

```
datetime
Material:
__init__(self, titulo, autor, codigo):
self.titulo = titulo
self.autor = autor
self.codigo = codigo
self.disponible = True
__str__(self):
estado = "Disponible" if self.disponible else "Prestado"
return f"[{self.codigo}] {self.titulo} - {self.autor} ({estado})"
```

```
Libro(Material):
__init__(self, titulo, autor, codigo, paginas):
super().__init__(titulo, autor, codigo)
self.paginas = paginas
Revista(Material):
__init__(self, titulo, autor, codigo, edicion):
super().__init__(titulo, autor, codigo)
self.edicion = edicion
```

```
Biblioteca:
__init__(self):
self.inventario = []
self.historial_prestamos = []
agregar_material(self, material):
self.inventario.append(material)
print(f"Sistema: '{material.titulo}' agregado al inventario.")
```

```
prestar_material(self, codigo_buscado):
for item in self.inventario:
```

```
class Biblioteca:
def __init__(self):
self.inventario = []
self.historial_prestamos = []
def agregar_material(self, material):
self.inventario.append(material)
print(f"Sistema: '{material.titulo}' agregado")
def prestar_material(self, codigo_buscado):
for item in self.inventario:
if item.codigo == codigo_buscado:
if item.disponible:
item.disponible = False
fecha = datetime.datetime.now().strftime("%Y-%m-%d")
self.historial_prestamos.append(f"{item.titulo} - {fecha}")
return f"Éxito: Has retirado '{item.titulo}'."
return "Error: El material ya está prestado."
return "Error: Código no encontrado."
```

```
def mostrar_inventario(self):
print("\n--- Inventario Actual ---")
for item in self.inventario:
print(item)
def menu():
mi_biblioteca = Biblioteca()
# Población inicial de datos
mi_biblioteca.agregar_material(Libro("Cien Años de Soledad"))
mi_biblioteca.agregar_material(Revista("Nación"))
```

```
while True:
print("\n1 Ver Inventario\n2 Prestar Material\n3 Salir")
```


CUANDO GENERAMOS CONTRASEÑA

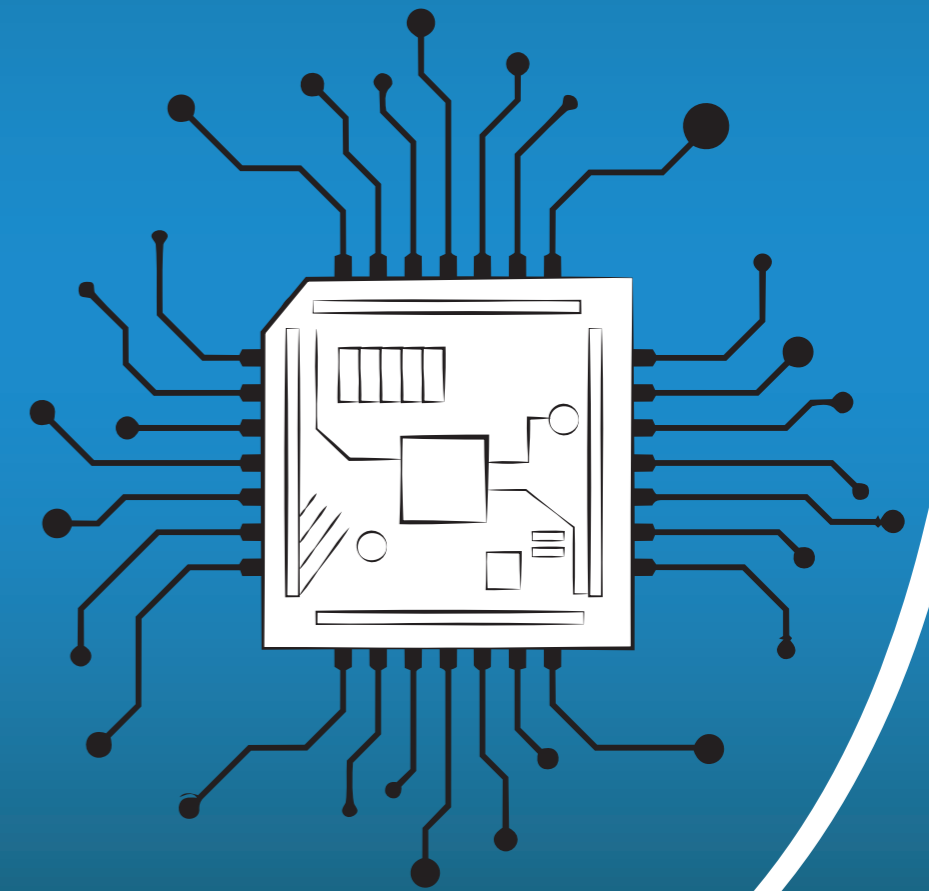


**A TRAVÉS DE
UN ALGORITMO**

#

**LA CPU LAS CONVIERTE EN UN
CÓDIGO *HASH***

eb1af65b1291460c62a25471178c37



SI UTILIZAMOS UNA CONTRASEÑA CORTA O MUY SENCILLA...



eb1af65b1291460c62a25471178c37
e22ae475bc58097fab1460c62a2178cc
e4291460c62a2547c62a217875bc2aea
1460c62a2178291460c62a254711 abe2
c62a2547c62a217eb1af65b1291ae375

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

LOS HACKERS SE APROVECHAN DE NUESTRAS ENTRADAS Y REGISTROS EN WEBS NO MUY SEGURAS PARA APROPIARSE MUY FACILMENTE DE USUARIOS Y HASH, QUE POR SÍ SOLOS NO LES SIRVE, PERO...

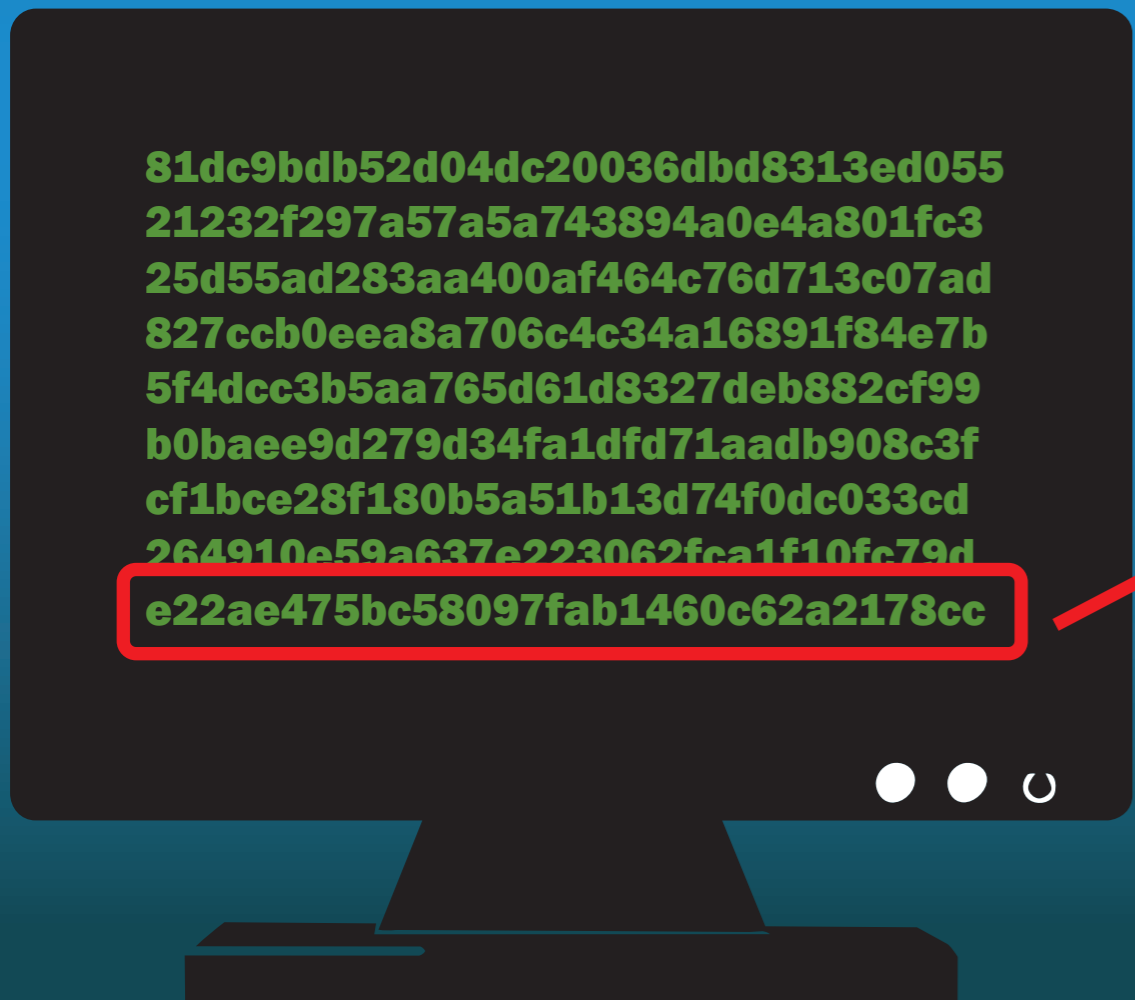
LOS HACKERS CONSIGUEN INFORMACIÓN BÁSICA SOBRE EL PERFIL DE LOS USUARIOS DE ESAS WEBS, Y CON ESA INFORMACIÓN GENERAN HASH ALEATORIOS DE LAS CONTRASEÑAS MÁS OBVIAS Y COMUNES.

HACEN BARRIDOS SISTEMÁTICOS BUSCANDO COINCIDENCIAS Y...

```
81dc9bdb52d04dc20036dbd8313ed055  
21232f297a57a5a743894a0e4a801fc3  
25d55ad283aa400af464c76d713c07ad  
827ccb0eea8a706c4c34a16891f84e7b  
5f4dcc3b5aa765d61d8327deb882cf99  
b0baee9d279d34fa1dfd71aadb908c3f  
cf1bce28f180b5a51b13d74f0dc033cd  
264910e59a637e223062fca1f10fc79d  
e22ae475bc58097fab1460c62a2178cc
```

```
e22ae475bc58097fab1460c62a2178cc  
e4291460c62a2547c62a2178750c2aea  
1460c62a2178291460c62a254711 abe2  
c62a2547c62a217eb1af65b1291ae375
```

```
.....  
.....  
.....  
.....  
.....
```



EUREKA!

e22ae475bc58097fab1460c62a2178cc



CON LA COINCIDENCIA DE ESTE HASH YA PUEDEN OBTENER USUARIO Y CONTRASEÑA CON LOS QUE PRUEBAN SUERTE EN TODOS AQUELLOS PORTALES HABITUALES PARA SUPLANTAR AL USUARIO:

NETFLIX





**DEMOSTRARÉ CÓMO DE SENCILLO ES AVERIGUAR UNA CONTRASEÑA
SI ÉSTA ES MUY CORTA O MUY COMÚN.**

PERO...¿CÓMO?

LOS VISITANTE DEBEN INSERTAR UNA CONTRASEÑA “SECRETA” EN UN DISPOSITIVO. AVERIGUARÉ CUÁL ES ESA CONTRASEÑA Y LES MOSTRARÉ UN CÓDIGO QR QUE PODRÁN LEER CON SUS TELÉFONOS Y LES MOSTRARÁ LA CONTRASEÑA QUE HABÍAN INTRODUCIDO.



Hash MD5:

eb1af65b129146160c62a25471178c37



GRACIAS!