Writeup Cyber Defense Competition 2016



# Pwning The World #



[*CEDECE16*]

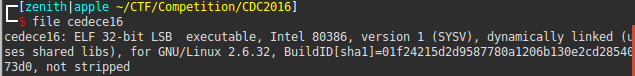
Laporan Capture The Flag

1. **Ringkasan Eksekutif**

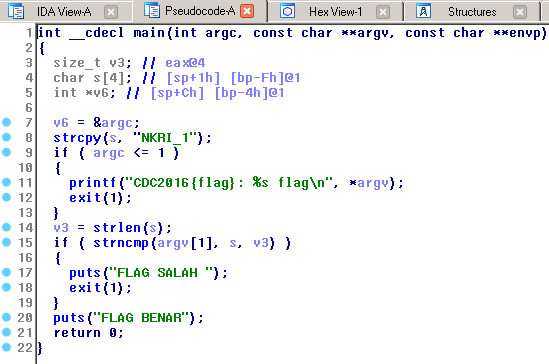
Reverse

1. **Laporan Teknis**

Diberikan sebuah file binary 32bit yang ketika dijalankan akan meminta sebuah inputan flag. Jika input salah, maka akan keluar output “FLAG SALAH”.



Buka dengan IDA 32 bit.



Terlihat jelas bahwa flag yang harus dimasukkan adalah “NKRI\_1”.

1. Kesimpulan

Flag : CDC2016{NKRI\_1}



[*bufow*]

1. **Ringkasan Eksklusif**

Si andi seorang pengangguran yang ingin melamar pekerjaan di sebuah perusahaan IT

tetapi dalam test melamar pekerjaan dia di berikan tantangan untuk mendapatkan sebuah pesan rahasia di server

Ayoo bantu Andi agar dapat di terima dan tidak menganggur !!!

Akses Server

IP : [128.199.181.69](http://l.facebook.com/l.php?u=http%3A%2F%2F128.199.181.69%2F&h=nAQHrKoOR) 2200

U : soldier

P : -!@#soldier!@#-

File soal terdapat di

/home/soal

File : bufow

Hint : Little Endian

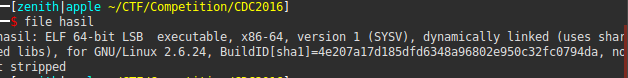
nb: Jika berhasil menemukan flag yang benar, submit dengan CDC2016{flag}

2. **Laporan teknis**

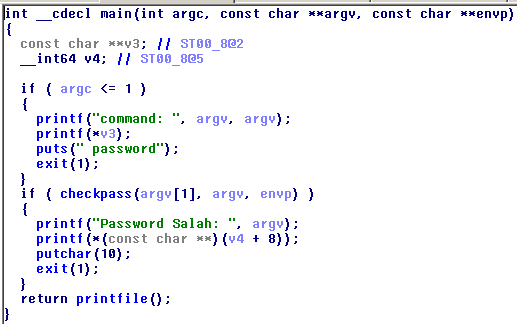
Diberikan akses ssh ke server. Soal terdapat di /home/soal. Di sana terdapat 2 file



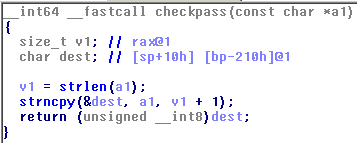
Encode base64 soal bufow agar dapat dikerjakan di local. File merupakan 64 bit elf.



Setelah dianalisa, terdapat fungsi menarik yaitu printfile(). Fungsi ini rupanya akan membuka flag.txt. Jadi tujuan kita adalah memanggil fungsi ini. Namun ternyata terdapat beberapa hal ketika dibuka dengan ida.



Jika mengikuti alur tersebut, tentunya kita tidak akan dapat memanggil printfile(). Ternyata fungsi checkpass() cukup menarik.



Kita bisa mengoverflow di bagian ini sehingga bisa memanggil printfile(). Setelah melakukan analisa dengan menggunakan gdb-peda, diketahui bahwa dibutuhkan 67\*8 bytes untuk mengoverwrite EIP, kemudian diarahkan ke fungsi printfile().

3. **Kesimpulan**

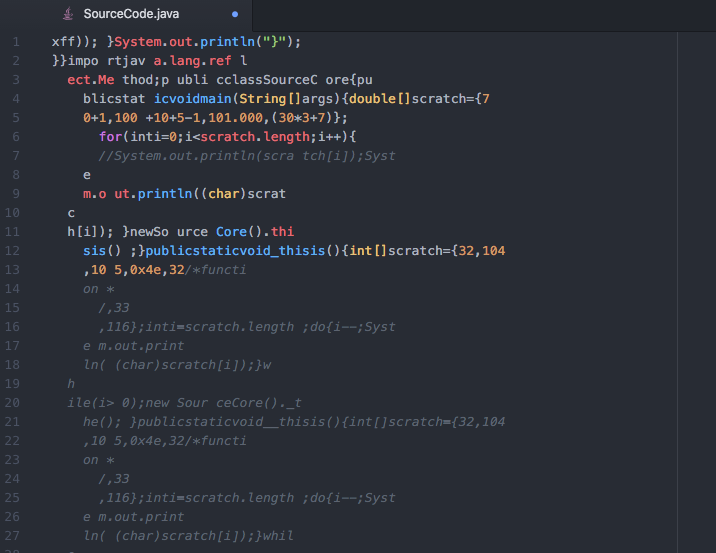
Flag : CDC2016{e45y-n33d-Coffe3}



[*SourceCore*]

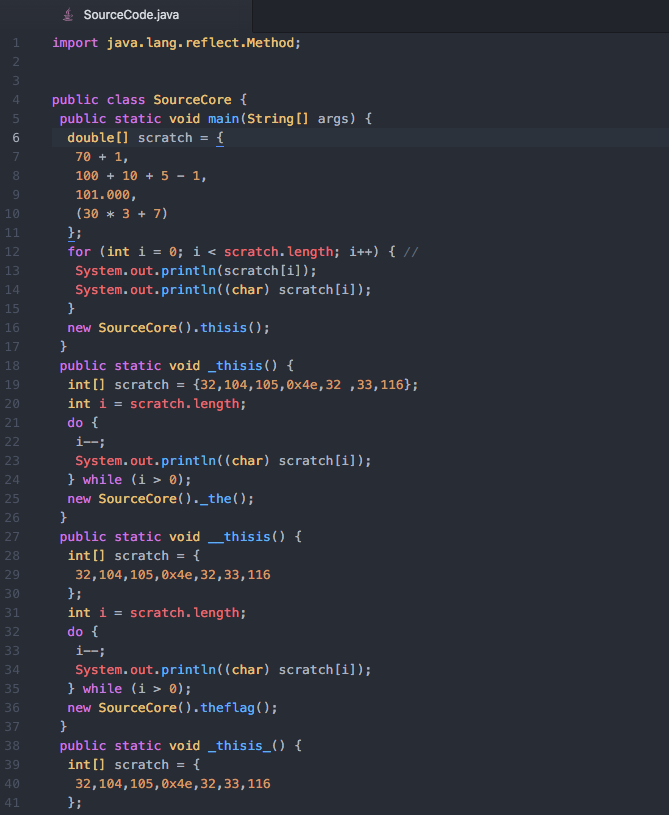
1. **Ringkasan Eksekutif**

Terdapat file SourceCode.java yang masih tidak beraturan sample nya ada dibawah ini



1. **Laporan Teknis**

Dari file SourceCode.java tersebut setelah diperbaiki, didapatkan source seperti dibawah ini



|  |
| --- |
| import java.lang.reflect.Method;  public class SourceCore {  public static void main(String[] args) {  double[] scratch = {  70 + 1,  100 + 10 + 5 - 1,  101.000,  (30 \* 3 + 7)  };  for (int i = 0; i < scratch.length; i++) { //  System.out.println(scratch[i]);  System.out.println((char) scratch[i]);  }  new SourceCore().thisis();  }  public static void \_thisis() {  int[] scratch = {32,104,105,0x4e,32 ,33,116};  int i = scratch.length;  do {  i--;  System.out.println((char) scratch[i]);  } while (i > 0);  new SourceCore().\_the();  }  public static void \_\_thisis() {  int[] scratch = {  32,104,105,0x4e,32,33,116  };  int i = scratch.length;  do {  i--;  System.out.println((char) scratch[i]);  } while (i > 0);  new SourceCore().theflag();  }  public static void \_thisis\_() {  int[] scratch = {  32,104,105,0x4e,32,33,116  };  int i = scratch.length;  do {  i--;  System.out.println((char) scratch[i]);  } while (i > 0);  new SourceCore().the();  }  public static void \_thisis\_\_() {  int[] scratch = {  32,104,105,0x4e,0x32,0x33,116  };  int i = scratch.length;  do {  i--;  System.out.println((char) scratch[i]);  } while (i > 0);  new SourceCore().thisistheflag();  }  public static void thisis() {  int[] scratch = {  32,104,105,0x4e,32,33,116  };  int i = scratch.length;  do {  i--;  System.out.println((char) scratch[i]);  } while (i > 0);  new SourceCore().the();  }  public static void the() {  int[] scratch = {  0x66,108,97,103,110,121,97,32,67,68,67,50,48,49,54,123,114,111  };  for (int i = 0; i < scratch.length; i++) { //  System.out.println(scratch[i] \* 3);  System.out.println((char) scratch[i]);  }  new SourceCore().flag();  }  public static void \_the() {  int[] scratch = {  0x66,1608,97,1403,1310,121,97,32,67,68,67,50,48,49,54,123,114,111  };  for (int i = 0; i < scratch.length; i++) { //  System.out.println(scratch[i] \* 3);  System.out.println((char) scratch[i]);  }  new SourceCore().theflag();  }  public static void theflag() {  int[] scratch = {  0x66,108,97,103,110,121,97,32,67,68,67,50,48,49,54,123,114,111  };  for (int i = 0; i < scratch.length; i++) { //  System.out.println(scratch[i] \* 3);  System.out.println((char) scratch[i]);  }  new SourceCore().flag();  }  public static void the\_flag() {  double[] scratch = {  0x66,108,97,103,0x110,0x121,97 / 3,32.000,67.00,68,67,50,48,49,54,123,114,111  };  for (int i = 0; i < scratch.length; i++) { //  System.out.println(scratch[i] \* 3);  System.out.println((char) scratch[i]);  }  new SourceCore().f\_lag();  }  public static void f\_lag() {  int[] scratch = {  2958550,44,293580,2472335,2902370,28305,29580,24735,29580,25755,29070,28305,29580,2473335,29580,29835,4323,2634265  };  for (int i = 0; i < scratch.length; i++) {  System.out.println(scratch[i]);  }  System.out.println("}");  }  public static void flags() {  int[] scratch = {  2958330,24735,29554580,24735,29070,28305,295380,24735,29580,25755,29070,28305,29580,24735,29580,29835,28050,26265  };  for (int i = 0; i < scratch.length; i++) {  System.out.println(scratch[i]);  }  System.out.println("}");  }  public static void thisistheflag() {  int[] scratch = {  29580,24735,29544580,2473335,2907220,2833305,2943580,24735,29580,25755,29070,28305,29580,24735,29580,29835,28342050,262634345  };  for (int i = 0; i < scratch.length; i++) {  System.out.println((char)(scratch[i]));  }  System.out.println("}");  }  public static void flag() {  int[] scratch = {  29580,24735,29580,4735,29070,28305,29580,24735,29580,25755,29070,28305,29580,24735,29580,29835,28050,26265  };  for (int i = 0; i < scratch.length; i++) {  System.out.println((char)(scratch[i] / 0xff));  }  System.out.println("}");  }} |

Setelah di compile dan execute didapatkan flagnya

1. **Kesimpulan**

Soal berhasil diselesaikan dengan cara membalikan menjadi source yang asli dari SourceCode.java yang tidak beraturan .

Flag : CDC2016{rotatrotaterotatung}



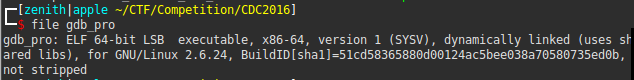
[gdb\_pro]

1. **Ringkasan Eksekutif**

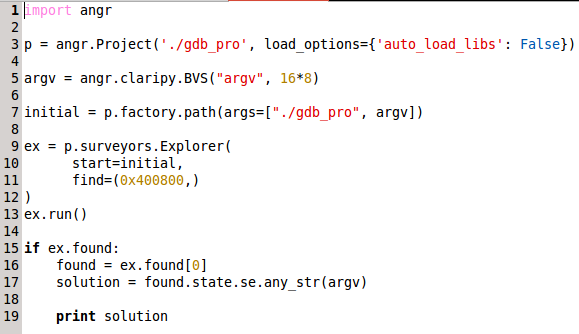
Isyana Andriana adalah seorang Reverse Engineer muda yang cantik nan jelita yang sedang magang di Perusahaan IT Security BeautifulEye, Inc. Ia ditugaskan, untuk mencari kelemahan binary yang merupakan lisensi produk terbaru GDB-Pro milik BeautifulEye. Bantulah ia untuk menemukan kelemahan lisensi produk GDB-Pro.

2. **Laporan Teknis**

Diberikan sebuah file binary 64 bit elf.



Kita diharuskan memasukkan jawaban yang benar, jika salah maka akan muncul tulisan You're not Reverse Engineer!. Jika benar, akan muncul tulisan Good job (:. Karena input diambil dari argv, maka dapat digunakan library angr dari python untuk menyelesaikan soal ini. Berikut ini merupakan script sederhana menggunakan library angr.



Tidak lama setelah script dijalankan, didapatkan hasilnya berupa jawaban yang sudah ditunggu - tunggu.

3. **Kesimpulan**

Flag : CDC2016{AES\_K3yExp4ns10n}



[*Error APK*]

1. **Ringkasan Eksekutif**

Omar adalah sebuah bandar narkoba yang memiliki jaringan sangat luas.

Untuk melancarkan proses distribusi narkoba yang dia miliki, dia menggunakan sebuah aplikasi khusus di androidnya.

Ketika kepolisian menangkapnya, ia berhasil merusak aplikasi yang ia miliki namun kepolisian berhasil mengamankan aplikasi rusak tersebut.

Tugas anda adalah membantu pihak kepolisian untuk memperolehnya dari aplikasi yang rusak ini. Jika berhasil ada Flag yang menanti anda.

Selamat bekerja!

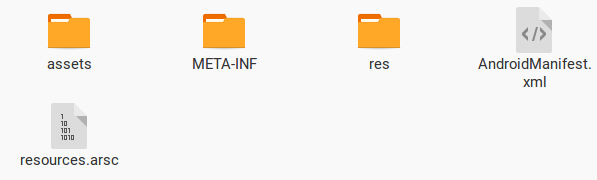
File

Clue:

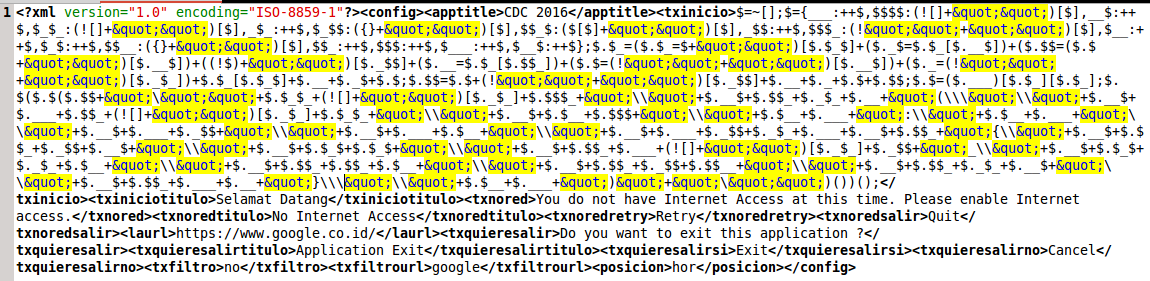
&quot; to "

2. **Laporan Teknis**

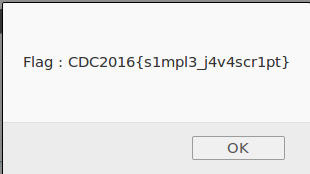
Diberikan sebuah file apk yang ketika diekstrak akan menghasilkan beberapa direktori.



Di dalam folder assets, terdapat file config.xml. Buka dengan text editor dan didapatkan hasil berikut.



Sesuai dengan hint dari soal, ubah &quot; menjadi “. Hasilnya adalah javascript obfuscate. Masukkan dalam console browser sehingga didapatkan hasil



3. **Kesimpulan**

Flag : CDC2016{s1mpl3\_j4v4scr1pt}



[*Ransomed*]

1. **Ringkasan Eksekutif**

""Dear professional hacker""

Website kami telah disusupi oleh seorang hacker yang tidak diketahui. Seluruh isi website telah dienkripsi dengan sebuah ransomware, dan meminta tebusan yang tidak sedikit.

Kami sudah berusaha menemukan hacker tersebut, namun tidak ada jejak.

Kami kesulitan untuk mengembalikan file kami. File tersebut berisi data rahasia sangat penting.

Administrator kami adalah orang yang sangat waspada, namun saat ini sedang dalam tugas ke jepang untuk membantu ultramen.

Sehingga tidak bisa membantuk kami.

Tolong bantuk kami mengembalikan file kami ke bentuk semula.

Website kami bisa ditemukan di sini:

62.4.3.124

Berikut kami berikan login ssh:

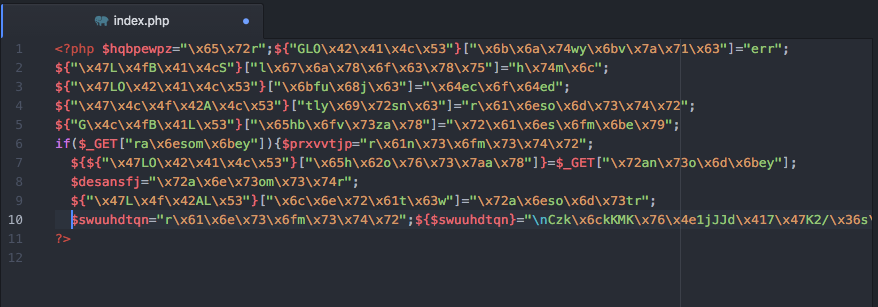
PORT: 22

USER: cdc2016

PASS: nkrihargamati

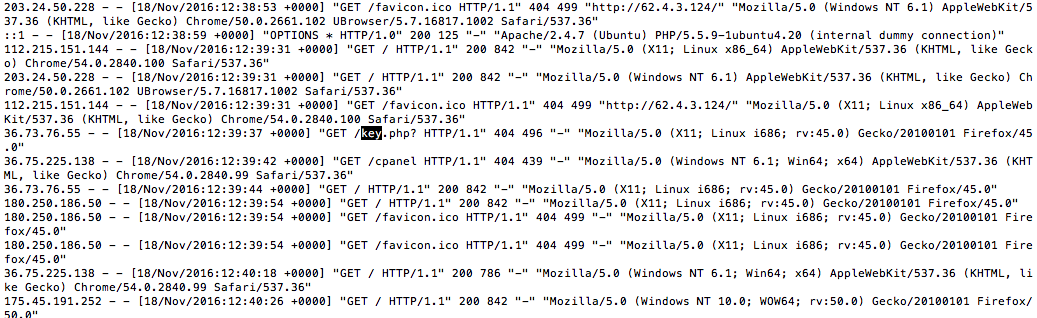
2. **Laporan Teknis**

Dari deskripsi soal diberikan akses ssh ke IP 62.4.3.124, setelah login didapatkan source code web index.php yang masih tidak beraturan dan berbentuk hexcode

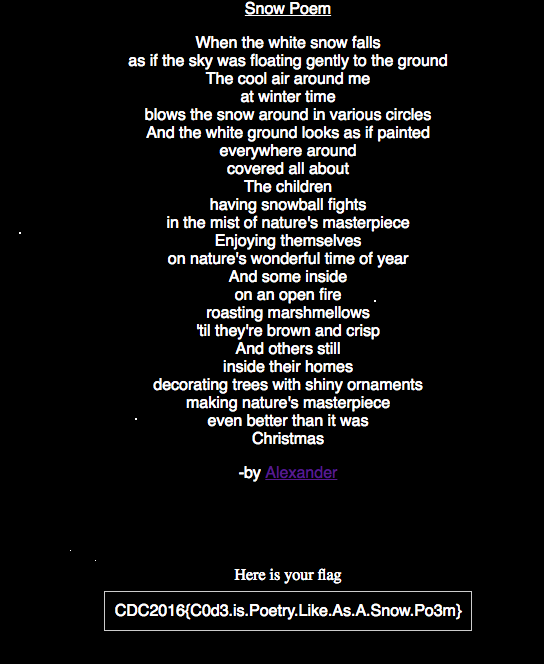


Namun file itu hanya tipuan, ada cara yang lebih efektif yaitu dengan mengakses log apache2 dengan command dibawah ini

|  |
| --- |
| $ less /var/log/apache2/access.log |



Di log tersebut ada ransomkey yang digunakan untuk di input pada web, namun pada saat pembuat write up ini ternyata log file GET ransomkey ada yang menghapus jadi kami tidak bisa menampilkannya, namun jika sudah mendapatkan key tersebut setelah di input maka terlihat seperti dibawah ini



3. **Kesimpulan**

Flag : CDC2016{C0d3.is.Poetry.Like.As.A.Snow.Po3m}



[Buggy]

1. **Ringkasan Eksekutif**

"Organisasi rahasia "Kucing 4 Community" akhirnya berhasil masuk ke dalam sistem bank dunia lewat malware dan mendapatkan sebuah program kecil.

Program kecil ini dinamakan X-Key-Protostar-s3cret! dan disamarkan menjadi nama BuggyCDC2016. Walaupun disamarkan, organisasi ini tau bahwa program ini menyimpan kunci utama untuk seluruh encrypted communication yang digunakan oleh bank dunia. Terdapat password kecil untuk membuka kunci tersebut. Organisasi ini bekerja keras siang malam untuk membypass login tersebut dan berhasil.

Pimpinan meyakini bahwa kunci utama tersebut tersimpan secara tersembunyi di baris kode program yang mungkin tidak tereksekusi ketika program dijalankan.

Kamu sebagai anggota baru dari organisasi Kucing 4 diberikan tugas untuk mengambil kunci tersebut. Apabila kamu berhasil, maka kamu akan diberikan uang serta kehormatan yang tinggi bagi komunitas

Sudah diuji di:

1) Windows XP 32-bit

2) Windows 8 Pro 64-bit

2. **Laporan Teknis**

Diberikan file binary 64 bit elf. Setelah dianalisa, terdapat fungsi yang menarik yaitu uselessFunction().



Nampaknya, di sinilah terdapat flag yang sesungguhnya. Dengan mudah, dapat dibuat fungsi yang sama persis dengan fungsi tersebut, sehingga didapatkan flag.

3. **Kesimpulan**

Flag : CDC2016{0fc6d3de27f2bc91e775ca708c391d49}