

PASCALCTF



INTRODUZIONE



ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA



CYBERLOOP

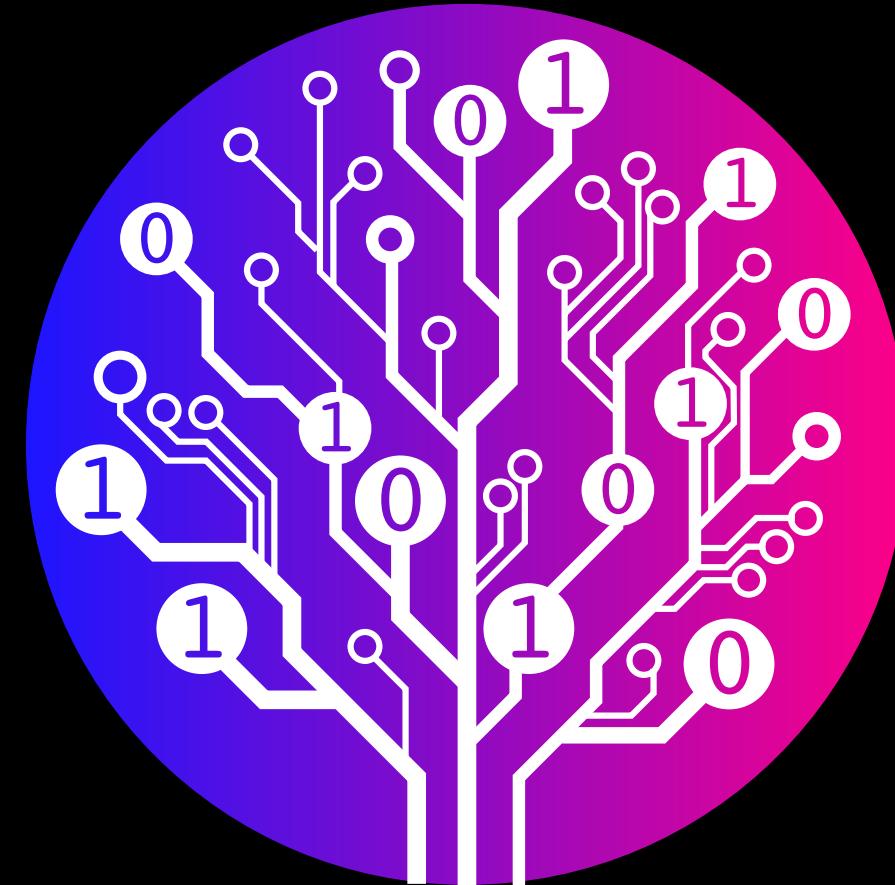
CATEGORIE AFFRONTATE

WEB

CRYPTO

MISC

REVERSE



PWN

Web Security

Protocollo HTTP

HTTP è il protocollo usato per il trasferimento di dati sul web. Nelle CTF, analizzare richieste e risposte HTTP aiuta a scoprire identificare molte delle vulnerabilità presenti .

HTML e Javascript

HTML struttura le pagine web, mentre JavaScript le rende dinamiche e interattive. Nelle CTF, analizzarli può rivelare commenti nascosti, input vulnerabili o attacchi XSS.

Cookie

I cookie sono piccoli dati memorizzati dal browser per mantenere sessioni e preferenze. Nelle CTF, possono essere "rubati" per ottenere la sessione di un utente privilegiato



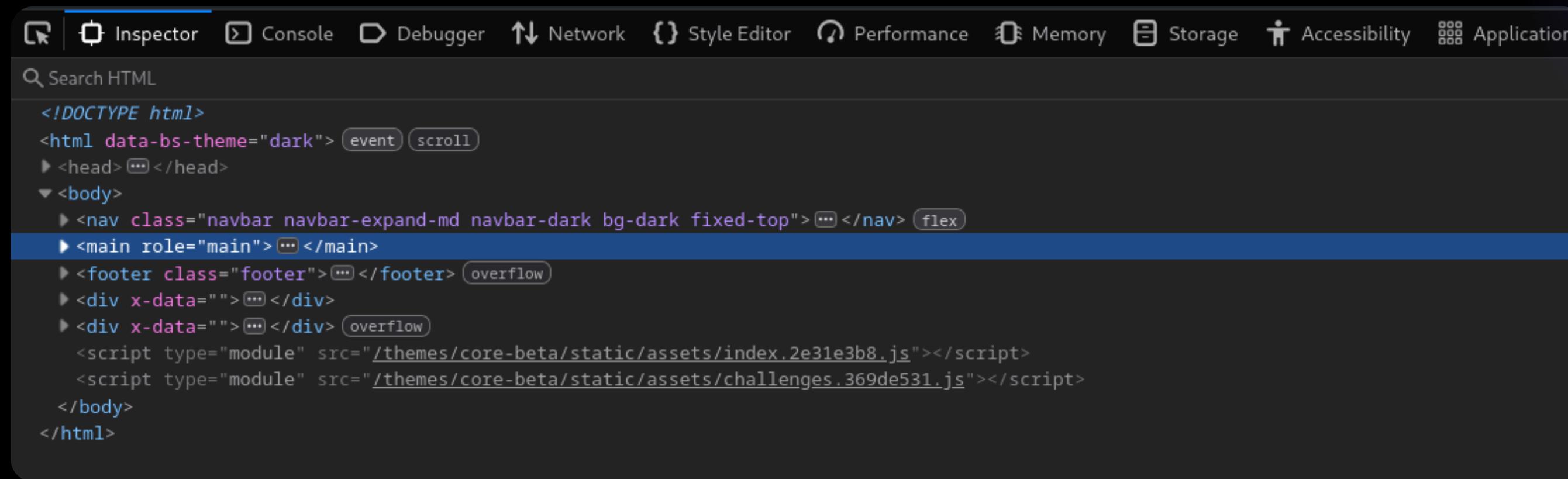
GET / HTTP/1.1
Host: localhost:4040
User-Agent: curl/8.11.1
Accept: */*



HTTP/1.0 200 OK
Server: SimpleHTTP/0.6 Python/3.13.1
Date: Wed, 29 Jan 2025 18:03:16 GMT
Content-type: text/html; charset=utf-8
Content-Length: 12

Ciao a tutti

Pannello sviluppatori di Firefox



The screenshot shows the Firefox Developer Tools Inspector panel. The top navigation bar includes tabs for Inspector, Console, Debugger, Network, Style Editor, Performance, Memory, Storage, Accessibility, and Application. The Inspector tab is active, indicated by a blue underline. Below the tabs is a search bar labeled "Search HTML". The main content area displays the HTML structure of a page. The root element is an `<!DOCTYPE html>`. It contains an `<html data-bs-theme="dark">` element with attributes `data-bs-theme="dark"`, followed by an `<event>` and an `<scroll>` element. The `<html>` element has two child nodes: a `<head>` node and a `<body>` node. The `<head>` node has no visible content. The `<body>` node contains several elements: a `<nav>` element with class `navbar navbar-expand-md navbar-dark bg-dark fixed-top`; a `<main role="main">` element with attribute `role="main"`; a `<footer>` element with class `footer`; and two `<div>` elements with attribute `x-data=""`. The `<main>` element is highlighted with a blue selection bar. The `<body>` and `<html>` elements have a dark gray background color. The rest of the page content is visible below the developer tools.

```
<!DOCTYPE html>
<html data-bs-theme="dark"> (event) (scroll)
  <head> (...)
  <body>
    <nav class="navbar navbar-expand-md navbar-dark bg-dark fixed-top"> (...) </nav> (flex)
    <main role="main"> (...) </main>
      <footer class="footer"> (...) </footer> (overflow)
      <div x-data=""> (...) </div>
      <div x-data=""> (...) </div> (overflow)
        <script type="module" src="/themes/core-beta/static/assets/index.2e31e3b8.js"></script>
        <script type="module" src="/themes/core-beta/static/assets/challenges.369de531.js"></script>
    </body>
</html>
```

SQL Injection

Le SQLi sono vulnerabilità dei database relazionali che permettono l'inserimento di codice SQL malevolo in input non filtrati, manipolando query per accedere o alterare dati.



LOGIC

UNION BASED

BLIND & TIME BASED

Logic SQL Injection

Questa categoria di SQL injection sfrutta l'inserimento dell'operatore booleano OR nelle query logiche per poter bypassare una condizione falsa.



```
SELECT * FROM utenti WHERE username='%s' AND password='%s'
```



```
SELECT * FROM utenti WHERE username='admin' AND password=' OR '1'='1'
```

Union based SQL Injection

L'operatore UNION in SQL permette di unire appunto i risultati di due query e viene utilizzato comunemente per poter esfiltrare dati del database come nome delle tabelle e delle loro colonne.



```
SELECT post FROM posts WHERE id='%s'
```



```
SELECT post FROM posts WHERE id='-1' UNION SELECT password FROM users WHERE username='admin'
```

Time based SQL Injection

Queste categorie sono più rare e si basano sul ritardare o far crashare la query SQL quando una determinata condizione si avvera.



```
SELECT post FROM posts WHERE id='%s'
```



```
SELECT post FROM posts WHERE id='0' AND IF(SUBSTRING(version(),1,1)=5,SLEEP(10),null)
```

Crittografia

La crittografia è una tecnica di trasformazione dei dati mediante algoritmi matematici e chiavi crittografiche per garantire riservatezza, integrità e autenticazione.

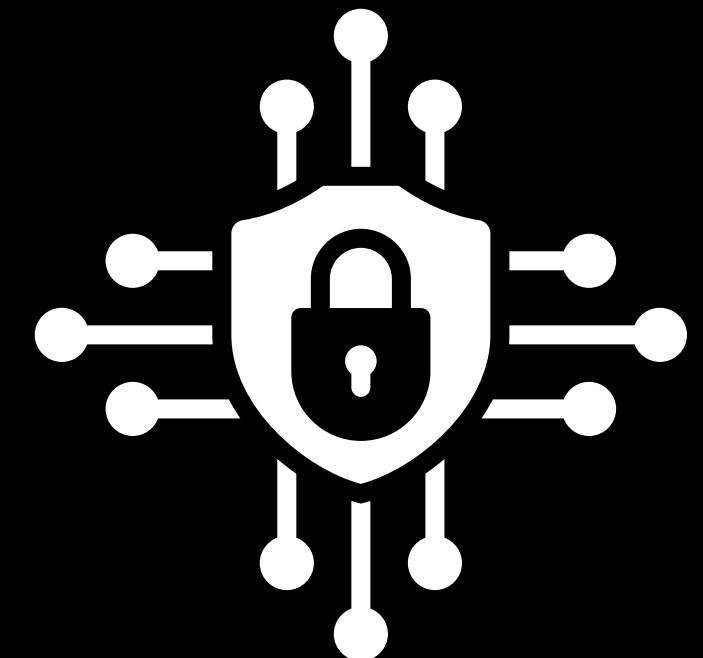
Si divide in **simmetrica** e **asimmetrica**.

Simmetrica

La crittografia simmetrica usa una singola chiave per cifrare e decifrare i dati (es. **AES**, **ChaCha20**). Questo metodo è veloce ed efficiente, ma richiede una gestione sicura delle chiavi.

Asimmetrica

La crittografia asimmetrica utilizza una coppia di chiavi (**pubblica** e **privata**) per, rispettivamente, cifrare e decifrare (es. **RSA**, **ECC**). Questo metodo garantisce autenticazione e scambio sicuro di chiavi, ma è più lenta rispetto alla simmetrica.

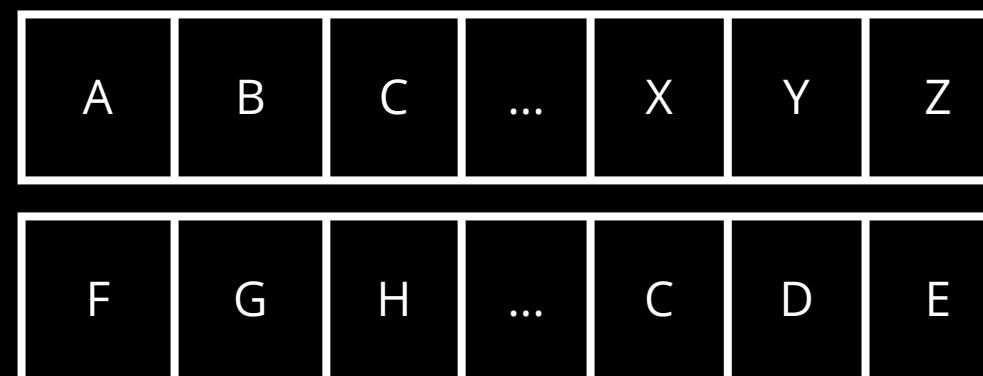


Metodi di cifratura

01 Cifrari a sostituzione

I cifrari a sostituzione consistono nella cifratura un messaggio passando da un alfabeto **A** ad un alfabeto **B** mediante una correlazione univoca tra i due.

<https://dcode.fr/>



02 Operazioni Bit a Bit

Questo metodo è spesso non sicuro e viene usato perlopiù nelle CTF per sviluppare un ragionamento matematico, sfrutta operatori come **AND**, **OR** e **XOR**.

03 RSA e Curve Ellittiche

Sono metodi di cifratura assimmetrici e prevedono l'utilizzo di una chiave pubblica (**N, e**) ed una privata (**N, d**). Richiedono conoscenze di algebra modulare.

Miscellaneous

Comprende tutte quelle challenge che non appartengono alle altre 4 categorie.

Sottocategorie

NETWORK

Sfide basate sull'analisi del traffico di rete. Spesso coinvolgono file PCAP e vulnerabilità nei protocolli di comunicazione.

STEGANOGRAFIA

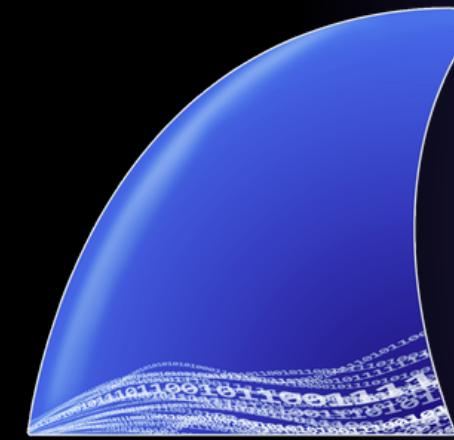
Tecnica per nascondere informazioni all'interno di file come immagini, audio o testi. Le sfide possono richiedere l'uso di tool per estrarre i dati nascosti.

OSINT

Sfide basate sulla raccolta di informazioni da fonti pubbliche. Richiedono abilità di ricerca e conoscenza di tool adatti.

Network

Nelle **CTF**, la categoria *Network Security* riguarda l'analisi del traffico di rete per estrarre informazioni sensibili, credenziali o flag. Le sfide possono includere **file PCAP** (Packet Capture) contenenti **comunicazioni registrate**, **richieste HTTP** e **richieste DNS**.



Wireshark

Wireshark è un tool che permette di analizzare pacchetti in tempo reale o da file **PCAP**. Consente di filtrare il traffico e di analizzarlo.

Steganografia

01 Cos'è?

È una tecnica per nascondere informazioni all'interno di immagini, audio e file in generale.

02 Come estraprolo i dati nascosti?

Attraverso diversi tool, quello migliore (perchè ne comprende molti) è **aperisolve**.

03 Come si usa aperisolve?

Aperisolve è accessibile da <https://aperisolve.com>, da qui è soltanto necessario uploadare il file da analizzare.

What is this ?

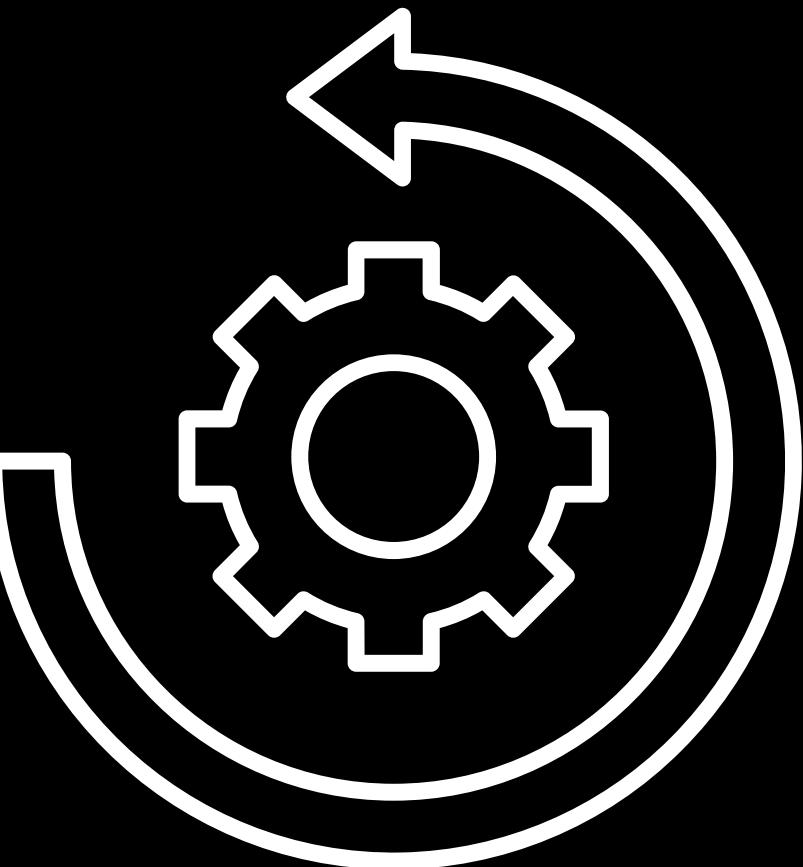
Aperi'Solve is an online platform which performs layer analysis on image. The platform also uses zsteg, steghide, outguess, exiftool, binwalk, foremost and strings for deeper steganography analysis. The platform supports the following images format: .png, .jpg, .gif, .bmp, .jpeg, .jfif, .jpe, .tiff...

The screenshot shows a dark-themed web application. At the top, a green header bar contains the text "What is this ?". Below it, a large black rectangular area is labeled "Select a file or drag here". To the right of this area is a small icon of a downward arrow with three dots. At the bottom of this area is a green button with the text "PLEASE SELECT AN IMAGE". At the very bottom of the page is a wide green footer bar with a white "SUBMIT" button in the center.

REVERSE ENGINEERING

01 In cosa consiste?

Analizzare software in cerca di vulnerabilità sfruttabili.



02 Quali strumenti posso usare?

Debugger, disassembler e decompiler.

IDA

Decompiler elogiato per la sua capacità di decomplazione in c.

GHIDRA

Decompiler più preciso di IDA nella decompilazione, allo stesso tempo è però “complesso” e meno intuitivo.

GDB

Debugger che permette di analizzare a runtime l'eseguibile ed eseguire passo passo il programma.

Struttura di un file binario



Al momento della sua esecuzione, un **ELF** è diviso in vari segmenti:

.text

Codice dell'eseguibile

.rodata

Variabili globali in sola lettura

.data

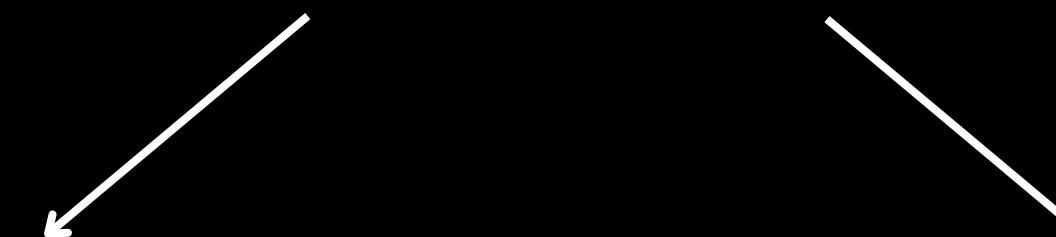
Variabili inizializzate

.bss

Variabili globali

Ma dove sono le variabili locali e dinamiche?

Area di **Stack**



Area di **Heap**

Nella pratica ...

Bypass controllo licenza

Il programma a lato non è sicuro. La string "licenza" è salvata nel segmento .rodata, perciò con un decompilatore possiamo trovarla e utilizzare la licenza senza pagare.



reversible.c

```
#include <stdio.h>
#include <string.h>

int main() {
    char input[14];
    const char *licenza = "licenzasegreta"; // licenza >:)
    printf("Inserisci la licenza per usare il mio programma \n> ");
    scanf("%14s", input);

    int result = strcmp(input, licenza);

    if (result == 0) {
        printf("Grazie per aver acquistato la licenza!\n");
    } else {
        printf("Torna quando mi avrai pagato!\n");
    }

    return 0;
}
```

Visto da IDA

The screenshot shows the IDA View-A window with the title "IDA View-A". The window displays the following pseudocode for the main function:

```
1 int __fastcall main(int argc, const char **argv, const char **envp)
2 {
3     char s1[14]; // [rsp+1Ah] [rbp-16h] BYREF
4     unsigned __int64 v5; // [rsp+28h] [rbp-8h]
5
6     v5 = _readfsqword(0x28u);
7     printf("Inserisci la licenza per usare il mio programma \n> ");
8     _isoc99_scanf("%14s", s1);
9     if ( !strcmp(s1, "licenzasegreta") )
10        puts("Grazie per aver acquistato la licenza!");
11    else
12        puts("Torna quando mi avrai pagato!");
13    return 0;
14 }
```

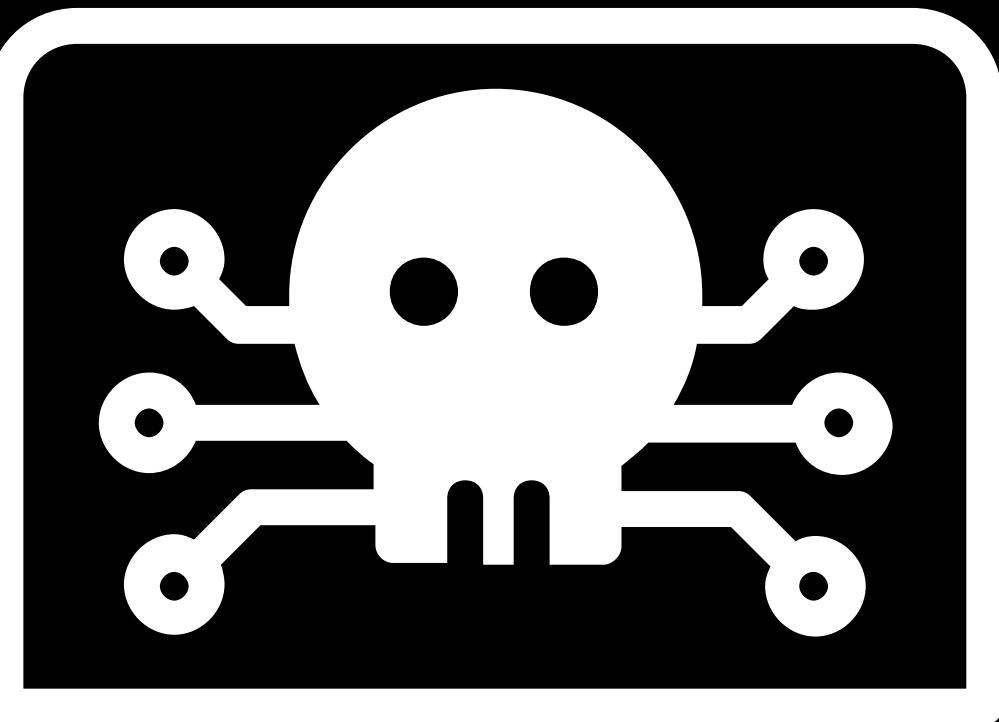
The screenshot shows the IDA View-A window with the title "IDA View-A" and the tab "Pseudocode-A" selected. Below the pseudocode, the raw assembly code for the .rodata section is displayed:

```
.rodata:0000000000002007          db  0
.rodata:0000000000002008 aLizenzasegreta db 'licenzasegreta',0 ; DATA XREF: main+1B↑o
.rodata:0000000000002017          align 8
.rodata:0000000000002018 ; const char format[]
.rodata:0000000000002018 format      db 'Inserisci la licenza per usare il mio programma ',0Ah
.rodata:0000000000002018           ; DATA XREF: main+26↑o
```

PWN

01 In cosa consiste?

Manipolare il comportamento di un'applicazione.



02 Quali tecniche devo conoscere?

Buffer overflow, ret2win e format string attack.

03 Cosa affronteremo?

Struttura stack

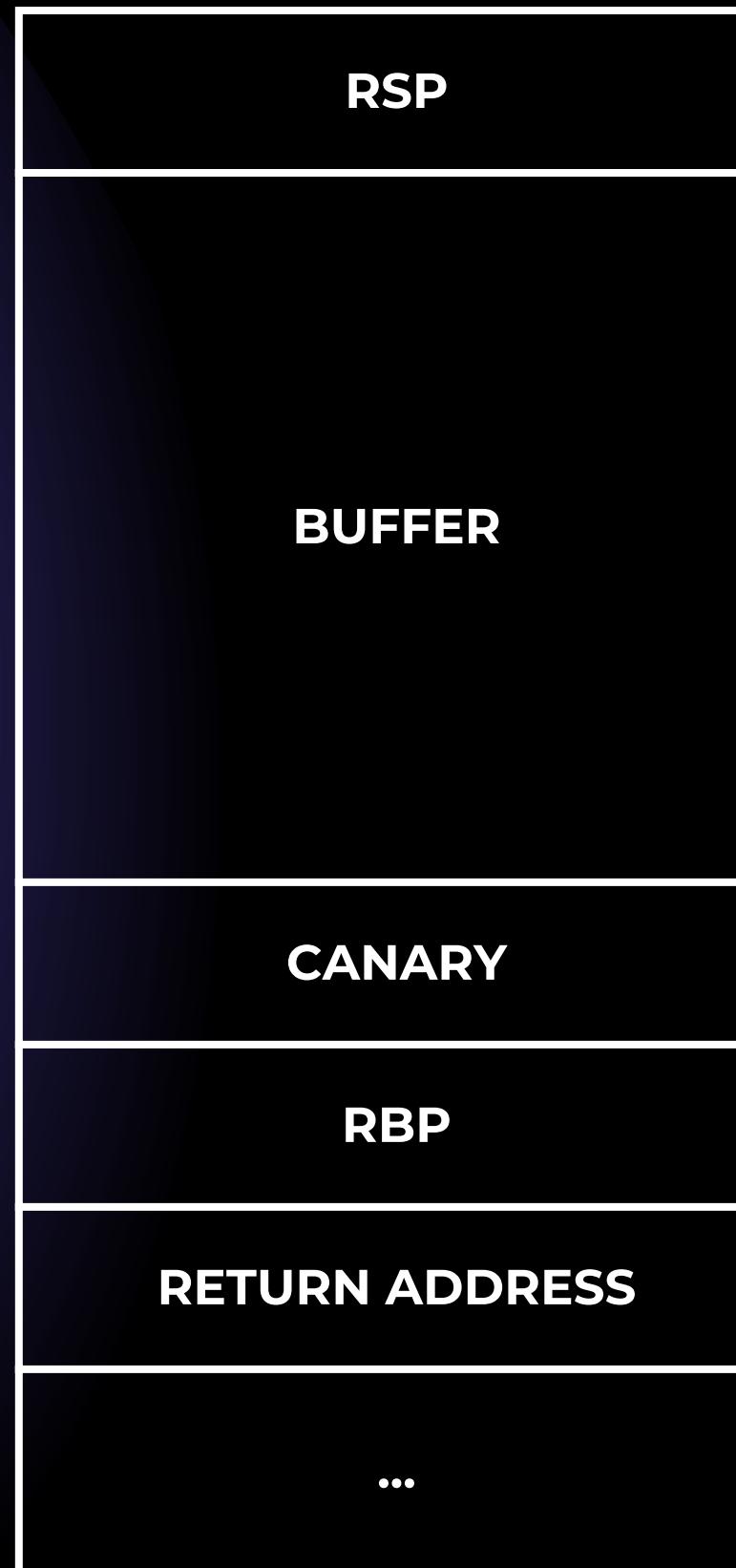
Come vengono salvate le variabili locali.

Vulnerabilità

Come individuare elementi sfruttabili.

Pwntools

Come automatizzare la comunicazione con i servizi locali e remoti.



Stack frame x86_64

Al momento della chiamata di una funzione, sullo stack viene costruito uno stack frame così composto:

RSP

Punta al top dello stack

BUFFER

Spazio in cui vengono salvate le variabili locali

CANARY

Stack protection

RBP

Punta alla base dello stack frame corrente

RETURN ADDRESS

Dove deve ritornare la funzione dopo una ret

E sotto?

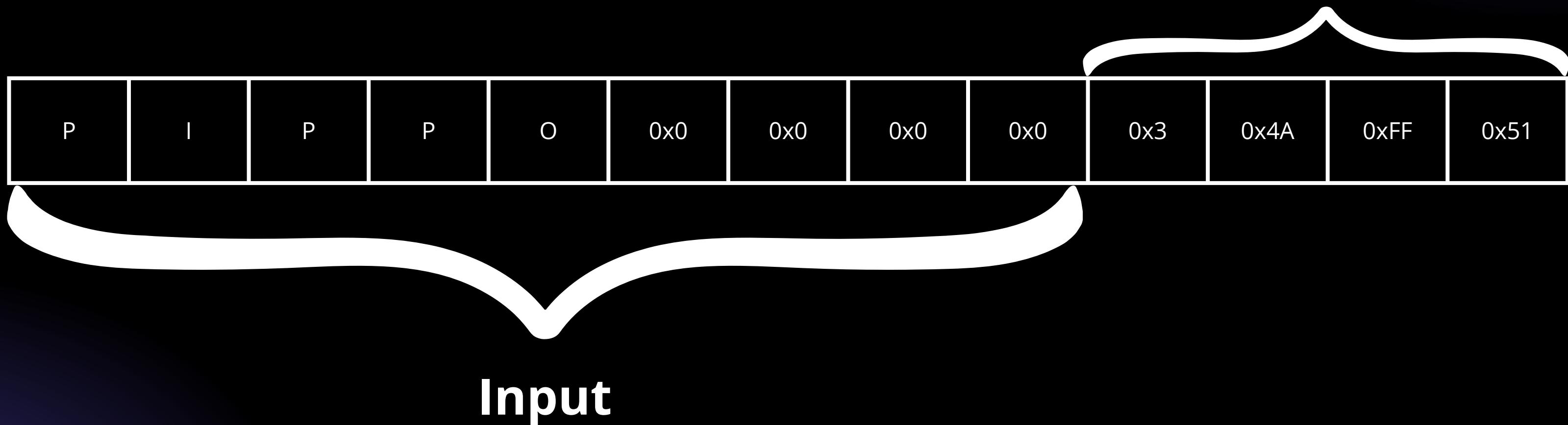
Ci sono gli stack frame precedenti.

Buffer Overflow

01 In cosa consiste?

Fuoriuscire dal buffer in cui viene scritto il nostro input.

Variabile da sovrascrivere



Buffer Overflow

02 Come faccio ad individuarlo?

Il limite di caratteri inseribili nel buffer è maggiore della dimensione del buffer stesso.



reversible.c

```
#include <stdio.h>

int main(){
    char buffer[40];
    int numero_preferito = 0x00069420;
    printf("Ciao, come ti chiami?\n > ");
    scanf("%100s", buffer);

    printf("Ciao %s, il mio numero preferito è %x.\n", buffer, numero_preferito);
}
```

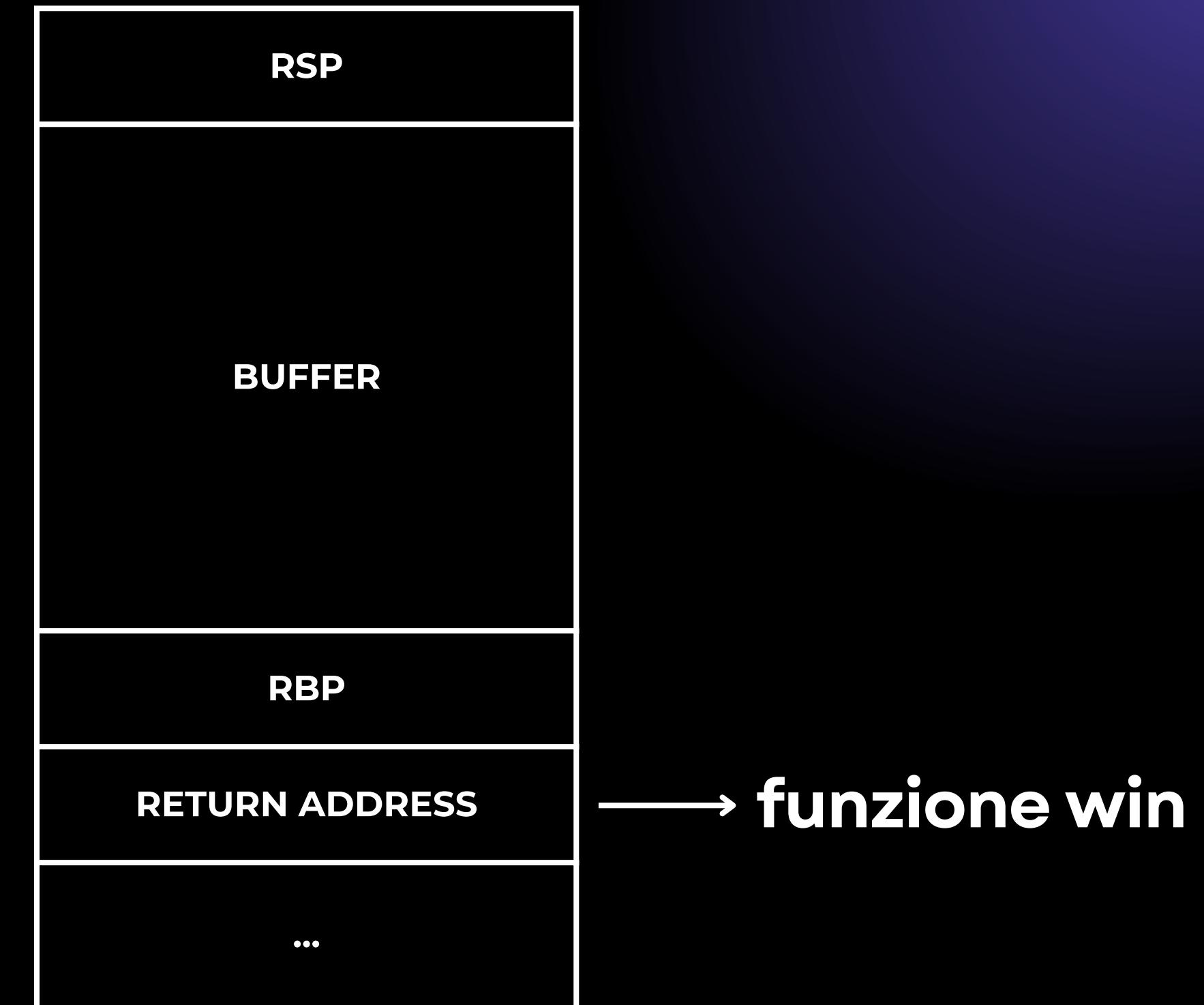
Ret2Win

03 Come funziona?

Sovrascrivo il valore del return address nello stack per eseguire il codice che voglio.

04 È così facile?

SI



Ret2Win



ret2win.c

```
#include <stdio.h>

int saluta() {

    char buffer[24];

    printf("Ciao, come ti chiami?\n > ");
    scanf("%60s", buffer);

    printf("Grazie %s per aver usato il mio programma!\n", buffer);
    return 0;
}

int main(){

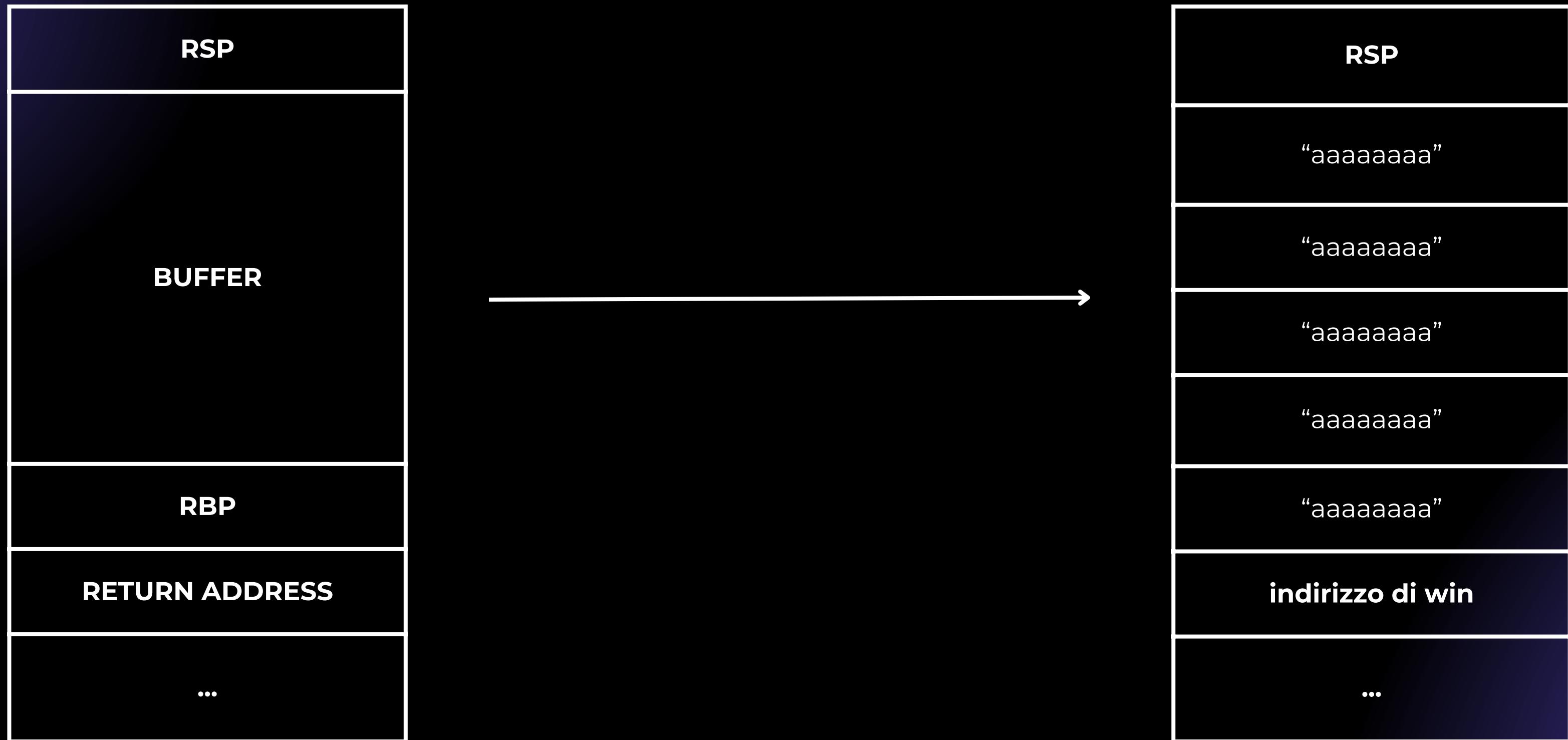
    saluta();
    printf("Se stai leggendo questo allora non sei un surgo!!1!");
}
```



```
int funzione_da_non_eseguire(){

    printf("No0oooo0o00!!1!, questa funzione non doveva essere eseguita!!!");
    return 0;
}
```

Ret2Win



Format string attack

01 In cosa consiste?

Inserire formattatori in una string che verrà stampata con una printf.

02 Cosa possiamo fare?

Leggere/scrivere arbitrariamente valori sullo stack e nei vari segmenti.

03 Come lo individuo?

Viene passato come argomento della printf direttamente il nostro buffer senza formattazioni.

Format string attack



printf.c

```
#include <stdio.h>
#include <stdint.h>

int64_t calcola_numero_segreto() {
    return 0x0000000000060000LL ^ 0x0000000000009420LL;
}

int main() {
    int64_t numero_super_segreto = calcola_numero_segreto();
    char buffer[40];

    printf("Inserisci qualcosa, tanto non mi interessa.\n > ");
    scanf("%40s", buffer);

    printf("Hai detto:\n -> ");
    printf(buffer);

    printf("\nBasta, ora me ne vado.\n");

    return 0;
}
```

Format string attack

04 Formattatori

Sequenza di caratteri speciali preceduta da %, utilizzata per specificare il tipo e il formato dei dati da stampare

```
Inserisci qualcosa, tanto non mi interessa.  
> %7$p  
Hai detto:  
-> 0x69420  
Basta, ora me ne vado.
```

%d	interi a 4 byte
%lld	interi a 8 byte
%f	float
%X	interi in hex
%p	parametro
%s	string
%C	char

Pwntools

01 Cos'è?

Una libreria di python utile per comunicare con servizi locali e remoti.

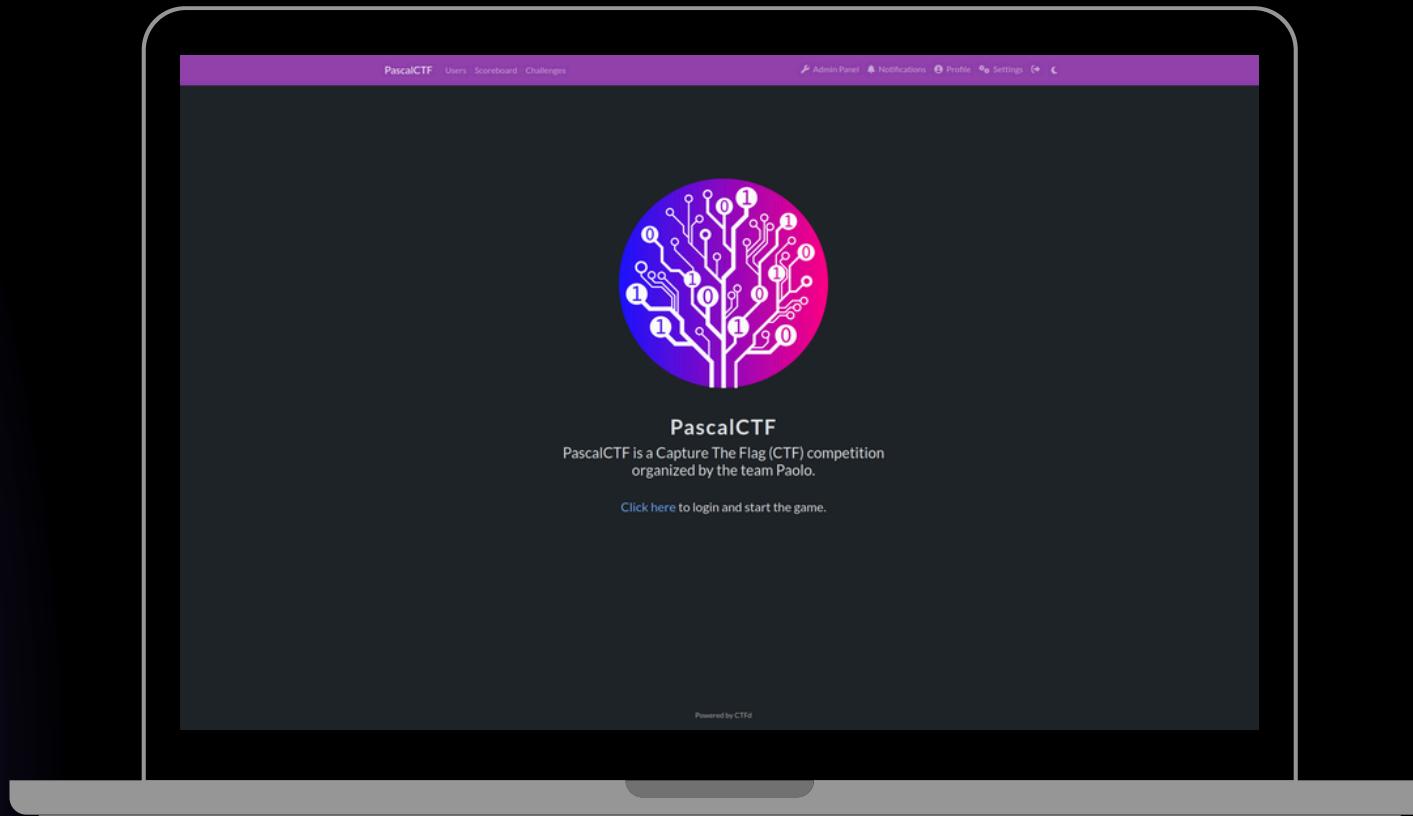
```
from pwn import *

if args.REMOTE:
    r = remote(host="servizio_remoto", port=1234)
else:
    r = process("precorso_eseguibile")

r.recvline()      #ricevo fino ad un a capo (\n)
r.recv(10)        #ricevo n bytes
r.recvuntil(b'>') #ricevo finchè non leggo un carattere '>'

r.sendline(b'ciao') #invio il contenuto seguito da (\n)
r.send(b'ciao')     #invio il contenuto senza aggiungere \n

r.interactive()    #prendo il controllo della connessione
```



01 CHALLENGE

Il portale mostrerà tutte le challenge in ordine di difficoltà e divise per categorie.

02 SCOREBOARD

La scoreboard sarà aperta a tutti i team e si potrà consultare per poter capire quali siano le challenge più risolte (probabile anche le più facili).

Portale di gara

Il portale di gara si trova all'URL

<https://ctf.pascalctf.it/>

Se avete domande o perplessità non esitate ad alzare la mano e qualcuno dello staff provvederà ad aiutarvi.

BUONA FORTUNA

La gara inizierà alle 14 e si concluderà alle 17

<https://ctf.pascalctf.it/>