



## CLAYDAY BLOG



JUNE 9, 2017 / CTF

## QUADRATHLON 2017 WRITEUP – PART 1

## Hello – Trivia 1pt

---

Selamat datang di Quadrathlon 2017~~

Format flag adalah : noob{flag\_format}. Silahkan dicoba submit :)

Nothing special, tinggal masukan contoh flag.

**Flag : noob{flag\_format}**

## NYN 1.0 – Trivia 10pts

---

Now You Know 1.0

Suatu cara untuk mencari tahu apakah file yang dikirim dan yang diterima dalam transmisi data tetap sama atau tidak (flag dalam huruf kecil semua)

**Flag : noob{checksum}**

## NYN 2.0 – Trivia 10pts

---

Now You Know 2.0

Nama kelompok peretas yang berhasil melakukan dumping tools hacking milik NSA yang kemudian dimanfaatkan seseorang/sekelompok yang lain untuk membuat WannaCry (flag dalam huruf kecil semua)

**Flag : noob[shadowbroker]**

## NYN 3.0 – Trivia 10pts

---

Now You Know 3.0

Nama domain yang digunakan WannaCry sebagai kill-switch

**Flag : noobliuqerfsodp9ifjaposdfjhgosurijfaewrwergwea.com!**

## NYN 4.0 – Trivia 10pts

---

Now You Know 4.0

Nomor port yang dimanfaatkan WannaCry

**Flag : noob[445]**

## Bas[el]ic – Crypto 50pts

---

Decrypt me please. Hehe

Setelah link pada soal dibuka. Ternyata muncul text alphanumeric yang sangat panjang. Lalu dilihat dari ciri-cirinya (diakhiri tanda = ), kemungkinan besar text tersebut adalah hasil enkripsi base64. Coba di decrypt lalu didapatkan



*Flag : noob{DIAMMM!}*

## Puter – Crypto 25pts

---

abbo{qnygbx\_jrf\_vxv\_synt\_r}

Ini adalah kriptografi dengan rot13. Metode ini mengkonfersi setiap string (misalkan) X menjadi  $(X+13) \bmod Z$ . Atau sama dengan di shift 13x dimana dalam shifting tersebut apabila melewati Z, maka kembali ke A.

*Flag : noob{daltok\_wes\_iki\_flag\_e}*

## Kembaran – Crypto 100pts

---

Shakijem menemukan sebuah peti harta karun. Di dalam peti itu terdapat sebuah surat yang mengatakan "Pecahkan misteri yang ada di sini untuk mendapatkan harta karun". Bantu Shakijem untuk memecahkan misteri tersebut!

Halaman yang ada di soal merujuk pada sebuah halaman yang berisi form untuk mengirimkan 2 file ke server. File tersebut meminta 2 file yang kembar. Berarti kita harus mengirimkan 2 file yang identik ke server. Dimana identik disini bukan berarti file yang sama persis.

Sedangkan untuk mengetahui apakah 2 file adalah file yang sama, kita harus melakukan checksum (sudah ada di soal trivia kan, hehe). Checksum dilakukan dengan mengenkripsi setiap byte yang ada di file dengan suatu algoritma kriptografi modern, misalkan MD5, SHA1, SHA256, dsb. Dengan digunakannya algoritma modern, apabila 2 file memiliki minimal 1 byte yang berbeda, akan menghasilkan checksum yang sangat berbeda di antara keduanya.

Namun bagaimana bisa 2 file yang tidak sama memiliki checksum yang sama persis? Hal inilah yang disebut *collision*, yaitu ketika 2 plain text yang berbeda lalu dienkripsi dengan algoritma yang sama, menghasilkan encrypted text yang sama.

SHA1 sendiri adalah salah satu algoritma yang paling terakhir ditemukan collision nya. Jadi soal ini bertujuan untuk membuat 2 file yang memiliki colliding SHA1 checksum. Bagaimana caranya? Sudah banyak bertebaran online tool untuk men-generate 2 file yang collide. Saya menggunakan website [ini](#). Cukup cari 2 file jpg yang memiliki ukuran lebar dan tinggi yang sama. Lalu akan di generate 2 pdf yang memiliki checksum yang sama. Lalu submit kedua pdf itu.

**Flag : noob/sha1\_now\_ezly\_collided/**

## Kasmaran – Crypto 175pts

---

Via menerima sebuah surat aneh di depan pintu rumahnya. Yang tertulis disana sangatlah aneh, kecuali nama pengirim yang bernama Kley. Akhirnya Via yang penasaran memintamu untuk mencari tahu isi sebenarnya surat misterius tersebut

Yang ada di dalam attachment soal ini adalah 2 file, yaitu file python dan file yang berekstensi 'out'. Dimana script python tersebut melakukan sebuah enkripsi pada setiap byte dalam file **uoy\_evol** dan menghasilkan output berupa **uoy\_evol.out**.

```
import sys, itertools

if(len(sys.argv) != 3):
    print("Cara Pemakaian: [FILE] [KEY]")
    exit(-1)

nama = sys.argv[1]
kunci = sys.argv[2]
```

```
with open(nama, 'rb') as i:
    raw = i.read()
    print(len(raw))
    with open(nama + '.out', 'wb') as jadi:
        for l, r in zip(raw, itertools.cycle(kunci)):
            jadi.write(chr(ord(l) ^ ord(r)))
```

Script ini menerima 2 input melalui argumen, yang pertama adalah nama file yang akan di enkripsi, yang kedua adalah sebuah key yang digunakan untuk mengenkripsi file tersebut. Proses enkripsi file terdapat di iterasi for. Skrip tersebut melakukan XOR di antara byte file dan string key. Jadi misalkan isi file adalah ABCDEFGH, dan di enkripsi menggunakan key NM, maka yang terjadi :

```
A ^ N = ...
B ^ M = ...
C ^ N = ...
D ^ M = ...
.....
dan seterusnya hingga H
```

Berarti untuk melakukan decrypt, kita harus terlebih dulu mengetahui kunci apa yang dimasukkan pembuat soal untuk mendekripsi file aslinya. Karena enkripsi ini dilakukan dengan XOR, maka kita bisa menggunakan [xortool](#).

```

x - Terminal File Edit View Search Terminal Help
clayday@prnslc:~/Documents/CTF/Soal/TCyber/crypt1$ xortool uoy_evol.out
The most probable key lengths:
 2: 11.1%
 4: 12.1%
 6: 16.4%
 8: 9.3%
12: 13.1%
14: 8.0%
18: 9.5%
20: 6.4%
24: 7.7%
30: 6.3%
Key-length can be 3*n
Most possible char is needed to guess the key!
clayday@prnslc:~/Documents/CTF/Soal/TCyber/crypt1$ xortool -l 6 -o uoy_evol.out
200 possible key(s) of length 6:
\x1c\x08\x17\x1d\x10\x1e
]\x08\x17\x1d\x10\x1e
\x1d\t\x16\x1c\x11\x1f
\\t\x16\x1c\x11\x1f
\x1e\n\x15\x1f\x12\x1c
...
Found 88 plaintexts with 95.0%+ printable characters
See files filename-key.csv, filename-char_used-perc_printable.csv
clayday@prnslc:~/Documents/CTF/Soal/TCyber/crypt1$

```

Penjelasan :

```

xortool [namafile] : menebak berapa panjang key yang paling memungkinkan
xortool -l [panjang] : mencari key dengan inputan panjang key
xortool -o : bruteforce dengan printable character yang memungkinkan

```

```

x - Terminal File Edit View Search Terminal Help
clayday@prnslc:~/Documents/CTF/Soal/TCyber/crypt1$ cat xortool_out/filename-key.csv
file_name;key_repr
xortool_out/000.out;\x1c\x08\x17\x1d\x10\x1e
xortool_out/001.out;\x08\x17\x1d\x10\x1e
xortool_out/002.out;\x1d\t\x16\x1c\x11\x1f
xortool_out/003.out;\t\t\x16\x1c\x11\x1f
xortool_out/004.out;\x1e\n\x15\x1f\x12\x1c
xortool_out/005.out;\n\x15\x1f\x12\x1c
xortool_out/006.out;\x1f\x0b\x14\x1e\x13\x1d
xortool_out/007.out;\x0b\x14\x1e\x13\x1d
xortool_out/008.out;\x18\x0c\x13\x19\x14\x1a
xortool_out/009.out;\x0c\x13\x19\x14\x1a
xortool_out/010.out;\x19\r\x12\x18\x15\x1b
xortool_out/011.out;\x12\x18\x15\x1b
xortool_out/012.out;\x1a\x0e\x11\x1b\x16\x18
xortool_out/013.out;\x0e\x11\x1b\x16\x18
xortool_out/014.out;\x1b\x0f\x10\x1a\x17\x19
xortool_out/015.out;\x0f\x10\x1a\x17\x19
xortool_out/016.out;\x14\x00\x1f\x15\x18\x16
xortool_out/017.out;\x00\x1f\x15\x18\x16
xortool_out/018.out;\x15\x01\x1e\x14\x19\x17
xortool_out/019.out;\x01\x1e\x14\x19\x17
xortool_out/020.out;MYFLAO
xortool_out/021.out;\x0cYFLAO
xortool_out/022.out;NZE0BL
xortool_out/023.out;\x0fZE0BL
xortool_out/024.out;0[DNCM
xortool_out/025.out;\x0e[DNCM
xortool_out/026.out;H\CIDJ
xortool_out/027.out;\t\CIDJ
xortool_out/028.out;IJBHEK
xortool_out/029.out;\x08JBHEK
xortool_out/030.out;J^AKFH
xortool_out/031.out;\x0b^AKFH
xortool_out/032.out;K_JGI
xortool_out/033.out;\n_JGI
xortool_out/034.out;DPOEHF
xortool_out/035.out;\x05POEHF
xortool_out/036.out;FONDIG

```

Kunci yang paling memungkinkan adalah MYFLAO, coba didekripsi dengan key tersebut

```

x - Terminal File Edit View Search Terminal Help
clayday@prnslc:~/Documents/CTF/Soal/TCyber/crypt1$ xortool-xor -s "MYFLAO" -f uoy_evol.out
Dalam(derap(langkihmu hiri
Keebali dorong%lorono kenafgan bmraksa
Kita(seiraea menohituno batu%batu jisu
Kata bezceriti tenting waktu yafg peroi
Dan(musim(yang jerlal)

Di `ati misih ala tanqa ingan lepis
Dara jendmla-jefdela qang tmrtutux
Mengantip kelah-kelah celam
foob{p}isi_xanta}
Easihkih ada(sepenogal hirapan[?]erslepan da sana&

Hara ini eestinqa kucitat
Da sinidah peztenuaf itu jerjada
Bukaf untuc bersmkutu
Jukan xula uftuk bmrperafg
Tapa Cuma(menggggyangkin lonkeng kmnangaf
Biar(bergeea denjanganqa memjelah ounung[?]emanjul dazi lemjah ke(lemba`
Kabazkan pida alim kerang kezontano
Di sanilah(kita jersat} lagi[?]
clayday@prnslc:~/Documents/CTF/Soal/TCyber/crypt1$

```

Terbaca, namun masih ada yang aneh dengan hasilnya. Coba dengan key MYFLAG



```
clayday@prnslc:~/Documents/CTF/Soal/TCyber/crypt1$ xortool-xor -s "MYFLAG" -f uoy_evol.out
Dalam derap langkahmu hari
Kembali lorong-lorong kenangan bersaksi
Kita seirama menghitung batu-batu bisu
Kita bercerita tentang waktu yang pergi
Dan musim yang berlalu

Di hati masih ada tanya ingin lepas
Dari jendela-jendela yang tertutup
Mengintip celah-celah kelim
noob{puisi_xinta}
Masihkah ada sepenggal harapan
Tersimpan di sana.

Hari ini mestinya kucatat
Di sinilah pertemuan itu terjadi
Bukan untuk bersekutu
Bukan pula untuk berperang
Tapi Cuma menggoyangkan lonceng kenangan
Biar bergema dentangnya membelah gunung
Memantul dari lembah ke lembah
Kabarkan pada alam kering kerontang
Di sinilah kita bersatu lagi

clayday@prnslc:~/Documents/CTF/Soal/TCyber/crypt1$
```

**Flag : noob{puisi\_xinta}**

## PREVIOUS POST

QUADRATHLON 2017 WRITEUP – PART 2

## NO NEWER POSTS

RETURN TO BLOG

## CATEGORIES

CTF

## BE FIRST TO COMMENT

Leave a Reply



Your email address will not be published. Required fields are marked \*

Enter Your Comment...

Your Name\*

Your Email\*

Your URL

Post Comment

# CLAYDAY BLOG

## ARCHIVES

JUNE 2017

MAY 2017

## LIFE