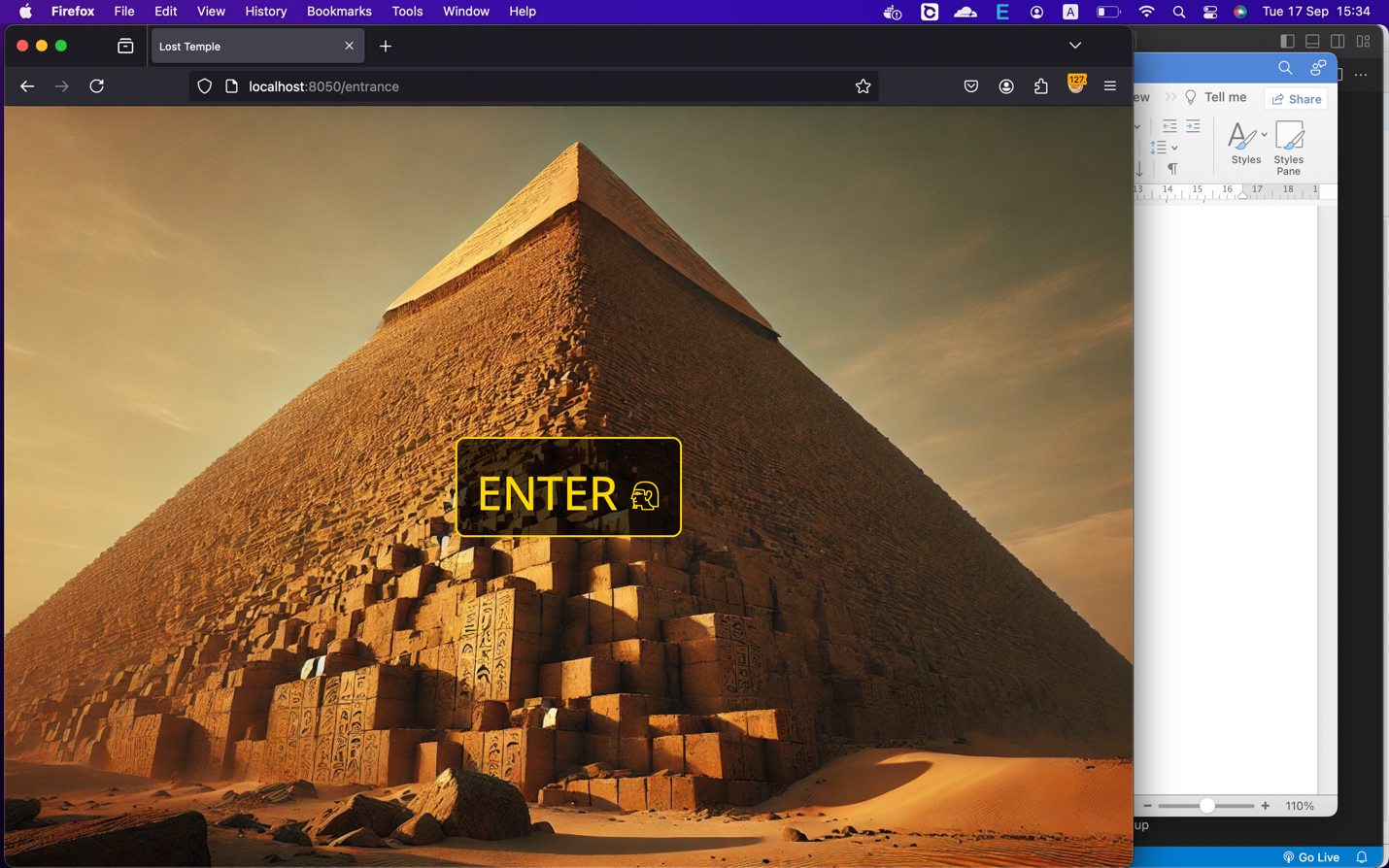
**Web Enumeration**

Giao diện trang web là **/entrance**

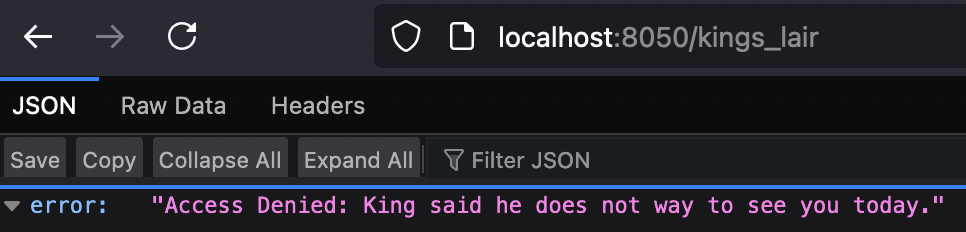


Khi truy cập /entrance, chúng ta sẽ được cấp cookie như sau:



EdDSA là thuật toán bất đối xứng.

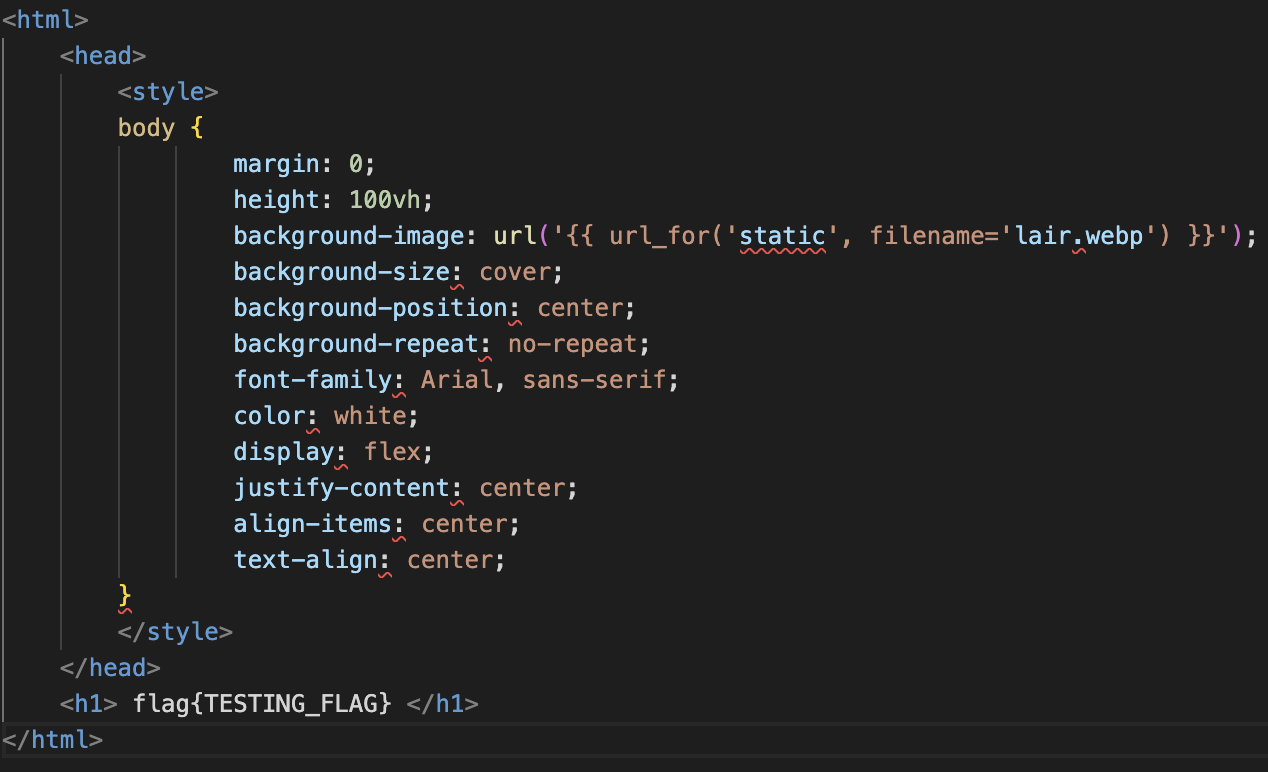
**/kings\_lair:**



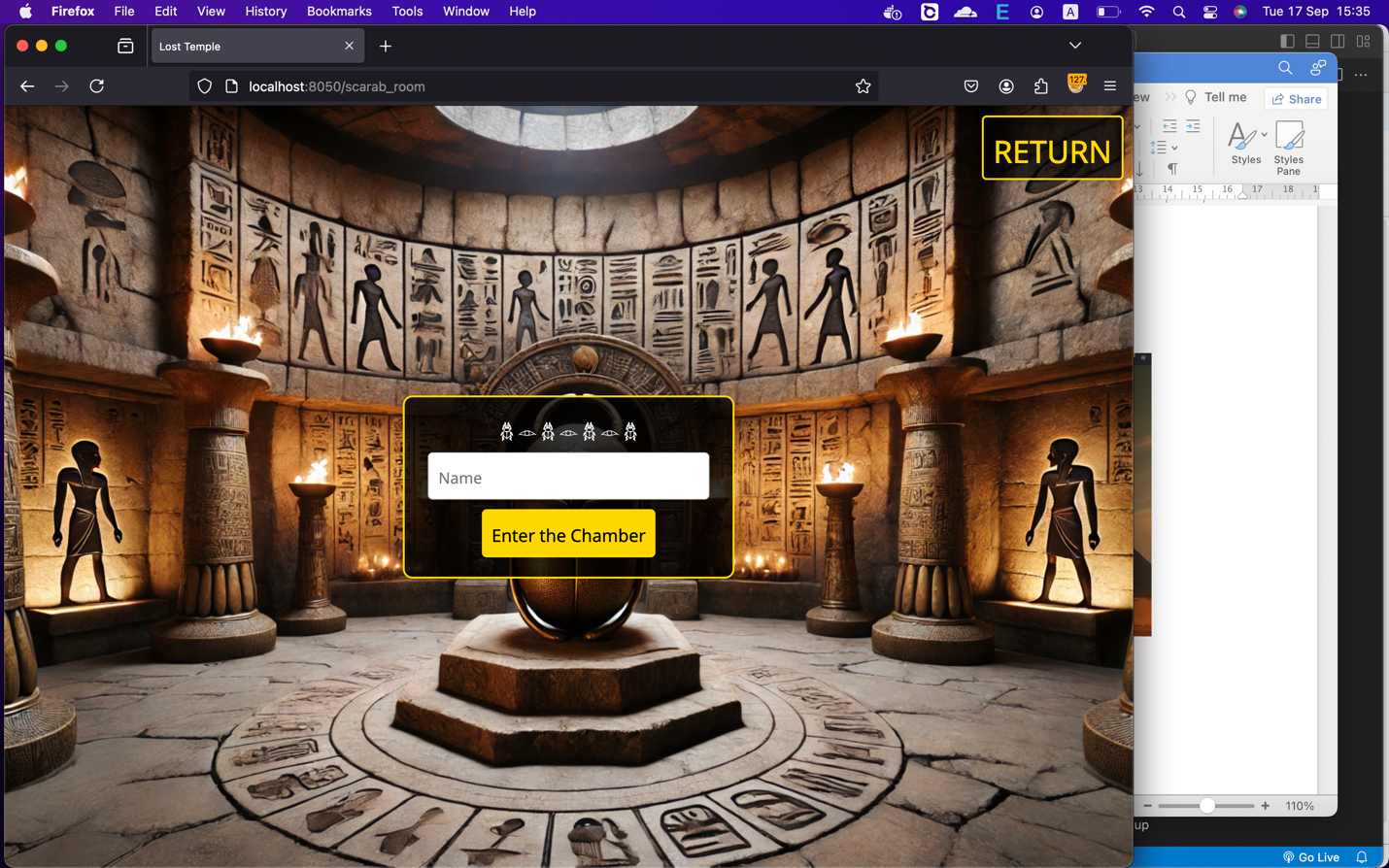
Nó giải mã cookie pyramid bằng publickey, nếu current\_date bằng KINGSDAY và role bằng royalty thì trả về trang kings\_lair.html.



Trang kings\_lair.html có flag

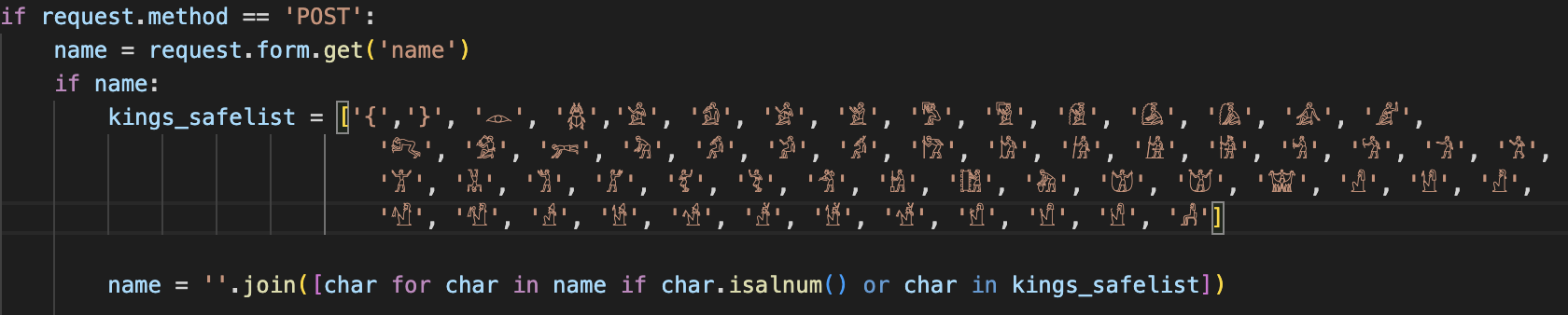


**/scarab\_room:**



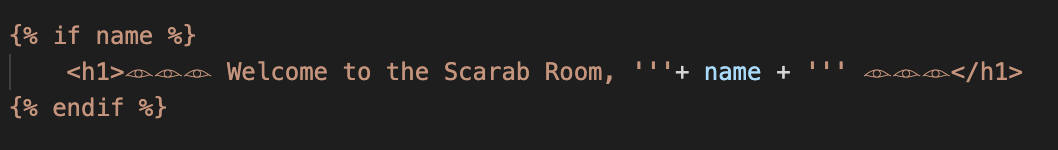
name=abc





Biến name được in trong render\_template\_string() nhưng bị lọc:

phải trong kings\_safelist và alphanumber



🡪 Lỗ hổng SSTI

**Web Exploit**

return render\_template\_string(''' … ''', name=name, \*\*globals())

Biến name truyền vào template là globals. Bạn có thể truyền vào template 1 biến nằm ngoài hàm scarab\_room() như PUBLICKEY và KINGSDAY.

Các kí tự đặc biệt bị lọc bởi máy chủ, tuy nhiên chúng ta vẫn còn { và }. PUBLICKEY và KINGSDAY có thể bị lộ bằng payload sau: {{PUBLICKEY}} và {{KINGSDAY}}

Public key: ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIPHEKCeUxejyNt6zHjmkiSuupQ82aSZm2IZlT0W5L04/

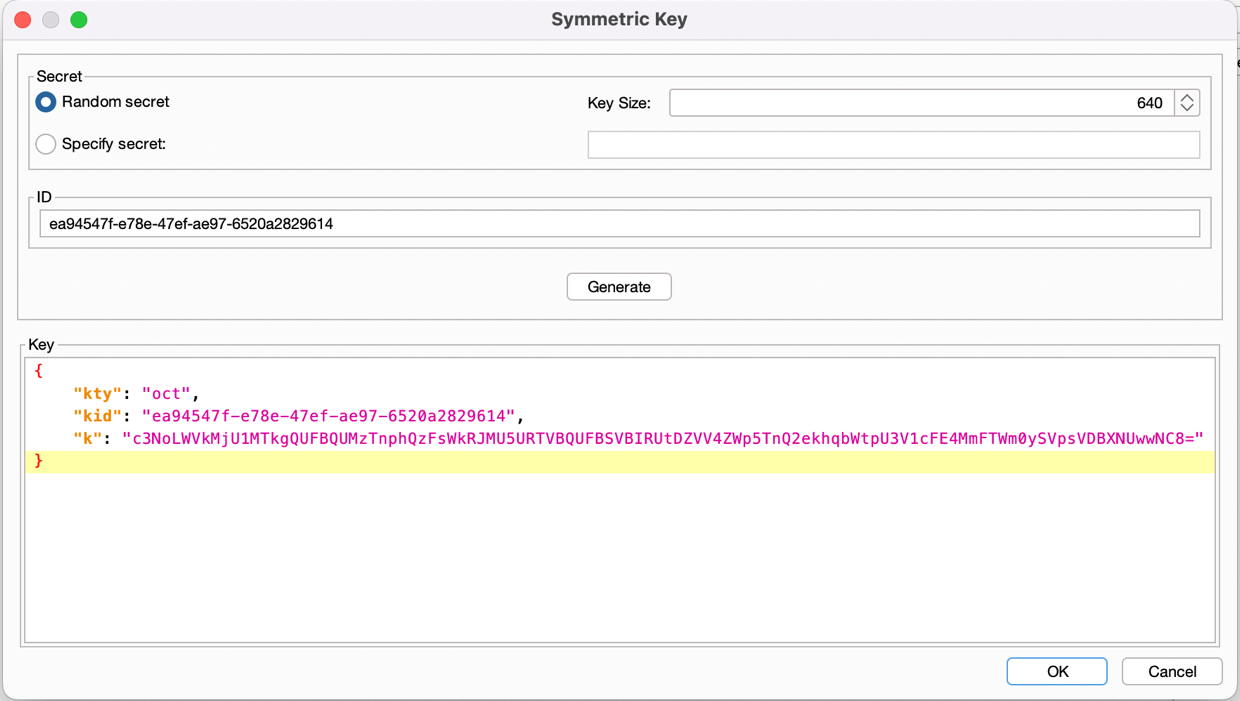
Kingsday: TESTINGDATE

Để lấy được flag, mình phải chỉnh sửa cookie pyramid

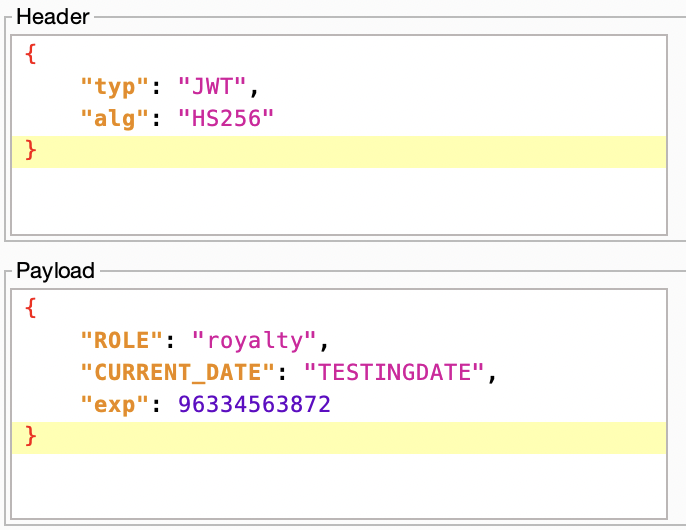
Khi trang web giải mã token, nó không chỉ định rõ thuật toán ban đầu là EdDSA mà cho 1 danh sách các thuật toán jwt.algorithms.get\_default\_algorithms(). Điều này cho phép kẻ tấn công giả mạo token bằng public key và chỉ định thuật toán đối xứng là HS256.

Làm sao dùng publickey để tạo 1 token hợp lệ?

Trong tab JWT Editor, tạo 1 symetric key ngẫu nhiên, sao chép public key (được mã hóa base64) vào tham số k:



Chỉnh sửa header và payload như sau:



Rồi kí bằng sysmetric key

**Flag: flag{TESTING\_FLAG}**

Lession We Learned:

* SSTI
* JWT Algorithm Confusion