, client_secret=client_secret,authorization_response=redirect_response)

r = github.get('https://api.github.com/user')
print(r.content)
```

4. Aplicação Flask com OAuth

Aplicação Flask ligada à API OAuth 2 do Github

Configuração geral

```
from requests_oauthlib import OAuth2Session
from flask import Flask, request, make_response, redirect, url_for, jsonify
...
app = Flask(__name__)

# Informação da aplicação registada no GitHub
client_id = "<your client key>"
client_secret = "<your client secret>"

redirect_uri= 'http://localhost:5000/callback'
github = OAuth2Session(client_id, redirect_uri=redirect_uri)
```

4. Aplicação Flask com OAuth

Aplicação Flask ligada à API OAuth 2 do Github

Autorização do Utilizador: redireciona o utilizador/dono do recurso para o fornecedor de OAuth (Github), usando a URL de autorização

```
@app.route('/login', methods = ["GET"])
def login():
    global github
    authorization_base_url = 'https://github.com/login/oauth/authorize'
    authorization_url, state = github.authorization_url(authorization_base_url)
    return redirect(authorization_url)
```

4. Aplicação Flask com OAuth

Aplicação Flask ligada à API OAuth 2 do Github

Aplicação autorizada: a aplicação, após autorizada, é redirecionada para o URI que registou como callback. O URL contém o authorization-code necessário para obter o token de acesso

```
@app.route('/callback', methods = ["GET"])
def callback():
    global github
    token_url = 'https://github.com/login/oauth/access_token'
    github.fetch_token(token_url,client_secret=client_secret,authorization_response=request.url)
    return redirect(url_for('.profile'))
```

4. Aplicação Flask com OAuth

Aplicação Flask ligada à API OAuth 2 do Github

Utilização de um recurso protegido: a aplicação utiliza o token obtido para aceder a um recurso protegido, usando o token.

```
@app.route("/profile", methods=["GET"])
def profile():
    global github
    protected_resource = 'https://api.github.com/user'
    return jsonify(github.get(protected_resource).json())

# execução da aplicação flask
if __name__ == "__main__":
    # This allows us to use a plain HTTP callback
    os.environ['OAUTHLIB_INSECURE_TRANSPORT'] = '1'

    app.run(debug=True)
```

Bibliografia

- Brandon Rhodes and John Goerzen. Foundations of Python Network Programing, second edition, Apress.

- Flask User's guide: https://flask.palletsprojects.com/en/1.1.x/

- Flask API: https://flask.palletsprojects.com/en/1.1.x/api/

- Módulo requests-oauthlib:
  - https://pypi.python.org/pypi/requests-oauthlib
  - http://requests-oauthlib.readthedocs.io/en/latest/index.html