

# Rotten Onion

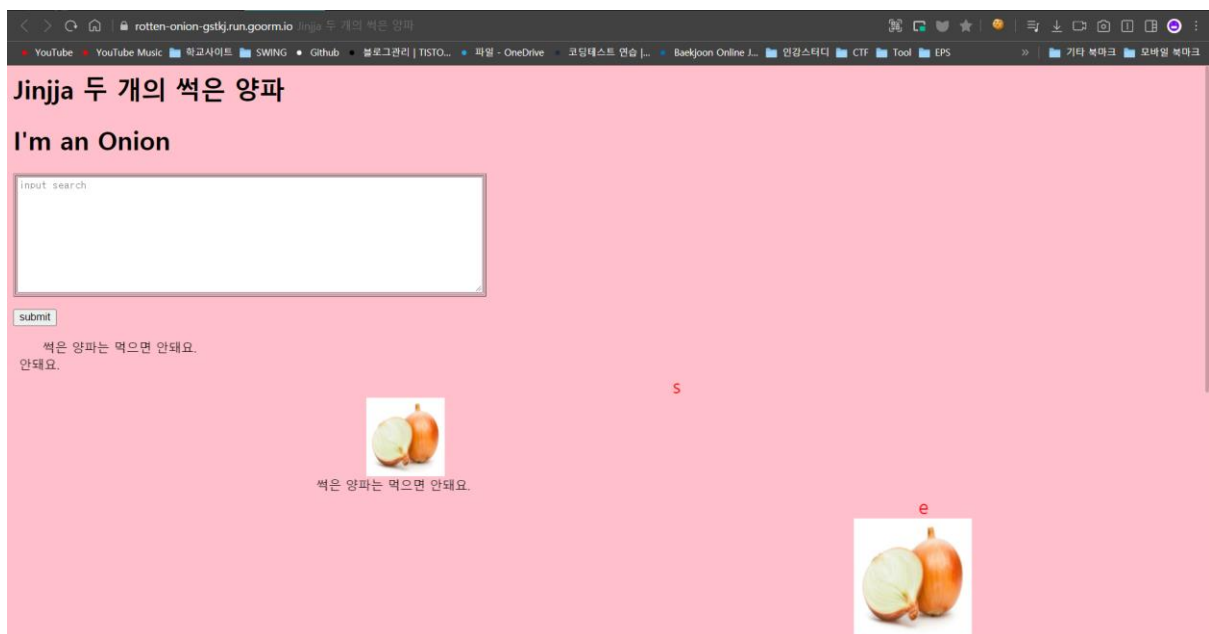
분야: Web, forensic

내용: 진짜진짜 양파다지기

## python의 flask, jinja2 엔진에서 사용할 수 있는 공격 코드

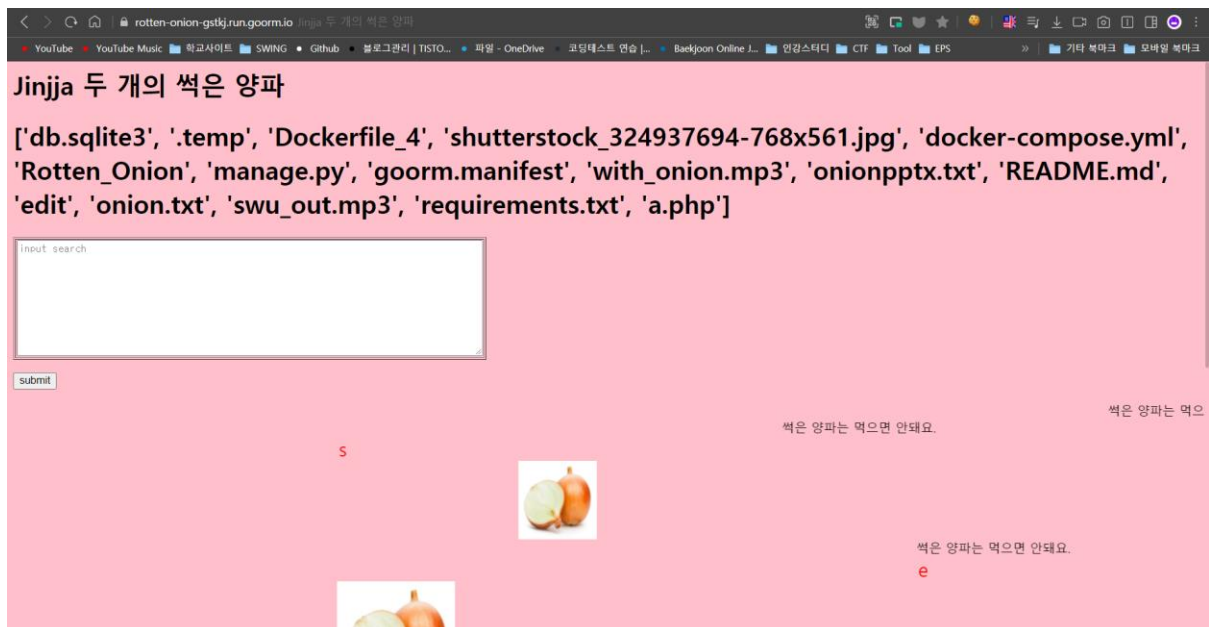
- `config` : 플라스크 앱 엔진 안의 변수 값들을 모두 가져온다.
- `self.__dict__` : jinja2 템플릿 엔진을 사용할 경우에 가능하며 템플릿 엔진의 값+ 플라스크 앱 엔진의 값들을 모두 가져온다.
- `url_for` : 플라스크 엔진에서 프로그램 전체 URL을 바꿀 때 사용하는 함수다. 그러나 변수처럼 사용할 경우 플라스크 엔진에 등록된 모든 함수 목록들을 볼 수 있다.
- `get flashed_messages` : 메시지 플래싱 기능이며 보통 필터링할 목록을 지정해서 보내는 기능을 가진다. 그러나 해당 클래스 안에 `_globals_` 변수가 있는데 해당 변수를 호출하면 해당 앱의 전역변수가 모두 나온다.

위 내용을 이용하여 SSTI 공격을 시도할 수 있다.



페이지에 들어가면 제목에서 힌트를 얻을 수 있다. jinja 두 개는 곧 jinja2를 의미한다. `{{7*7}}`을 입력하면 49라는 결과값이 출력되므로 jinja2 템플릿을 사용한 것이 맞다.

\*떠돌아다니는 빨간 글씨를 조합하면 search라는 단어가 나오는데 이는 search라는 변수에 GET방식으로 넘겨줄을 의미한다. 물론 텍스트창에서도 확인 가능하긴 하다.

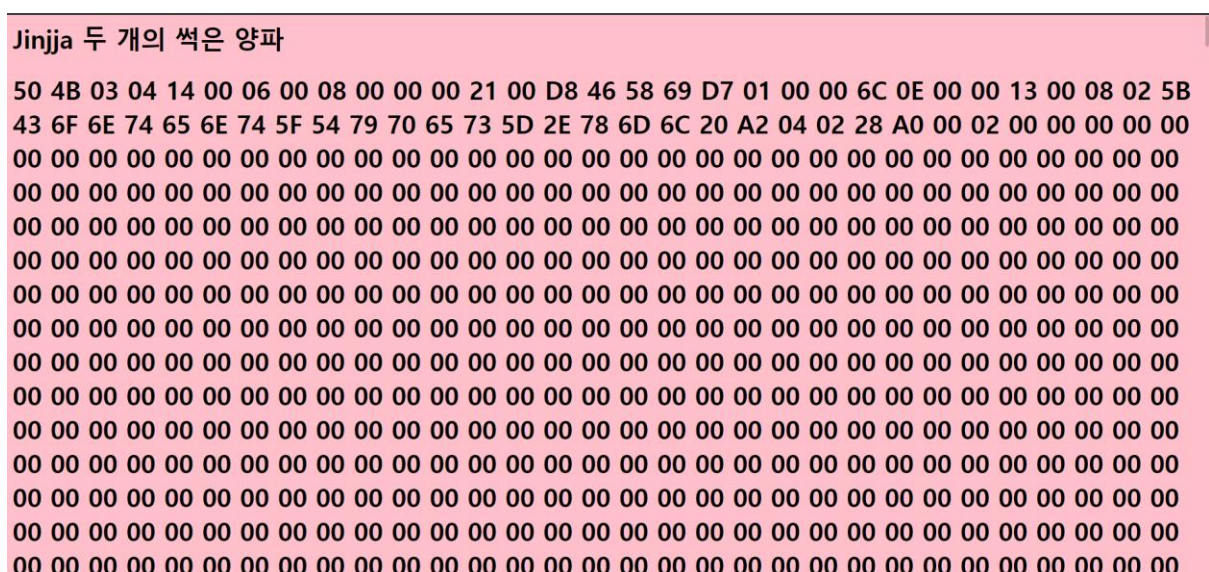


텍스트 영역에 아래 코드를 입력하면 문제 파일의 위치에 있는 모든 파일들을 볼 수 있다.

```
{{url_for.__globals__.__os__.__dict__.__listdir('./')}}}
```

onionpptx.txt파일과 onion.txt파일이 보인다.

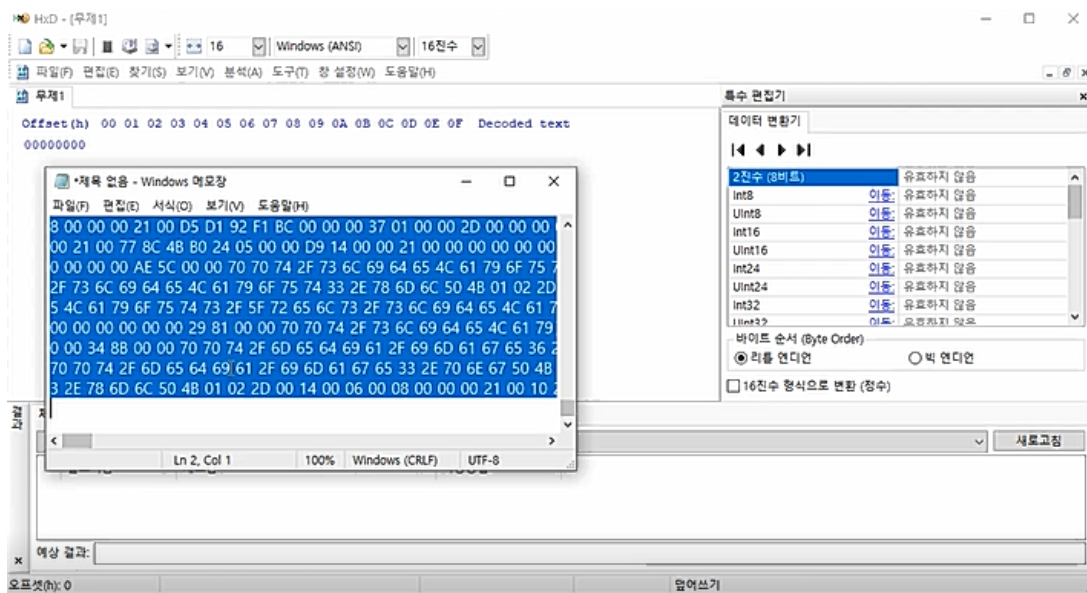
이제 둘 중 하나의 파일을 사용해 pptx파일을 얻으면 된다.



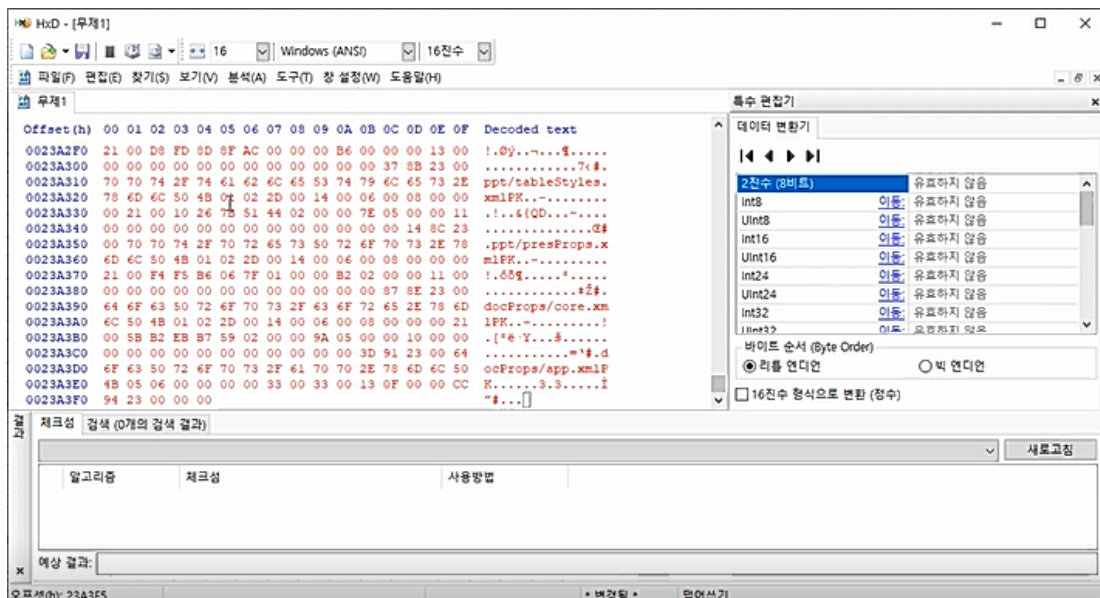
```
{{url_for.__globals__.__builtins__.__open__('onionpptx.txt').read()}}
```

이 명령어를 입력하면 onionpptx.txt파일의 내용 Hex 값으로 읽어올 수 있다.

위의 파일을 ctrl+a를 통해서 전체를 긁어온 다음에 메모장 등 편집할 수 있는 프로그램에 붙여넣어 앞 뒤의 한글 부분을 삭제한다



이를 hxd에 넣어준 후 파일을 새로 저장해주면 된다.



플래그를 구할 수 있는 pptx를 저장할 수 있다.

## Jinja 두 개의 썩은 양파

ppt로 가는 길... 오 여기까지 왔어? 좀만 더 힘내봐~~ <https://drive.google.com/uc?export=download&id=1ptBFgPo2LFzrGnjPu5xl16r9FIQ6cwxZ>

A screenshot of a web form. It features a large, empty rectangular input field with the placeholder text "input search" in the top-left corner. Below the input field is a small, rectangular button labeled "submit". The entire form is set against a light pink background.

썩은 양파는 먹!

두번째 방법이다.

```
{{url_for.__globals__.__builtins__.open('onion.txt').read()}}
```

이번에는 onion.txt파일을 열어보았다. 공유 드라이브 주소가 보이고 해당 주소로 이동하면 바로 pptx 파일을 다운 받을 수 있다.

Pptx 파일을 압축 해제(pptx -> zip으로 확장자 변경)하면 그 파일 안에 들어있는 그림을 볼 수 있다.

경로: WonionWpptWmedia


\* 참고로, ppt 내의 flag가 담긴 이미지는 ppt내에서 저장하면 원본 이미지가 아니라 새로운 이미지로 저장된다. 따라서 ppt 파일의 확장자를 zip으로 변경하여 ppt내의 원본 이미지에 접근해야 한다.

그 중 세번째 슬라이드의 ?(물음표)그림을 hxd로 열어 맨 밑을 보면 스테가노그래피 기법으로 넣어둔 fake flag가 들어있다.



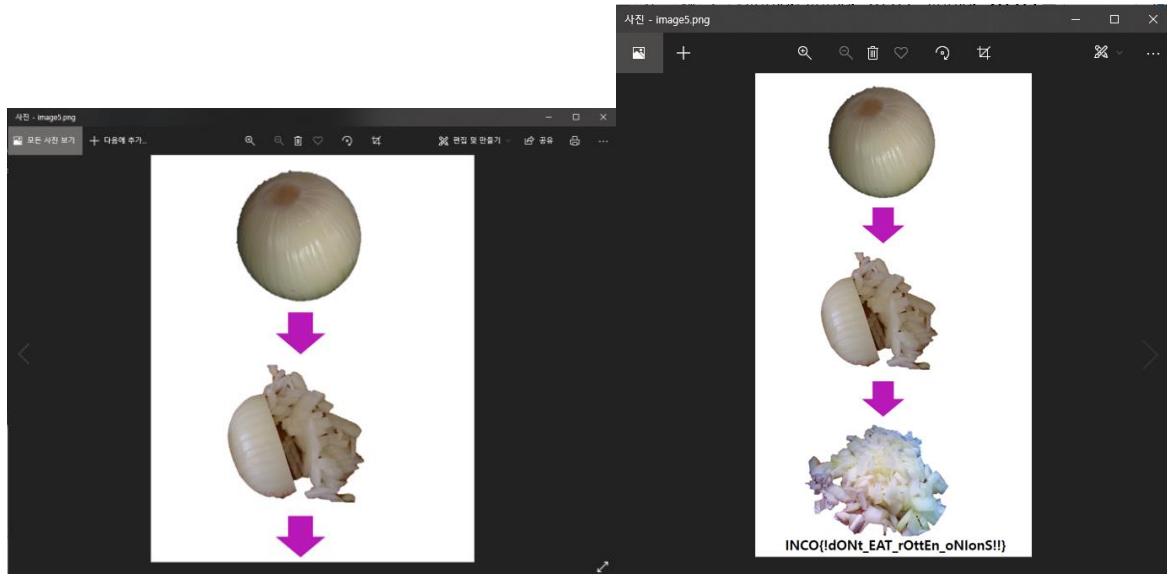


05정도로 수정해주면 flag가 포함된 전체 이미지를 확인할 수 있다.


image5.png

Offset(h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	Decoded text
00000000	89	50	4E	47	0D	0A	1A	0A	00	00	00	0D	49	48	44	52	%PNG.....IHDR
00000010	00	00	02	69	00	00	03	08	08	06	00	00	00	52	73	4B	...i.....RsK
00000020	BA	00	00	20	00	49	44	41	54	78	01	EC	9D	87	77	64	°.. .IDATx.i. #wd
00000030	57	99	ED	DF	BF	F3	1E	B8	73	54	4B	6A	65	A9	72	CE	W"iB¿ó.,sTKje@rî
00000040	39	87	7B	EB	56	54	95	72	6E	75	70	3B	1B	30	30	64	9#{ëVT•rnup;.00d

flag가 포함된 파일을 hxd로 열어서 hex 코드를 확인한 화면



세로 크기를 변경 전의 이미지

세로 크기를 변경 후의 이미지