

KKSI KOMPETISI KOMUNITAS SIBER INDONESIA

27-28 November 2019, Grand Cempaka Business Hotel, Jakarta Pusat



NAMA TIM : [Cyber Security Trunojoyo]

Ketua Tim

1. Wijanarko Putra Rajeb

Member

1. Muhamad Hendrik Wicaksono
2. Muhammad Zaelani

Universitas Trunojoyo Madura
KKSI 2019 UMUM - Surabaya

Daftar Isi

Testing

Testing [1 Point]	2
---------------------	---

Misc

Welcome To KKSII2019 [50 Point]	3
-----------------------------------	---

Forensics

Login Trafic [50 Point]	4
---------------------------	---

Read The Log [70 Point]	6
---------------------------	---

Member Have Journal [70 Point]	8
----------------------------------	---

Web

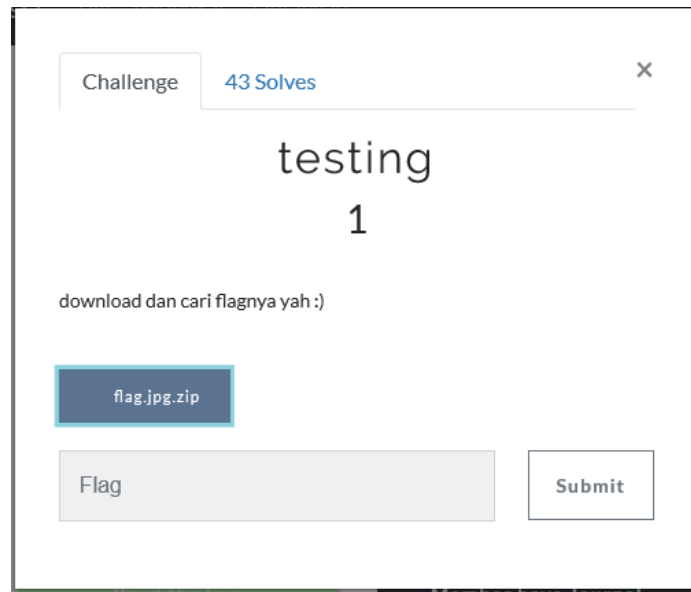
Tsunade Gambling Master [100 Point]	11
---------------------------------------	----

Limited Eval [200 Point]	13
----------------------------	----

Testing

Testing [1 Point]

Disediakan soal berikut:



Challenge 43 Solves

testing
1

download dan cari flagnya yah :)

flag.jpg.zip

Flag

Submit

Beserta file zip yang berisi flag.jpg berikut ini.

Flag=KKS12019{selamat_b3rjuang}

FLAG : KKS12019{selamat_b3erjuang}

Misc

Welcome To KKS12019 [50 Point]

Disediakan soal berikut.

Challenge

33 Solves

×

Welcome To KKS12019
50

Help me find the piece of flag

1663323d00434ad7#ca8ecca2b#22844

I just have md5 of full flag.

1fee4be0b38ae6b8722b49e4db037bbd

Submit with

KKS12019{}

Flag

Submit

Disajikan flag yang tidak utuh dan hash md5 dari flag tersebut. Kami menggunakan md5 online decrypter <https://www.md5online.org> kemudian menginputkan hash md5 yang sudah ada dan keluar hasil decrypternya.

MD5 Decryption

Enter your MD5 hash below and cross your fingers :

Decrypt

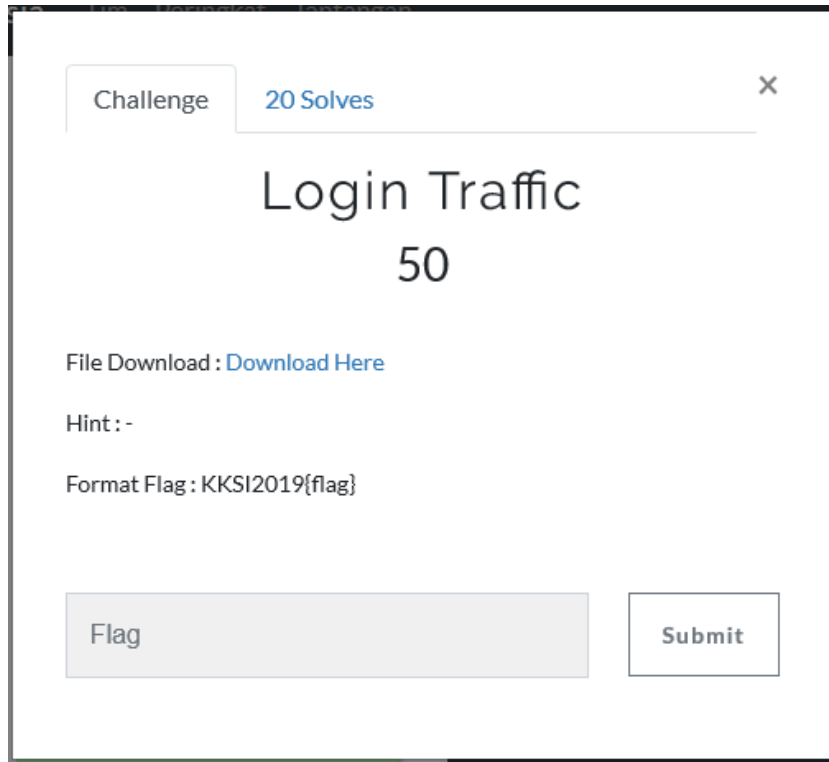
Found : 1663323d00434ad78ca8ecca2ba22844
(hash = 1fee4be0b38ae6b8722b49e4db037bbd)

FLAG : KKS12019{1663323d00434ad78ca8ecca2ba22844}

Forensics

Login Traffic [50 Point]

Disedikan Soal Berikut:



Challenge 20 Solves

Login Traffic

50

File Download : [Download Here](#)

Hint : -

Format Flag : KKS12019{flag}

Flag

Submit

Dengan file **login_traffic.pcapng** dan Description: File Download : [Download Here](#)

Dilakukan Analisa terhadap log tersebut menggunakan aplikasi WireShark, kami menemukan banyak sekali lalu lintas file. Namun tidak ada yang menarik dari semua file lalu lintas yang ada. Kami sempat berfikir sejenak. Kemudian kami ingat bahwa soal menunjukkan kata login dimana berarti biasanya sebuah login bisa menggunakan method GET atau POST. Setelah itu dilakukan pencarian log dengan string POST maka didapatkan sebuah baris log yang menunjukkan POST ke sebuah halaman.

No.	Time	Source	Destination	Protocol	Length	Leftover Capture Data	Length	Info
4381	17.452304	10.0.2.15	10.0.2.255	NBNS	92			Name query NB UPAD-NB
4382	17.452305	10.0.2.15	10.0.2.255	NBNS	92			Name query NB UPAD-NB
4418	18.549530	10.0.2.15	10.0.2.255	NBNS	92			Name query NB UPAD-NB
4431	19.395647	10.0.2.15	10.0.2.255	NBNS	92			Name query NB UPAD-NB
4432	19.549608	10.0.2.15	10.0.2.255	NBNS	92			Name query NB UPAD-NB
4437	19.550800	10.0.2.15	10.0.2.255	NBNS	92			Name query NB UPAD-NB
4438	19.550804	10.0.2.15	10.0.2.255	NBNS	92			Name query NB UPAD-NB
4518	21.094355	10.0.2.15	10.0.2.255	NBNS	92			Name query NB UPAD-NB
4520	21.234221	10.0.2.15	10.0.2.255	NBNS	92			Name query NB UPAD-NB
4520	21.234227	10.0.2.15	10.0.2.255	NBNS	92			Name query NB UPAD-NB
4521	21.235064	10.0.2.15	10.0.2.255	NBNS	92			Name query NB UPAD-NB
4685	34.795240	10.0.2.15	10.0.2.255	NBNS	92			Name query NB UPAD-NB
4688	35.547170	10.0.2.15	10.0.2.255	NBNS	92			Name query NB UPAD-NB
4689	35.580123	10.0.2.15	10.0.2.255	NBNS	92			Name query NB UPAD-NB
3062	34.468000	10.0.2.15	118.67.248.41	HTTP	770			POST /src/redirect.php HTTP/1.1 (application/x-www-form-urlencoded)
3072	34.474031	117.18.237.29	10.0.2.15	OSCP	842			Response
3089	37.674085	117.18.237.29	10.0.2.15	OSCP	842			Response
4313	41.834270	117.18.237.29	10.0.2.15	OSCP	842			Response
4082	47.480213	117.18.237.29	10.0.2.15	OSCP	841			Response
64	1.841988	37.228.100.132	10.0.2.15	TLSv1	1474			Server Hello
770	34.612168	172.217.194.94	10.0.2.15	TLSv1	1472			Server Hello
2774	48.576633	172.217.194.94	10.0.2.15	TLSv1	1472			Server Hello
2882	48.928800	172.217.194.94	10.0.2.15	TLSv1	1472			Server Hello
133	8.628647	172.217.27.3	10.0.2.15	TLSv1.3	1472			Server Hello, Change Cipher Spec
533	11.523333	74.125.130.83	10.0.2.15	TLSv1.3	1472			Server Hello, Change Cipher Spec
539	11.524664	74.125.130.84	10.0.2.15	TLSv1.3	1472			Server Hello, Change Cipher Spec
3279	68.828312	172.217.194.94	10.0.2.15	TLSv1.3	1472			Server Hello, Change Cipher Spec
3283	68.828700	172.217.194.94	10.0.2.15	TLSv1.3	1472			Server Hello, Change Cipher Spec
4333	77.883284	74.125.130.84	10.0.2.15	TLSv1.3	1472			Server Hello, Change Cipher Spec
4370	77.928513	172.217.194.139	10.0.2.15	TLSv1.3	1472			Server Hello, Change Cipher Spec
4313	78.367820	74.125.24.138	10.0.2.15	TLSv1.3	1472			Server Hello, Change Cipher Spec
4862	139.488411	74.125.24.93	10.0.2.15	TLSv1.3	1472			Server Hello, Change Cipher Spec

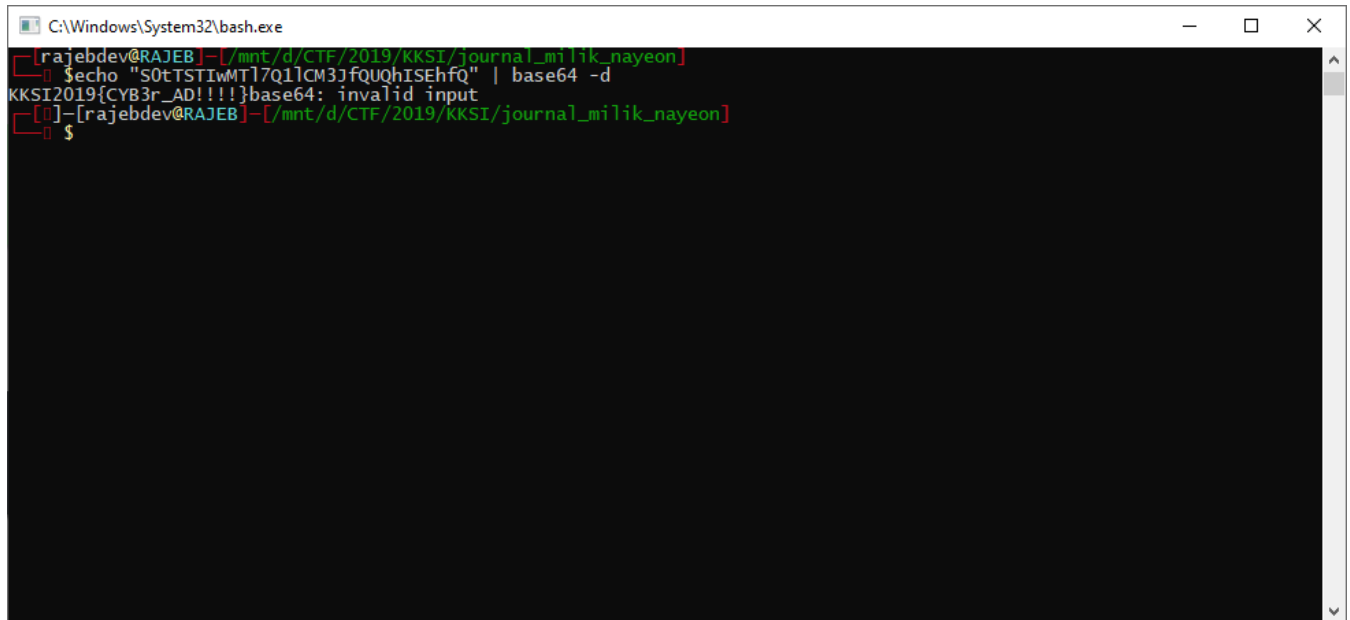
Dari situ kami membuka log pada baris tersebut maka didapatkan tampilan data form html sebagai berikut.

Wireshark · Packet 3062 · login_traffic.pcapng

- Frame 3062: 770 bytes on wire (6160 bits), 770 bytes captured (6160 bits) on interface 0
- Ethernet II, Src: PcsCompu_fe:21:ee (08:00:27:fe:21:ee), Dst: RealtekU_12:35:02 (52:54:00:12:35:02)
- Internet Protocol Version 4, Src: 10.0.2.15, Dst: 118.67.248.41
- Transmission Control Protocol, Src Port: 49532, Dst Port: 80, Seq: 1, Ack: 1, Len: 716
- Hypertext Transfer Protocol**
 - HTML Form URL Encoded: application/x-www-form-urlencoded
 - Form item: "js_autodetect_results" = "1"
 - Key: js_autodetect_results
 - Value: 1
 - Form item: "just_logged_in" = "1"
 - Key: just_logged_in
 - Value: 1
 - Form item: "login_username" = "user@user.com"
 - Key: login_username
 - Value: user@user.com
 - Form item: "secretkey" = "S0tTSTIwMT17Q1lCM3JfQUQhISEhfQ"
 - Key: secretkey
 - Value: S0tTSTIwMT17Q1lCM3JfQUQhISEhfQ

Terdapat Sebuah secretkey berupa base64 yang kami yakini sebagai flagnya. Maka setelah di

decode didapatkan flagnya.

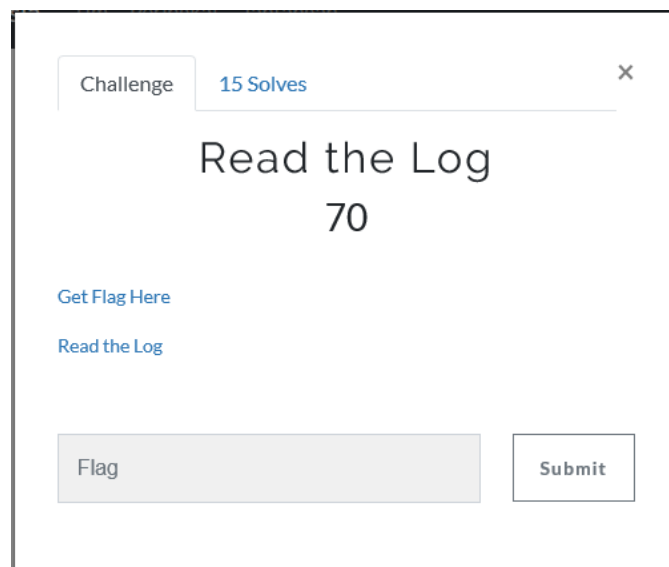


```
C:\Windows\System32\bash.exe
[rajobdev@RAJEB] - [mnt/d/CTF/2019/KKSI/journal_milik_nayeon]
$ echo "S0tTSTIwMTl7Q1lCM3JfQUQhISEhfQ" | base64 -d
KKSI2019{CYB3r_AD!!!!}base64: invalid input
[rajobdev@RAJEB] - [mnt/d/CTF/2019/KKSI/journal_milik_nayeon]
$
```

FLAG : KKSI2019{CYB3r_AD!!!!}

Read The Log [70 Point]

Disediakan soal sebagai berikut.



Challenge 15 Solves

Read the Log

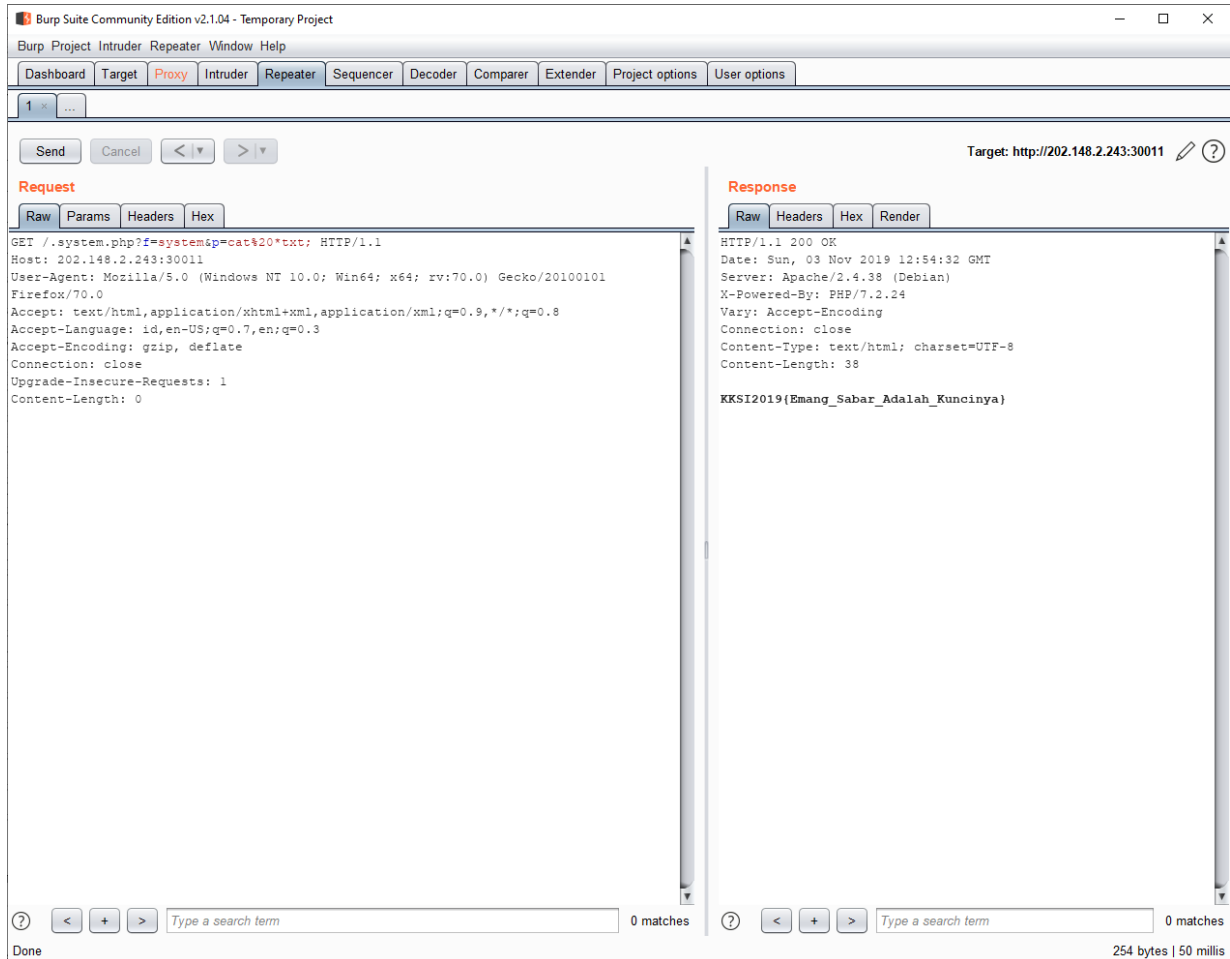
70

[Get Flag Here](#)

[Read the Log](#)

Flag

Terdapat file flag dengan nama sangat panjang. Kemudian kami melakukan cat terhadap flag tersebut dengan mengganti **id** menjadi **cat%20*txt**. Persen 20 berarti spasi. Maka kami mendapatkan flagnya.



FLAG : KKS12019{Emang_Sabar_Adalah_Kuncinya}

Member Have Journal [70 Point]

Disediakan soal berikut.

Description: 'camel' script its the key

[Download Here](#)



Beserta beberapa file dengan extensi journal.

- system.journal
- system@f7433012530a40e2a1ffbfd0fd517cb8-0000000000001f1b-00059436480e5d3d.journal
- user-1000.journal
- user-1000@1e1ff651682d49aebf6d0c2fca0bbc1f-0000000000001f2b-000594364811d83e.journal

Kami sempat bingung dengan file journal tersebut. Kemudian kami melakukan pencarian di google dan didapatkan jikalau file journal tersebut merupakan sebuah track record dari sebuah file system. Kemudian seperti biasa kami melakukan kebiasaan melakuka strings grep. Setelah berfikir agak keras, kami sadar biasanya jika ada soal dengan log file system flagnya tidak jauh dari user yang ada di dalam file system tersebut. Maka kami memutuskan untuk mencari dengan menggunakan strings dan grep. Pertama kami mencari strings yang ada dengan filter grep USER maka didapatkan hasil sebagai berikut.

```

[rajbdev@RAJEB]-[/mnt/d/CTF/2019/KKSI/journal_milik_nayeon]
$strings * | grep USER
_SYSTEMD_USER_SLICE=-.slice
_SYSTEMD_USER_SLICE
USER_ID=hasan
USER_ID
USERSPACE_USEC=13924626
USERSPACE_USEC
MESSAGE= hasan : TTY=tty1 ; PWD=/home/hasan ; USER=root ; COMMAND=/bin/su
USER_ID=hasan
USER_ID
USERSPACE_USEC=11469727
USERSPACE_USEC
_SYSTEMD_USER_SLICE=-.slice
_SYSTEMD_USER_SLICE
MESSAGE= hasan : TTY=tty1 ; PWD=/home/hasan ; USER=root ; COMMAND=/bin/su
USER_UNIT=default.target
USER_UNIT
USER_INVOCATION_ID=ee063f54dc874f928f1b1ee0e5edf572
USER_INVOCATION_ID
_SYSTEMD_USER_UNIT=init.scope
_SYSTEMD_USER_UNIT
_SYSTEMD_USER_SLICE=-.slice
_SYSTEMD_USER_SLICE

```

Pada gambar diatas terlihat bahwa ada user hasan. Kemudian kami melakukan strings grep dengan filter grep hasan. Maka didapatkan hasil sebagai berikut.

```

[rajbdev@RAJEB]-[/mnt/d/CTF/2019/KKSI/journal_milik_nayeon]
$strings * | grep hasan
MESSAGE=Stopping Session 1 of user hasan.
MESSAGE=Stopped Session 1 of user hasan.
MESSAGE=Removed slice User Slice of hasan.
USER_ID=hasan
MESSAGE=pam_unix(login:session): session opened for user hasan by LOGIN(uid=0)
MESSAGE=Created slice User Slice of hasan.
MESSAGE=pam_unix(systemd-user:session): session opened for user hasan by (uid=0)
MESSAGE=Started Session 1 of user hasan.
MESSAGE=New session 1 of user hasan.
MESSAGE= hasan : TTY=tty1 ; PWD=/home/hasan ; USER=root ; COMMAND=/bin/su
MESSAGE=pam_unix(sudo:session): session opened for user root by hasan(uid=0)
MESSAGE=pam_unix(su:session): session opened for user root by hasan(uid=0)
MESSAGE=Can't open perl script "/home/hasan/.2e3f3e17ebcb87baad8539475a1f91d41953c15": No such file or directory
LCMDLINE=/usr/bin/perl /home/hasan/.2e3f3e17ebcb87baad8539475a1f91d41953c15 8888
MESSAGE=Stopped Session 1 of user hasan.
USER_ID=hasan
MESSAGE=Removed slice User Slice of hasan.
MESSAGE=pam_unix(login:session): session opened for user hasan by LOGIN(uid=0)
MESSAGE=Created slice User Slice of hasan.
MESSAGE=Started Session 1 of user hasan.
MESSAGE=pam_unix(systemd-user:session): session opened for user hasan by (uid=0)
MESSAGE=New session 1 of user hasan.
MESSAGE= hasan : TTY=tty1 ; PWD=/home/hasan ; USER=root ; COMMAND=/bin/su
MESSAGE=pam_unix(sudo:session): session opened for user root by hasan(uid=0)
MESSAGE=pam_unix(su:session): session opened for user root by hasan(uid=0)
MESSAGE=pam_unix(systemd-user:session): session opened for user hasan
MESSAGE=pam_unix(systemd-user:session): session closed for user hasan
MESSAGE=pam_unix(sudo:auth): authentication failure; logname=hasan uid=1000 euid=0 tty=/dev/tty1 ruser=hasan rhost= user=hasan

```

Dari hasil strings grep diatas. Karena tidak ada hal yang jelas seperti password. Kami melihat ada beberapa string hexadecimal yang sama, kami menduga bahwa itu merupakan flagnya. Maka kami memutuskan bahwa string hexa tersebut merupakan flagnya. Setelah kami coba inputkan ternyata benar. Maka flagnya adalah.

FLAG : KKSI2019{2e3f3e17ebcb87baad8539475a1f91d41953c15}

Web

Tsunade Gambling Master [100 Point]

Disedikan soal berikut.

Challenge

16 Solves

×

Tsunade Gambling Master

100

http://202.148.2.243:20001

You have maximum input on this challenge 3 attempts!

Flag

Submit

File

Edit

View

History

Bookmarks

Tools

Help

Gacha Game

×

+

←

→

↻

🏠

🔍

Search with Google or en

You must reach 13333333337 -14

Go Away. Hus Hus

Tebakanmu: 7 Tebakan server: 38

Ayo diadu!

Ambil Flag

Disajikan sebuah web sederhana 'Gacha Game'. Kami melakukan analisis terhadap source code dari situs tersebut. Kami mendapatkan script JavaScript sebagai berikut:

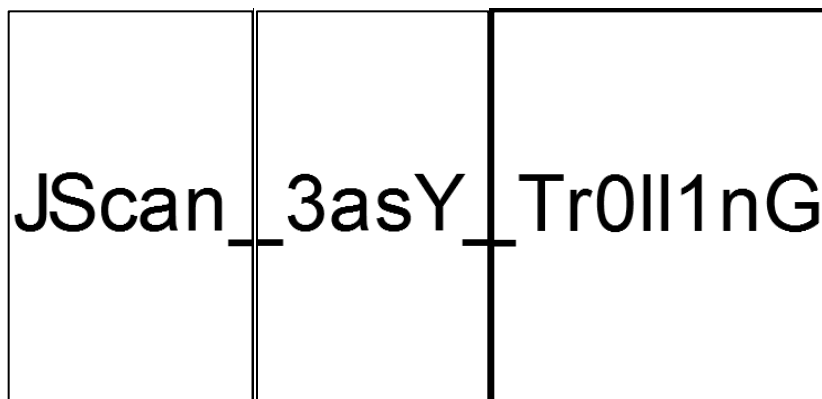
```

23 <script type="text/javascript">
24   //It's not flag! Don't Submit it
25   //I Warn you!
26   var kepla_flag="KKSII2019{"",place_flag="Tr0ll1ng_th3_Us3r",penutup="}";
27   function get_point_now(){
28     var t=$("#point").text();
29     return parseInt(t)
30   }
31   function generate_judi_server(t){
32     return Math.round(Math.random()*t)
33   }
34   function genertae_judi_client(){
35     return batas=generate_judi_server(100),Math.round(Math.random()*batas)
36   }
37   function ready_to_serve(){
38     return place_flag.split("_")
39   }
40   function serve(t){
41     var e=t;
42     for($i=0;$i<e.length;$i++)$("#flag"+$i).html("<img src='./fl4g/'+e[$i]+''.png'>")
43   }
44   $(document).on("click","#adu",()=>{var t=genertae_judi_client(),e=generate_judi_server(100);
45     $("#client").text(t),$("#server").text(e);
46     var n=get_point_now();t>e?($("#point").text(n+1):$("#point").text(n-1)),$(document).on("click","#judii",()=>{
47       get_point_now()>13333333337?(console.log("I know you inspect element it!"),$("#flag").text(place_flag+"
Don't Submit it Bratan! It's wrong one!")):$("#flag").text("Go Away. Hus Hus"));
48   })
49 </script>

```

Kami menganalisis code javascript diatas dan mendapati 2 function yang membuat kami curiga yaitu function ready_to_serve() dan serve(t). Pada function ready_to_serve() nilai kembali fungsi tersebut adalah place_flag.split('_') yang akan membagi place_flag="Tr0ll1ng_th3_Us3r" menjadi 3 elemen yaitu 'Tr0ll1ng', ' th3', ' Us3r'. Kemudian pada function serve(t) kami mendapati fungsi tersebut untuk membuat link yang menuju gambar dalam server sebagai berikut:

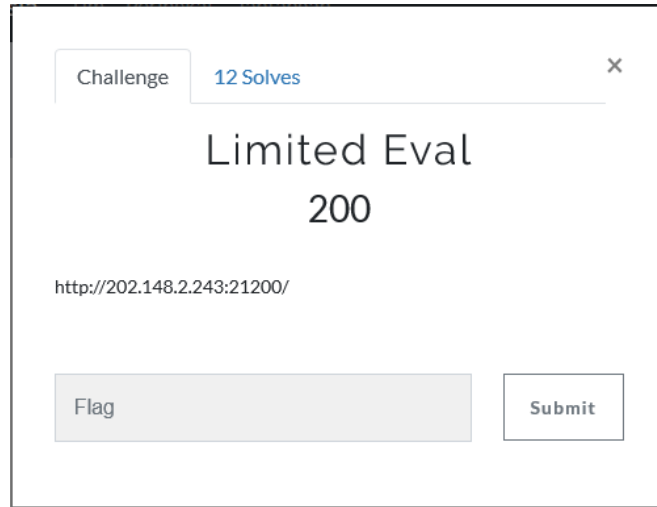
Ketika kami mencoba mengaksesnya hasilnya 3 gambar sebagai berikut:



FLAG : KKSII2019{JScan_3asY_Tr0ll1nG}

Limited Eval [200 Point]

Disedian soal berikut.



Challenge 12 Solves

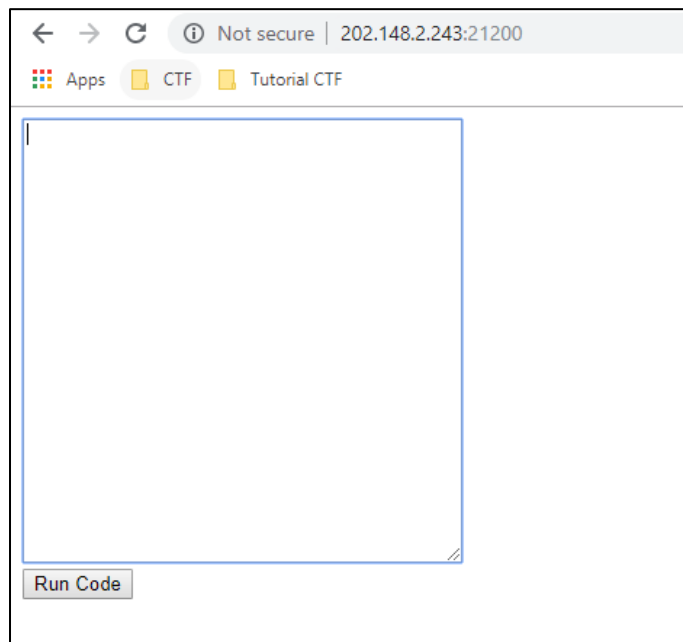
Limited Eval
200

<http://202.148.2.243:21200/>

Flag Submit

Description: <http://202.148.2.243:21200/>

Diberikan tampilan web sebagai berikut.



← → ↻ ⓘ Not secure | 202.148.2.243:21200

Apps CTF Tutorial CTF

Run Code


Menurut kami soal merupakan hint dasarnya, yaitu eval code dimana setiap kode yang kita input maka akan dijalankan seperti script php. Namun karena limited maka sudah pasti tidak

semua code bisa dijalankan. Kami mencoba mempelajari PHP eval code. Maka seperti biasa dilakukan percobaan `echo(1+2)` maka output akan muncul 3. Ya seperti dugaan. Kemudian kami mencoba menginputkan `phpinfo();` maka didapatkan hasil sebagai berikut.

The screenshot shows a web browser window with the address bar displaying '202.148.2.243:21200'. The page content is the output of the `phpinfo();` function, which displays the PHP version 7.2.24 and various system configuration details.

PHP Version 7.2.24	
System	Linux db13d190758c:4.4.0-131-generic #157-Ubuntu SMP Thu Jul 12 15:51:36 UTC 2018 x86_64
Build Date	Oct 25 2019 04:21:18
Configure Command	'/configure' '--build=x86_64-linux-gnu' '--with-config-file-path=/usr/local/etc/php' '--with-config-file-scan-dir=/usr/local/etc/php/conf.d' '--enable-option-checking=fatal' '--with-mhash' '--enable-ftp' '--enable-mbstring' '--enable-mysqlnd' '--with-password-argon2' '--with-sodium=shared' '--with-pdo-sqlite=lib' '--with-sqlite=lib' '--with-curl' '--with-ibmtdf' '--with-openssl' '--with-zlib' '--with-ibmtdf=libx86_64-linux-gnu' '--with-apr2' '--disable-cgi' 'build_alias=x86_64-linux-gnu'
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/usr/local/etc/php
Loaded Configuration File	/usr/local/etc/php.ini
Scan this dir for additional .ini files	/usr/local/etc/php/conf.d
Additional .ini files parsed	/usr/local/etc/php/conf.d/docker-php-ext-sodium.ini
PHP API	20170718
PHP Extension	20170718
Zend Extension	320170718
Zend Extension Build	API320170718.NTS
PHP Extension Build	API20170718.NTS
Debug Build	no
Thread Safety	disabled
Zend Signal Handling	enabled
Zend Memory Manager	enabled
Zend Multibyte Support	provided by mbstring

Dari situ kami banyak mengetahui apa saja fungsi yang di disable. Maka kami mencoba beberapa fungsi atau code yang biasa digunakan. Ternyata include bisa dijalankan. Maka kami mencoba melakukan include file online injection kami. Namun ternyata url dan include terlalu panjang. Otomatis kami berfikir keras, mencari di google referensinya. Kami mencari cara untuk membaca directory yang ada untuk melihat file apa sajakah. Kemudian kami menemukan cara menggunakan **scandir** namun ternyata scandir didisable juga. Kemudian kami mencoba adiknya yaitu `opendir`, ternyata tidak di block, maka harus ditemani oleh `readdir` untuk membaca dir yang diopen tadi. Kemudian kami menyusun code pendek sebagai berikut. Maka dihasilkan.

A screenshot of a web browser window. The address bar shows "202.148.2.243:21200/?flag". The browser has tabs for "Apps", "CTF", and "Tutorial CTF". The main content area is a code editor with the following PHP code:

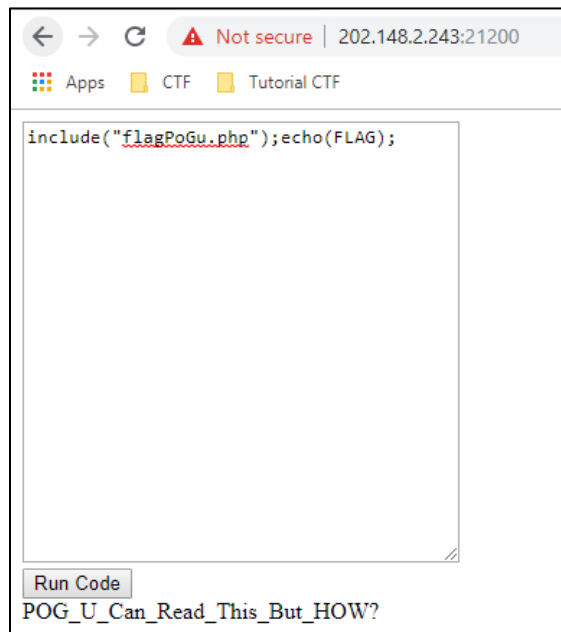
```
print(readdir(opendir(".")));
```

 Below the code editor is a "Run Code" button and the filename "flagPoGu.php".

```
print(readdir(opendir(".")));
```

Run Code
flagPoGu.php

Terlihat bahwa ada file flagPogu.php. Kemudian kami lakukan include dan melakukan echo(FLAG);. Hal ini kami lakukan melihat hampir semua code untuk memanggil sebuah FLAGnya. Maka didapatkan hasil.

A screenshot of a web browser window. The address bar shows "202.148.2.243:21200". The browser has tabs for "Apps", "CTF", and "Tutorial CTF". The main content area is a code editor with the following PHP code:

```
include("flagPoGu.php");echo(FLAG);
```

 Below the code editor is a "Run Code" button and the output "POG_U_Can_Read_This_But_HOW?".

```
include("flagPoGu.php");echo(FLAG);
```

Run Code
POG_U_Can_Read_This_But_HOW?

Maka didapatkan flagnya.

FLAG : KKS12019{POG_U_Can_Read_This_But_HOW?}