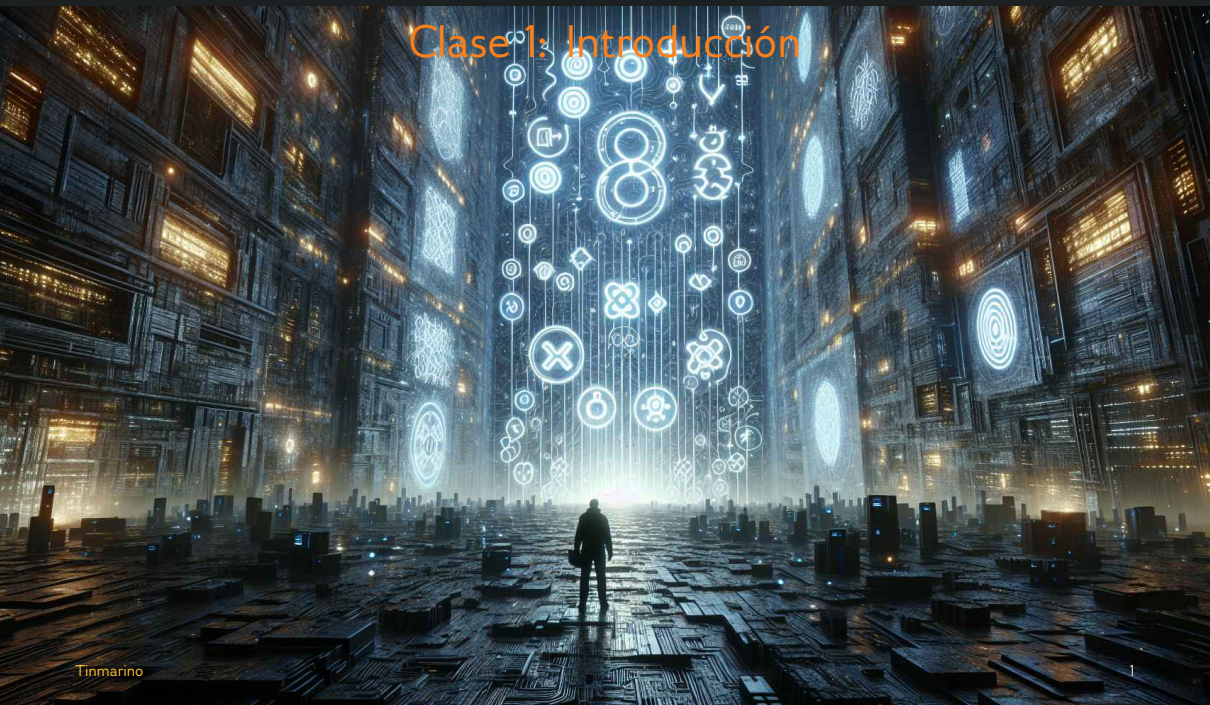
# Pentest Web

## Clase 1: Introducción



# Clase 1: Introducción

| M | Nombre | Descripción |
|---|---|---|
| 1 | Contexto | Plan docente del curso |
| 2 | Ciberseguridad | Paisaje de la ciberseguridad |
| 3 | Web | Tecnologias de aplicaciones web |
| 4 | Hacktitud | Estado de espíritu |

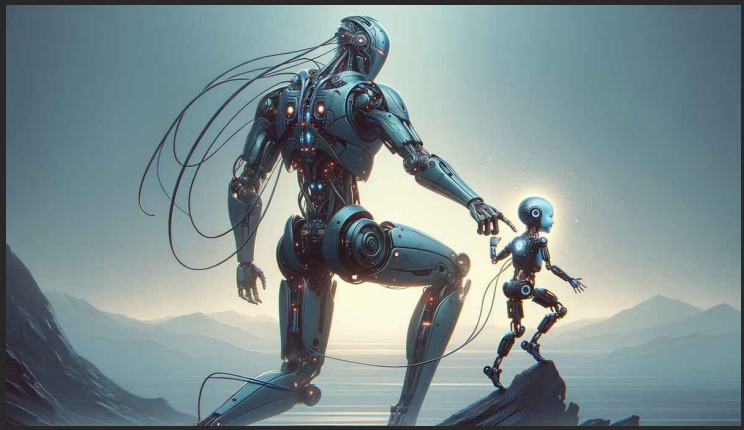
# Módulo 1: Contexto

# Módulo 1: Contexto

| S | Nombre | Descripción |
|---|---|---|
| 1 | Curso | Cómo se inscribe el curso en la serie |
| 2 | Syllabus | Plan docente del curso de pentest web |
| 3 | Pentest web | Especificidades del pentest web |
| 4 | Demo | Demostraciones de pentest de aplicaciones web |

# Sesión 1: Curso

## (Módulo 1: Contexto)



# Este curso

*Herramientas y explotación de vulnerabilidades **web**.*

# Aprendizaje

1. **Distinguir las técnicas y vulnerabilidades** más explotadas por los atacantes cibernéticos en el proceso de explotación de sistemas y sitios web.
2. **Identificar las herramientas** más utilizadas por los pentesters, para simular ataques reales a los sistemas informáticos.
3. **Analizar el proceso de explotación** paso a paso de las vulnerabilidades más comunes de OWASP y aplicándolo en plataformas controladas, como CTF.

# Estrategia metodológica

Estrategia de **enseñanza activa**.

- **Clases** expositivas con apoyo de material audiovisual (PDF)
- **Ejercicios** guiados (CTF)
- **Pruebas** de penetración a entornos controlados (Lab)

Enfocada en la metodología, cuasi agnóstica de la tecnología.

# Estrategia evaluativa

- **Control individual**: Evaluación teórica escrita que cubre los conceptos básicos y aplicados del curso. Ponderación: **40%** de la nota final.
- **Trabajo práctico grupal**: Desarrollo de un proyecto de pruebas de penetración en un entorno controlado, aplicando las herramientas y técnicas aprendidas, con un informe grupal detallado que incluya análisis y recomendaciones. Ponderación: **60%** de la nota final.

# Enlaces

| N. | Nombre | Descripción | Enlace |
|---|---|---|---|
| 1 | CTFd | CTF | http://ctf.tinmarino.com # «Uc1Uc2Uc3!!» |
| 2 | UC | Preguntas | https://ep.ingenieriauc.cl/course/view.php?id=5540 |
| 3 | BancoPenca | Examen final | http://bancopenca-XXXXXXXX.tinmarino.com |
| 4 | PortSwigger | Lab | https://portswigger.net/web-security/all-labs |

# Sesión 3: Plan docente

## (Módulo 1: Contexto)

# Estructura

| H. | N. | Tipo | Ejemplo | Tiempo |
|---|---|---|---|---|
| h0 | 1 | Diplomado | Pentest | 96 horas |
| h1 | 4 | Curso | Pentest web | 24 horas |
| h2 | 8 | Clase | Introducción | 3 horas |
| h3 | 4 | Módulo | Contexto | 30 minutos |
| h4 | 4 | Sesión | Plan docente | 10 minutos |

# Veni

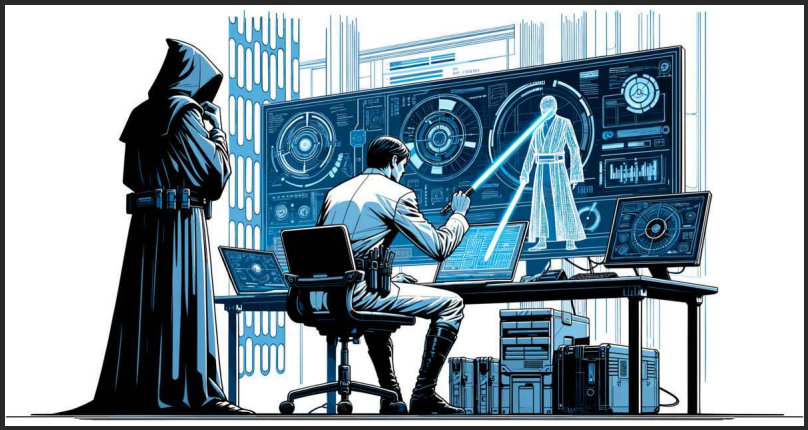
| N. | Clase | M1 | M2 | M3 | M4 |
|---|---|---|---|---|---|
| 1 | **Introducción** | **Contexto** | **Ciberseguridad** | **HTTP** | **Hacktitud** |
| | | Clase | Definición | Tecnología | Espíritus |
| | | Syllabus | Terreno | HTTP | Reglas |
| | | Pentest web | Ciberataque | TLS | Escuela |
| | | Demo | Paisaje | Extra | Trucos |
| 2 | **Reconocimiento** | **Subfinder** | **Nmap** | **FFuF** | **BurpSuite** |
| | | Descripción | Equipos | Rutas | Proxy |
| | | Fuentes | Puertos | Filtros | Escáner |
| | | Avanzado | Configuración | Parámetros | Herramientas |
| | | Integración | Servicios | Trucos | Extensiones |

# Vidi, Vici, Dixi

| N. | Clase | M1 | M2 | M3 | M4 |
|---|---|---|---|---|---|
| 3 | **Acceso** | Fundamentos | Criptografía | Tecnología | IDOR |
| 4 | **Incursión** | CVSS | Divulgación | Listas | Otros |
| 5 | **Lógica** | Negocio | Flujo | Aritmética | Diseño |
| 6 | **Inyección** | SQL | OS | Código | Parámetros |
| 7 | **informe** | Equipos | Objetivo | Metodología | Reporte |
| 8 | **Conclusión** | Resumen | Reflexiones | CVE | Futuro |

# Sesión 2: Pentest web

## (Módulo 1: Contexto)



# Objetivos del pentest web

1. **Identificar** vulnerabilidades.
2. **Evaluar** el impacto.
3. **Recomendar** mitigaciones.
4. **Mejorar** la seguridad.

# Ejemplos de pentest

| | | | | |
|---|---|---|---|---|
| Red | **Web App** | Móvil | API | Emulación |
| IoT | Nube | Código fuente | Físico | Empleados |
| Huéspedes | Reversing | Políticas de seguridad | Firewalls | IDS/IPS |
| Servidores | Externo | Interno | Suministros | Bases de datos |

# Herramientas web

| N. | Herramienta | Uso |
|---|---|---|
| 1 | Firefox | Navegador web |
| 2 | BurpSuite | Proxy HTTP |
| 3 | cURL | Cliente URL |
| 4 | Nmap | Escáner de puertos |
| 5 | Nuclei | Escáner de vulnerabilidades |
| 6 | FFuF | Fuzeador de rutas HTTP |
| 7 | Subfinder | Buscador de subdominios |
| 8 | SQLmap | Detector de inyecciones SQL |

# Herramientas locales

| N. | Herramienta | Uso |
|---|---|---|
| 1 | Bash | Lenguajes de scripting o Powershell, Python o Perl |
| 2 | OpenSSH | Herramienta de conexión SSH |
| 3 | Tmux | Multiplexor de terminal |
| 4 | Netcat | Herramienta de red |
| 5 | OpenVpn | Cliente VPN |
| 6 | RipGrep | Herramienta de búsqueda |
| 7 | Vim o VSCode | Editores de texto |
| 8 | Git | Sistema de control de versiones |
| 9 | Pandoc | Conversor de documentos |

# Herramientas sitios

| N. | Herramienta | Uso |
|---|---|---|
| 1 | Google | Motor de búsqueda |
| 2 | Virustotal | Análisis de archivos y URLs |
| 3 | Shodan o Censys | Motores de búsqueda de dispositivos |
| 4 | Firefox HTTP Obseevatory | Verificación de encabezados HTTP |
| 5 | OWASP | Guía de pruebas |

# Herramientas otras

| N. | Herramienta | Uso |
|---|---|---|
| 1 | Wappalyzer | Identificación de tecnologías web |
| 2 | SecList | Listas de pruebas de seguridad |
| 3 | Searchsploit | Búsqueda de exploits |
| 4 | Metasploit | Framework de explotación |

# Sesión 4: Demostración

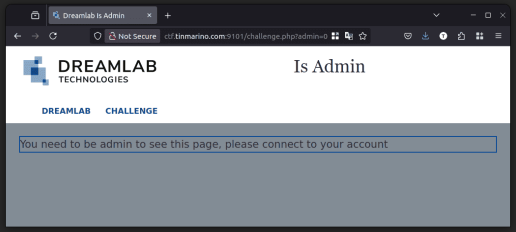
## (Módulo 1: Contexto)

# Leer la barra de URL

CTF: Is-Admin: 9101

Intenta obtener la *flag* en la página web del desafío. Para eso, deberás usurpar la identidad del usuario administrador para luego descubrir la *flag*.

- La *flag* se revelará en la pagina web para un usuario *admin* y tendrá el siguiente formato: `Dreamlab{flag-xxxxxx}`



# Leer el código fuente

CTF: Matrix: 9090

Tu misión: hackear el sitio web de la Matrix para encontrar la *flag* escondida.

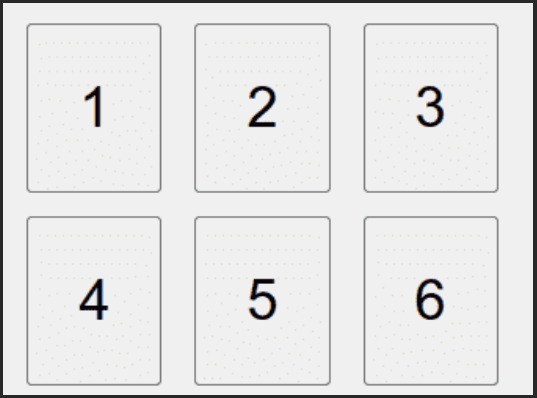
- La *flag* está en la Matrix, solo hay que saber leerla.

# Jugar

CTF: Pin Pass: 9302

Tu misión: descubrir el PIN correcto que desbloqueará la revelación de la bandera. Buena suerte!

- La *flag* será retornada en texto HTML. Por ejemplo, se puede buscar «flag» en las repuestas.



# Servidor y cliente

## CTF: PHP en JS: 9103

Bienvenido al desafío «Javascript y PHP»!

Tu misión, lograr controlar el servidor y demostrarlo encontrando la *flag*. Buena suerte!

- La *flag* es el contenido del archivo `/var/www/flag.txt`.

# Módulo 2: Ciberseguridad

# Módulo 2: Ciberseguridad

| S | Nombre | Descripción |
|---|---|---|
| 1 | Definición | Significado del término «ciberseguridad» |
| 2 | Terreno | Entorno en el que se desarrolla la ciberseguridad |
| 3 | Ciberataque | Fases involucradas en un ciberataque |
| 4 | Paisaje | Panorama global de la ciberseguridad |

# Sesión 1: Definición

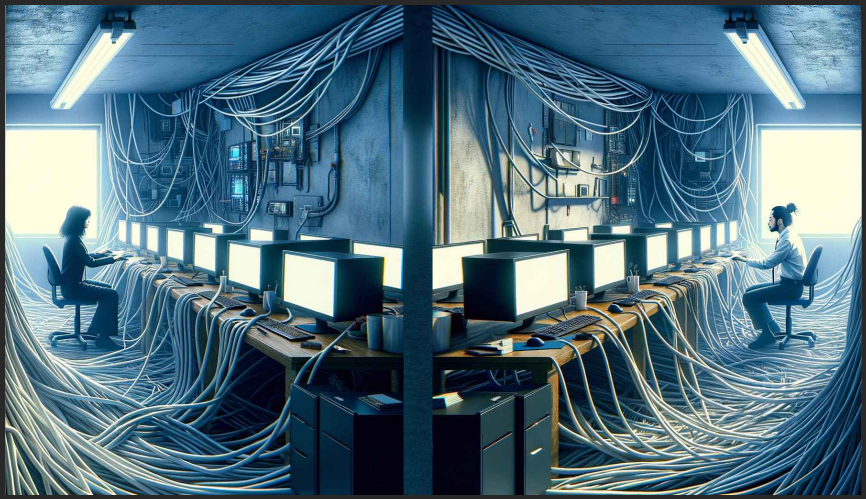
## (Módulo 2: Ciberseguridad)

# ¿Qué es la ciberseguridad?

| Raíz | Definición |
| --- | --- |
| Ciber | Computador |
| Seguridad | Pelea |

# Pelea con computadores

# Pelea con computadores



# La ciber (o computación)

1. Desarrollo de Software
2. Administración de Sistemas
3. Redes y Comunicaciones
4. Ciencia de Datos
5. Nube
6. **Seguridad**

Para participar en el terreno ciber, hay que saber utilizar un **computador**.

# La seguridad (o pelea)

| Fecha | Terreno | Arma | Lugar |
|---|---|---|---|
| -8000 | Tierra | masa, bastón | África, China |
| -2200 | Mar | botes de papiro | Egipto |
| 1911 | Aire | avión de hélice | Francia |
| 1957 | Espacio | satélite espía | Rusia, USA |
| 2010 | Ciber | gusano informático | Irán |

# Sesión 2: Terreno

## (Módulo 2: Ciberseguridad)

# El terreno ciber

| N. | Terreno | Permite | Como |
|---|---|---|---|
| 1 | Nuevo | El que llega primero reclama | colonización |
| 2 | Barato | un terreno al cual todos pueden acceder. | agua |
| 3 | Conectado | Y se puede apuntar lejos | telescopio |
| 4 | Rápido | a la velocidad luz | relámpago |
| 5 | Anonimizado | sin que nadie sepa quién fue. | invisibilidad |

La ciberseguridad tiene el viento a favor.

# La ciberseguridad tiene el viento a favor

# Un terreno asimétrico

Atacar y defender son dos profesiones distintas.

# La ciberdefensa

**¡Hay que saber atacar para poder defender!**

Por ejemplo, para buscar extraterrestres, meterse en el lugar de extraterrestres que buscarían humanos. (tener una metodología pragmática, tener humildad).

Otro ejemplo, para tapar 1000 hoyos de manera industrial, primero tapar un hoyo de manera artesanal. (no hacer optimización prematura, ensuciar sus manos).

# Sesión 3: Ciberataque

## (Módulo 2: Ciberseguridad)

# El ciberataque

1. Buscar víctimas (humanas).
2. Buscar superficie de exposición de sus víctimas.
3. Buscar vulnerabilidades en la superficie.
4. Explotar vulnerabilidades.
5. Mantener persistencia en los computadores infectados.
6. Ejecutar la carga (ex: robar dinero).

# ¿Qué es una vulnerabilidad?

«Un ciberataque se hace mediante la explotación de una **vulnerabilidad**.»

# ¿Qué es una vulnerabilidad?

«Un ciberataque se hace mediante la explotación de una **vulnerabilidad**.»

Una vulnerabilidad informática es lo que permite que un programa haga algo que **sus creadores o usuarios no habían contemplado**.

# La búsqueda de vulnerabilidad

Una **vulnerabilidad** informática aquello que que permite que un programa realice un acción que sus creadores o usuarios no habían contemplado.

1. ¿Que hace el programa?
2. ¿Como lo implementa?
3. ¿Como lo implementaría yo para que sea seguro?
4. ¿Lo implementa de mi forma? ¿Sino, porque?

# Sesión 4: Paisaje

## (Módulo 2: Ciberseguridad)

# Humanos contra humanos

| Sujeto | Pre verbo | Verbo |
|---|---|---|
| Yo | no | me quiero hackear. |
| Mi circulo de confianza | probablemente no | me quiere hackear. |
| Los animales non humanos | no | me pueden hackear. |
| Los extraterrestres | no | me quieren hackear. |
| Todo los demás (8G humanos) | potencialmente | me quieren hackear. |

# Actores internacionales

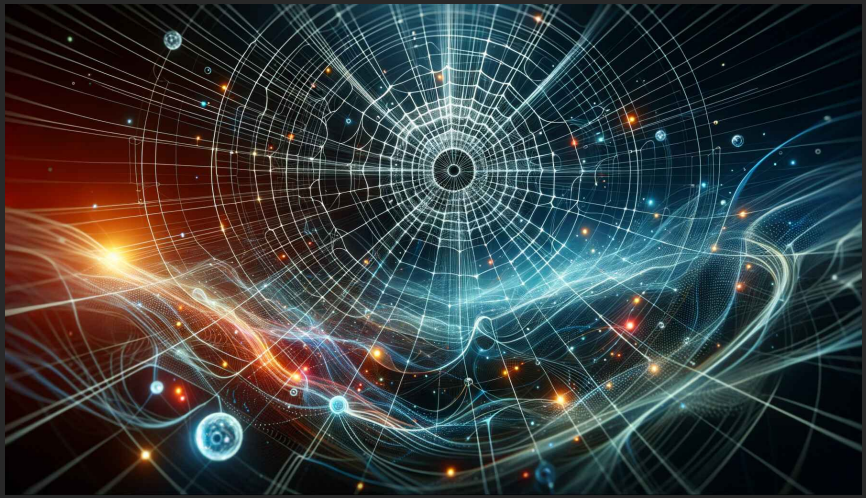
# Actores nacionales

Comparar con el mapa de los actores estatales en Francia.

| Sector | Ejemplo |
|---|---|
| Público | Gobierno, defensa, interior |
| Privado | Bancos, telecom, servicio |
| Universitario | Investigadores, practicantes |
| Independientes | Freelancer, mafia, RASS |

(estos actores también saben usar un computador para otra cosa que pelear)

# Módulo 3: Web



# Módulo 3: Web

| S | Nombre | Descripción |
|---|---|---|
| 1 | Tecnología | Componentes de la pila tecnológica web |
| 2 | HTTP | Protocolo principal de la web |
| 3 | TLS | Estándar de criptografía para la web |
| 4 | Extra | Cabeceras HTTP y uso de cURL |

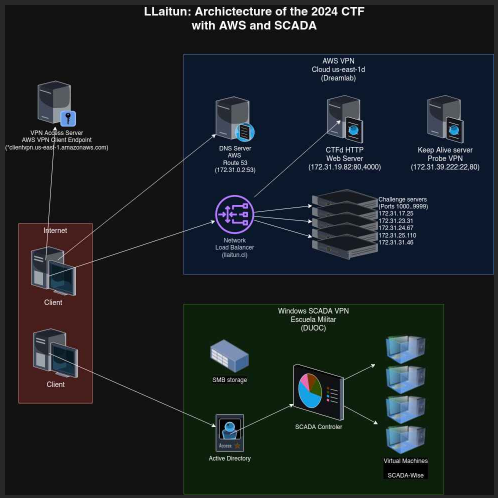
# Sesión 1: Tecnología web

## (Módulo 3: Web)

# Componentes

1. Cliente
2. Servidor
3. Servicios (Micro servicios)
4. Funciones (Serverless)

# Arquitectura: cliente[s] y servidor[es]



# Componentes de Frontend

# HTML: HyperText Markup Language

```
<nav id="id_cyber" tabindex="-1" class="sidebar sidebar2
 ↪  w3-small w3-center">
  <!-- Cyber 6: El terreno ciber -->
  <a href="./pdf/2024_slide_ciber1_dreamlab_v2.00.pdf"
    onclick="hideHome(this);"
  >
    <!-- Description -->
    <div class="description">
      PDF presentation in Spanish<br/><br/>
    </div>
    <img src="./img/single-levi-first-page-256.jpg">
    <p>Ciber Land<br/>Slide</p>
  </a>
</nav>
```

# CSS: Cascading Style Sheets

```css
/* Solarized colors */
html, body {
  background: #002b36;
  color: #fdf6e3;
}

/* Interactive hide */
input[type=checkbox]:checked ~ .sidebar { display: none; }

#bar_opener {
  position: absolute; top: 0; left: 0;
}

.sidebar:target { display: none; }
```

# JS: JavaScript

```
function mainPro() {
  declareGlobal(); document.getElementById("home").focus();
}

function declareGlobal() {
  // Declare globals (array of openable navigation ids)
  window.aNavId = [ "id_cv", "id_web"]

  window.Key = {
    BACKSPACE: 8, TAB: 9,
  }
}

window.onload = mainPro;
```

# Componentes de *backend*

1. Servidor de acceso VPN
2. Balanceador de carga
3. Proxy inverso
4. Servidor web
5. Servidor aplicativo
6. Base de datos

# PHP: PHP Hypertext Preprocessor

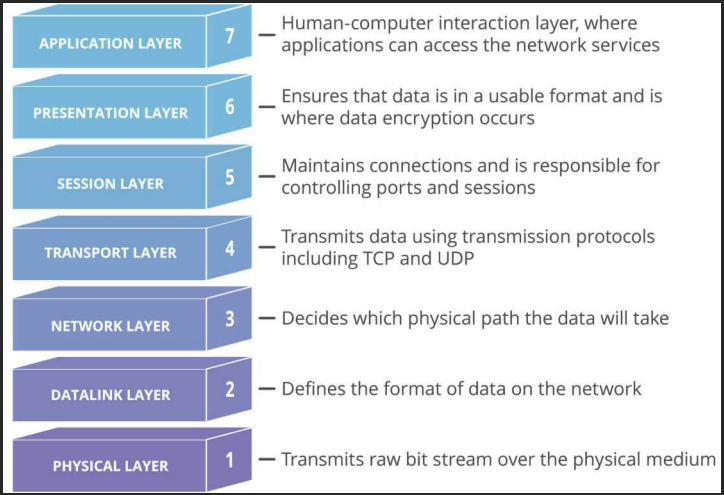
```php
<?php
// Remove the 'guid' cookie
if (isset($_COOKIE['guid'])) {
    setcookie("guid", "", time() - 3600, "/");
}

// Redirect the user to the login page or home page
header("Location: index.php");
exit;
?>
```

# Sesión 2: Protocolo HTTP

## (Módulo 3: Web)

# «Dos puntos primero»

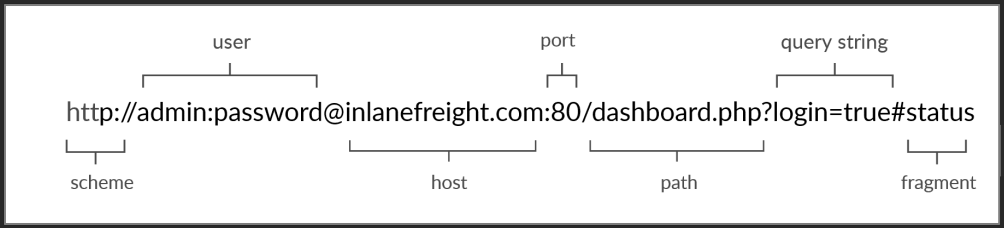
# Modelo OSI


APPLICATION LAYER
7
Human-computer interaction layer, where applications can access the network services
PRESENTATION LAYER
6
Ensures that data is in a usable format and is where data encryption occurs
SESSION LAYER
5
Maintains connections and is responsible for controlling ports and sessions
TRANSPORT LAYER
4
Transmits data using transmission protocols including TCP and UDP
NETWORK LAYER
3
Decides which physical path the data will take
DATALINK LAYER
2
Defines the format of data on the network
PHYSICAL LAYER
1
Transmits raw bit stream over the physical medium

# Modelo OSI

| Capa | Nombre | Ejemplo |
|---|---|---|
| 7 | Aplicación | **HTTP**, **DNS**, FTP, SNMP, Telnet |
| 6 | Presentación | **TLS**, SSL |
| 5 | Sesión | **Cookies**, NetBIOS, PPTP |
| 4 | Transporte | **TCP**, **UDP** |
| 3 | Red | **IP**, ARP, **ICMP**, IPSec |
| 2 | Enlace | **MAC** PPP, ATM |
| 1 | Física | **WiFi**, Ethernet, USB, Bluetooth |

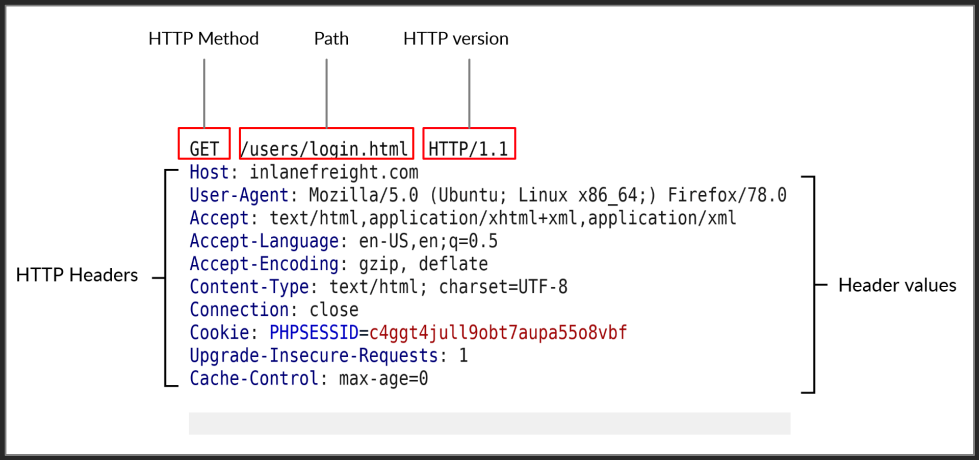
# Esquema URL


user
port
query string
http://admin:password@inlanefreight.com:80/dashboard.php?login=true#status
scheme
host
path
fragment

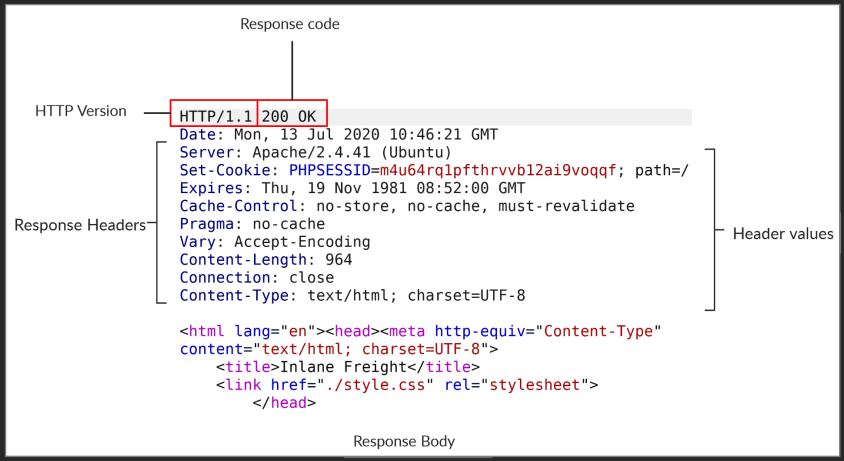
# Esquema URL

| Componente | Ejemplo | Descripción |
|---|---|---|
| **Esquema** | `http://` `https://` | Se utiliza para identificar el protocolo al que accede el cliente y termina con dos puntos y doble barra ( `://` ). |
| **Información de usuario** | `admin:password@` | Es un componente opcional que contiene las credenciales (separadas por dos puntos `:` ) utilizadas para autenticarse en el host y se separa de este con una arroba ( `@` ). |
| **Host** | `inlanefreight.com` | Indica la ubicación del recurso. Puede ser un nombre de dominio o una dirección IP. |
| **Puerto** | `:80` | Separa el Host con dos puntos ( `:` ). Si no se especifica, los esquemas `http` usan por defecto el puerto 80 y `https` el puerto 443. |
| **Ruta** | `/dashboard.php` | Indica el recurso al que se accede, ya sea un archivo o una carpeta. Si no se especifica, el servidor devuelve el índice por defecto (por ejemplo, `index.html` ). |
| **Cadena de consulta** | `?login=true` | Inicia con un signo de interrogación ( `?` ) e incluye un parámetro (por ejemplo, `login` ) y un valor ( `true` ). Se pueden agregar múltiples parámetros separados por un ampersand ( `&` ). |
| **Fragmentos** | `#status` | Son procesados por los navegadores en el cliente para localizar secciones dentro del recurso principal (por ejemplo, un encabezado o una sección de la página). |

# Solicitud


HTTP Method
Path
HTTP version
GET
/users/login.html
HTTP/1.1
HTTP Headers
Host: inlanefreight.com
User-Agent: Mozilla/5.0 (Ubuntu; Linux x86_64;) Firefox/78.0
Accept: text/html,application/xhtml+xml,application/xml
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: text/html; charset=UTF-8
Connection: close
Cookie: PHPSESSID=c4ggt4jull9obt7aupa55o8vbf
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0
Header values

# Respuesta


Response code
HTTP Version
HTTP/1.1 200 OK
Date: Mon, 13 Jul 2020 10:46:21 GMT
Server: Apache/2.4.41 (Ubuntu)
Set-Cookie: PHPSESSID=m4u64rq1pfthrvvb12ai9voqqf; path=/
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate
Pragma: no-cache
Vary: Accept-Encoding
Content-Length: 964
Connection: close
Content-Type: text/html; charset=UTF-8
Response Headers
Header values
<html lang="en"><head><meta http-equiv="Content-Type"
content="text/html; charset=UTF-8">
<title>Inlane Freight</title>
<link href="./style.css" rel="stylesheet">
</head>
Response Body

# Sesión 3: TLS: Transport Layer Security

(Módulo 3: Web)

# HTTPS: SSL y TLS

- **SSL (Secure Sockets Layer)**: Protocolo de seguridad que establece un enlace cifrado entre un servidor web y un navegador.
- **TLS (Transport Layer Security)**: Sucesor de SSL, proporciona una mayor seguridad y es el estándar actual.

# Funciones clave: la criptografía

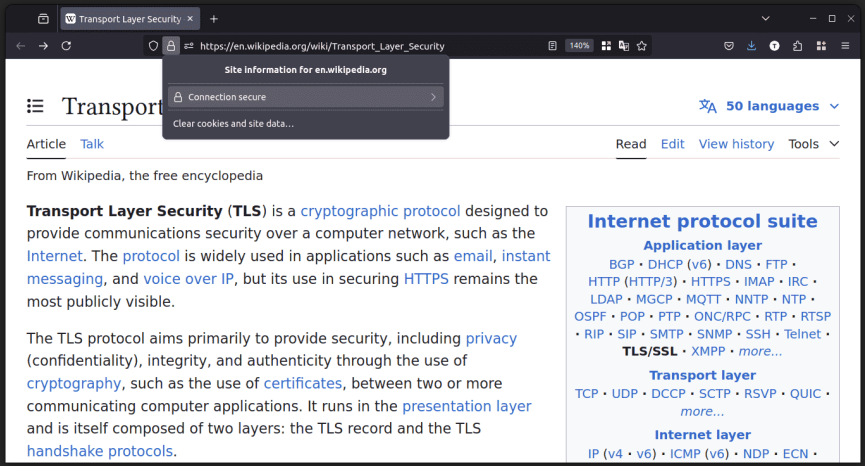
1. **Cifrado**: Protege la información transmitida entre el cliente y el servidor.
2. **Autenticación**: Verifica la identidad del servidor y, opcionalmente, del cliente.
3. **Integridad de los datos**: Asegura que los datos no sean alterados durante la transmisión.

# Certificados TLS

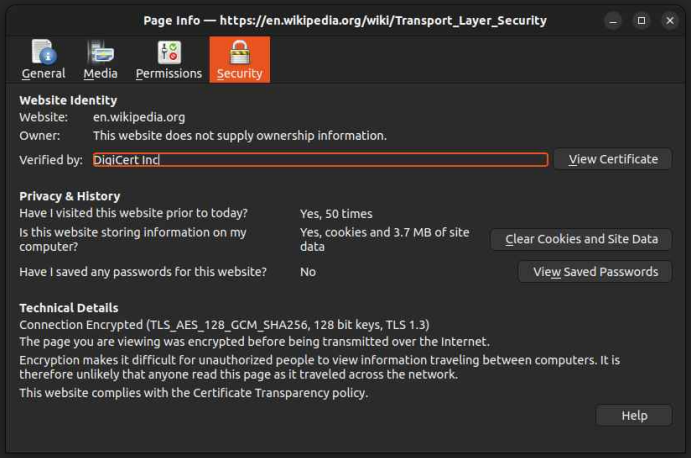
1. **Certificados de Validación de Dominio (DV)**: Verifican que el solicitante tenga control sobre el dominio.
2. **Certificados de Validación de Organización (OV)**: Incluyen información sobre la organización y requieren verificación adicional.
3. **Certificados de Validación Extendida (EV)**: Proporcionan el más alto nivel de confianza y requieren una verificación exhaustiva de la organización.

- Los certificados TLS son esenciales para establecer conexiones seguras y son un factor clave en la confianza del usuario en un sitio web.

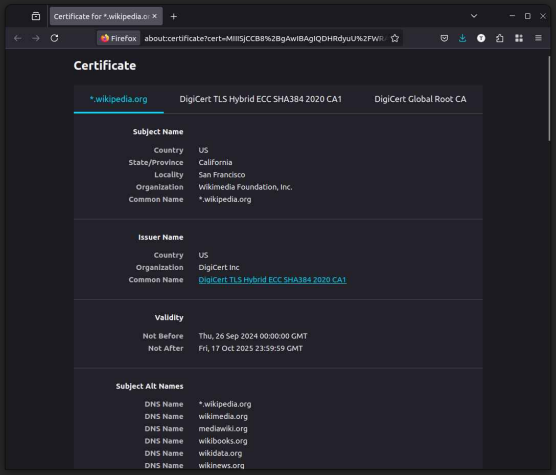
# Certificados TLS



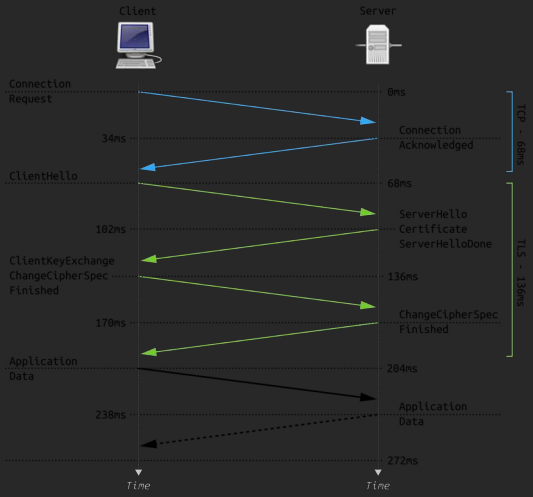
# Certificados TLS



# Certificados TLS



# Apretón de manos



# Recordar

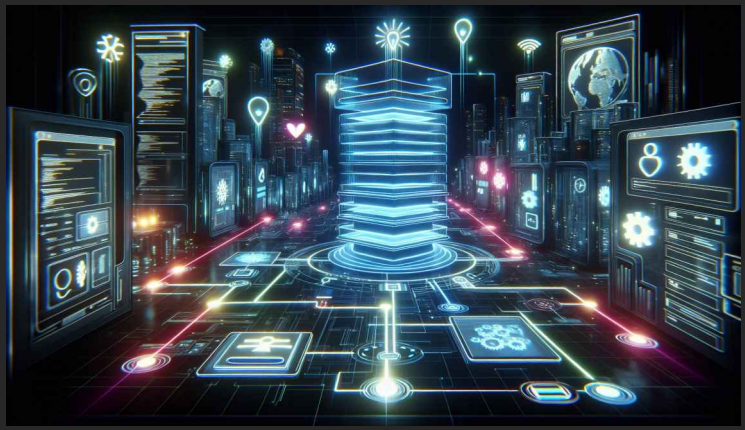
**SSL and TLS protocols**

| Protocol | Published | Status |
|---|---|---|
| SSL 1.0 | Unpublished | Unpublished |
| SSL 2.0 | 1995 | Deprecated in 2011 (RFC 6176) |
| SSL 3.0 | 1996 | Deprecated in 2015 (RFC 7568) |
| TLS 1.0 | 1999 | Deprecated in 2021 (RFC 8996)[20][21][22] |
| TLS 1.1 | 2006 | Deprecated in 2021 (RFC 8996)[20][21][22] |
| TLS 1.2 | 2008 | In use since 2008[23][24] |
| **TLS 1.3** | 2018 | In use since 2018[24][25] |

Old version, not maintained | Old version, still maintained | **Latest version**

# Sesión 3: Extra: Cabeceras y cURL

(Módulo 3: Web)



# Tipo de cabeceras

| N. | Nombre | Describe | Ejemplo |
|---|---|---|---|
| 1 | Genéricos | Tipo de mensaje | Connection: close |
| 2 | Entidades | Tipo contenido | Content-Type: text/json |
| 3 | Solicitud | Contenido de solicitud | Host: www.uc.cl |
| 4 | Respuesta | Instrucciones | Set-Cookie: PHPSESSID=abcdefabcdef |
| 5 | Seguridad | Reglas | Content-Security-Policy: script-src 'self' |

# Métodos HTTP

| Método | Uso |
|---|---|
| OPTIONS | Leer los métodos aceptados |
| GET | Leer un recurso |
| POST | Escribir un recurso |
| HEAD | Leer los encabezados |
| PUT | Crear nuevo recurso |
| DELETE | Elimina un recurso |
| PATCH | Modificar un recurso |

# Códigos de respuesta

| Tipo | Descripción |
|---|---|
| 1xx | Proporciona información, no afecta el procesamiento. |
| 2xx | Indica que la solicitud fue exitosa. |
| 3xx | Indica que el servidor redirige al cliente. |
| 4xx | Señala solicitudes incorrectas del cliente. |
| 5xx | Indica problemas con el servidor HTTP. |

# Ejemplos de códigos de respuesta

| Código | Descripción |
|---|---|
| 200 OK | Solicitud exitosa |
| 302 Found | Redirige al cliente a otra URL. |
| 400 Bad Request | Solicitudes mal formadas. |
| 403 Forbidden | El cliente no tiene acceso adecuado al recurso. |
| 404 Not Found | Recurso solicitado no existe en el servidor. |
| 500 Internal Server Error | El servidor no puede procesar la solicitud. |

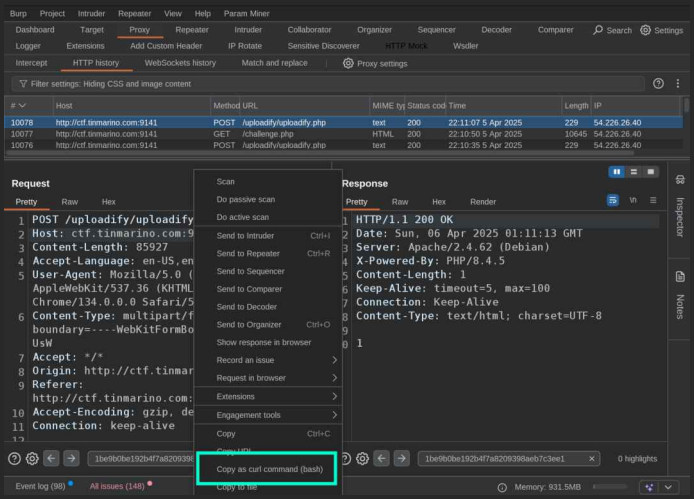
# cURL

```
# Help
curl -h

# HTTP GET
curl ipinfo.io

# Include headers
curl -vvv -i https://ipinfo.io

# Only HEAD
curl -I ipinfo.io
```

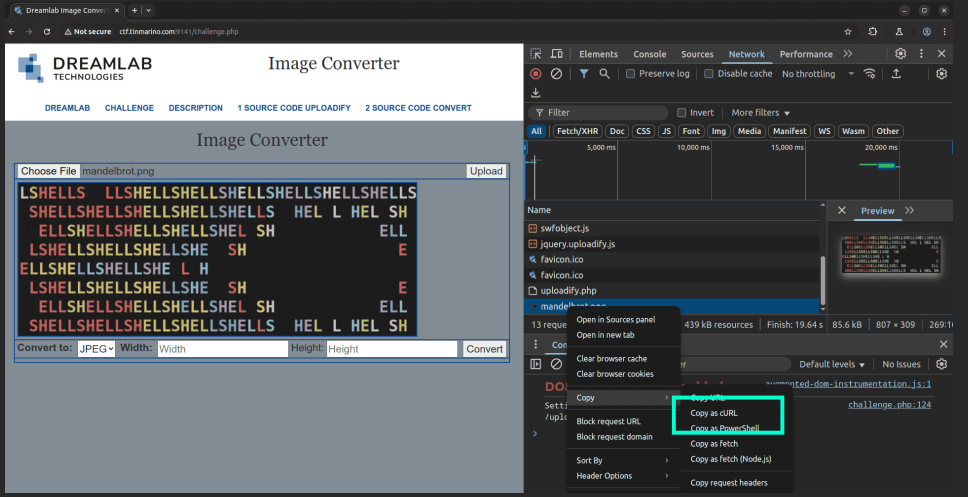
# cURL

```
curl -X POST \
  --path-as-is -i -s -k \
  --proxy localhost:8080 \
  -H 'Cookie: PHPSESSID=d720jtoh9td1qf4rvattst1p54' \
  -H 'Content-Type: application/json' \
  -d '{"search":"flag"}' \
  http://localhost:31722/admin/search.php &
```

# cURL desde BurpSuite



# cURL desde Chrome



# Módulo 4: Hacktitud

# Módulo 4: Hacktitud

| S | Nombre | Descripción |
|---|---|---|
| 1 | Espíritus | El hackeo es un estado de espíritu |
| 2 | Reglas | Normativas de los CTFs |
| 3 | Escuela | Estrategias para potenciar habilidades en pentesting web |
| 4 | Trucos | Recomendaciones para realizar pentesting web |

# Sesión 1: Espíritus

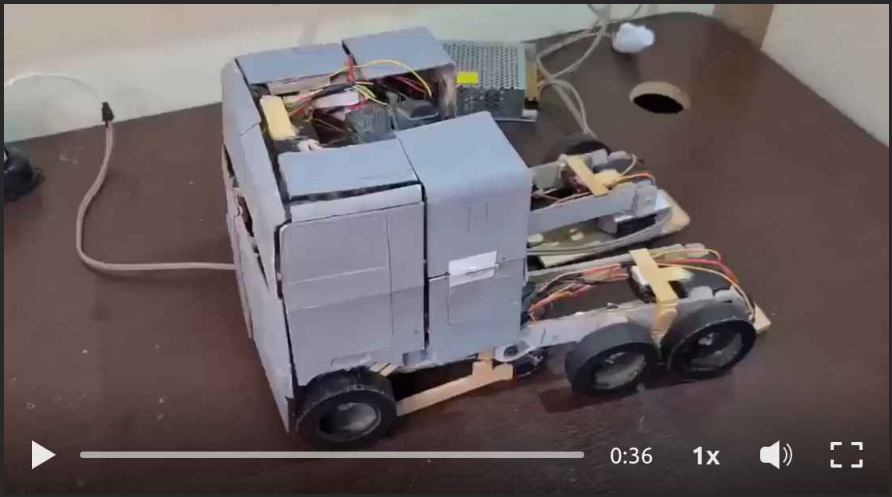
## (Módulo 4: Hacktitud)

# El hackeo es un estado de espíritu

# El hackeo es un estado de espíritu

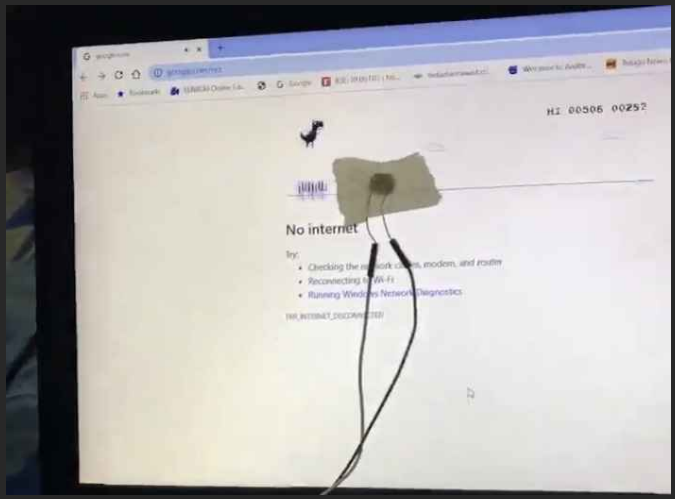
# El hackeo es un estado de espíritu

# El hackeo es un estado de espíritu



# El hackeo es un estado de espíritu

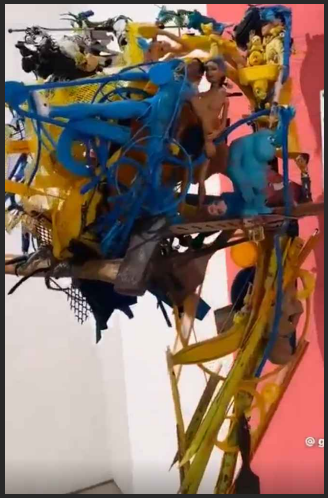
# El hackeo es un estado de espíritu



# El hackeo es un estado de espíritu

# El hackeo es un estado de espíritu

# Artista: Thomas Deininger



# El hackeo es un estado de espíritu

# El hackeo es un estado de espíritu

# El hackeo es un estado de espíritu

# Fuente Julien Metayer

# El hackeo es un estado de espíritu

| | | | |
|---|---|---|---|
| Persistencia | Creatividad | Curiosidad | Mentalidad de juego |
| Pasión | Confianza | Documentación | Atención al detalle |
| Disciplina | Autocrática | Empatía | Toma de riesgos |
| Cultura | Lógica | Técnica | Mentalidad de aprendizaje |

# El hackeo es un estado de espíritu

| | | | |
|---|---|---|---|
| Persistencia | Creatividad | Curiosidad | Mentalidad de juego |
| Pasión | Confianza | Documentación | Atención al detalle |
| Disciplina | Autocrática | Empatía | Toma de riesgos |
| Cultura | Lógica | Técnica | Mentalidad de aprendizaje |

1. Sé tu mismo!
2. Disfruta cada momento!

# El hackeo es un estado de espíritu

Cual es la diferencia entre un hacker (actor de amenazas) y un usuario legitimo?

# El hackeo es un estado de espíritu

Cual es la diferencia entre un hacker (actor de amenazas) y un usuario legitimo?

La misma que entre una persona de buena fe y una persona ilegitima ...

# El hackeo es un estado de espíritu

Cual es la diferencia entre un hacker (actor de amenazas) y un usuario legitimo?

La misma que entre una persona de buena fe y una persona ilegitima ...

.. la intencionalidad.

# El hackeo es un estado de espíritu

Cual es la diferencia entre un hacker (actor de amenazas) y un usuario legitimo?

La misma que entre una persona de buena fe y una persona ilegitima ...

.. la intencionalidad.

**El estado de espíritu!**

# Sesión 2: Reglas

## (Módulo 4: Hacktitud)

# ¿Cuáles son las reglas de la ciber?

1. No pierdas de vista tu objetivo!
2. Sé legitimo en todo tiempo!

# ¿Cuáles son las reglas de la ciber?

1. No pierdas de vista tu objetivo!
2. Sé legitimo en todo tiempo!

Cuando un jugador ha sido detectado, está descalificado.

# ¿Cuáles son las reglas de la ciber?

1. No pierdas de vista tu objetivo!
2. Sé legitimo en todo tiempo!

Cuando un jugador ha sido detectado, está descalificado.

El jugador que controla más computadores va ganando.

# ¿Cuáles son las reglas de la ciber?

1. No pierdas de vista tu objetivo!
2. Sé legitimo en todo tiempo!

Cuando un jugador ha sido detectado, está descalificado.

El jugador que controla más computadores va ganando.

El juego nunca acaba.

# ¿Cuáles son las reglas de la ciber?

1. No pierdas de vista tu objetivo!
2. Sé legitimo en todo tiempo!

Cuando un jugador ha sido detectado, está descalificado.

El jugador que controla más computadores va ganando.

El juego nunca acaba.

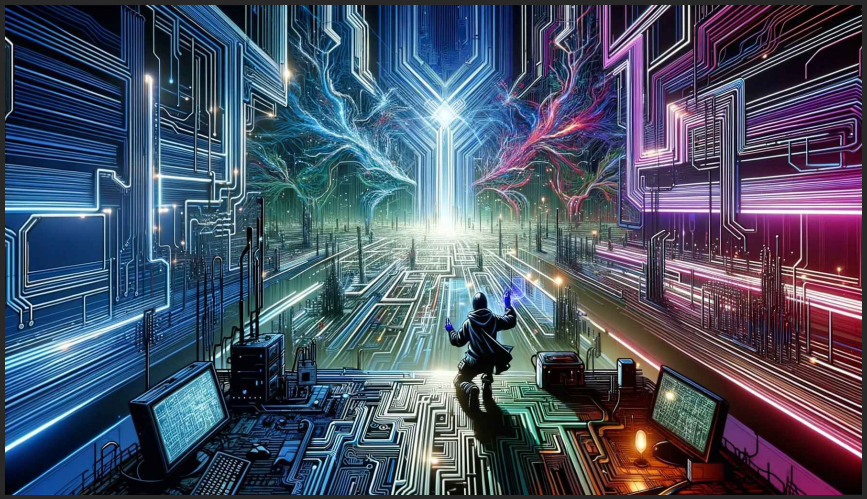
**La ciberseguridad es un camino no un destino**

# ¿Cuáles son las reglas de la ciber?

En realidad no hay reglas!

Pero para practicar la pelea con computador, existe un deporte, el CTF que el si tiene reglas!

# ¿Cuáles son las reglas del juego?



# Las reglas de un CTF

1. No **atacar a otros participantes** ni a sus sistemas.
2. Permitir que todos los jugadores tengan la **oportunidad de participar**.
3. Mantener la ética y el **respeto** en la competencia.

# Las reglas de un CTF

1. No **atacar a otros participantes** ni a sus sistemas.
2. Permitir que todos los jugadores tengan la **oportunidad de participar**.
3. Mantener la ética y el **respeto** en la competencia.

**recomendaciones**

1. Aprovechar vulnerabilidades no contempladas por los organizadores.
2. Eliminar cualquier rastro de actividad.
3. Divertirse y aprender durante el proceso.

# Las reglas de un CTF

1. No **atacar a otros participantes** ni a sus sistemas.
2. Permitir que todos los jugadores tengan la **oportunidad de participar**.
3. Mantener la ética y el **respeto** en la competencia.

**recomendaciones**

1. Aprovechar vulnerabilidades no contempladas por los organizadores.
2. Eliminar cualquier rastro de actividad.
3. Divertirse y aprender durante el proceso.

Todo lo que no está explícitamente prohibido se considera permitido.

# Sesión 3: Escuela

## (Módulo 4: Hacktitud)

# ¿Como mejorar su capacidad?

1. **Clases** como la presente.
2. **Ejercicio** tipo CTF: HTB o PortSwigger.
3. **Casos reales** en programas de Bug Bounty como CyScope o Intigriti.
4. **Desarrollo** de un vídeo juego, una aplicación web o un utilitario.
5. **Lectura** de bases de programación y seguridad.
6. **Cultura** en *blog* y *chats* de hacker.

# Sesión 4: Trucos

## (Módulo 4: Hacktitud)

# Enseñanzas de pentest web

1. Invertir tiempo.
2. Empezar con un IDOR mediante RUT.
3. Tener en la mente de romper el cliente, no ganar recompensas.
4. Ser legitimo primero, para reconocer el buen camino.
5. Mantener notas.
6. No exponerse y parar la pruebas antes cualquier duda.

# Enseñanzas de retroingeniería

1. Avanzar con hipótesis y validaciones.
2. No olvidar de donde uno viene, el porque uno está estudiando eso ahora.
3. No irse demasiado profundo demasiado tiempo
4. No quedarse superficial tampoco.
5. Ser oportunista.
6. Encontrar su propio estilo.

# Buscar caracteres pocos comunes

```
| & ; ( ) < > space tab newline # Bash metacaracters
* @ # ? - $ ! 0               # Shell special parameters
" ' \ % $ - { } [ ] * !       # Additions
```

Mi preferida es la nueva linea: `%0A`.

# Aprender la capacidad de servir

1. VPN
2. Ngrok proxy
3. AWS