セキュリティ競技CTFって何？
～CTFを通じて楽しくセキュリティとふれ合おう～

OSC京都ローカルスタッフ有志

もくじ

- CTFの概要
- DEMO1 (フォレンジック)
- DEMO2 (ネットワーク)
- CTFに興味が湧いた人へ

セキュリティのイメージ

学校での講義

情報セキュリティ

創作の世界

アニメや映画とかのハッカー!?

社会的な事件

企業の情報漏洩

難易度

難しそう・・・

情報セキュリティに必要なこと

コンピュータの基礎知識全般

- ネットワーク
- オペレーティングシステム
- ファイルシステム
- Webアプリケーション
- データベース
- プログラミング
- 数学

などなど…

情報セキュリティに必要なこと

コンピ
-
-
-
- ネ
- データ
- プログ
- 数学

などなど…

多すぎ
＼(^o^)／オワタ

学生の声

A「何を勉強したらいいかわからない・・・」
B「勉強するだけではおもしろくなくて続かない・・・」
C「学んだ技術を活かしたいが・・・」

学生の声

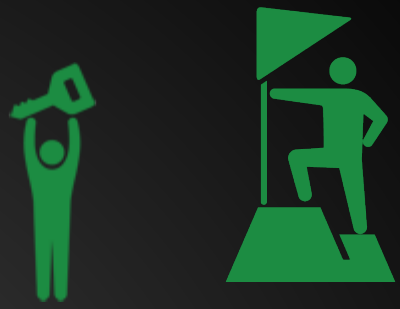
A「何を勉強したらいいかわからない・・・」
B「勉強するだけではおもしろくなくて続かない・・・」
C「学んだ技術を活かしたいが・・・」

CTFを足がかりに
情報セキュリティを
勉強しよう!!

CTF??

CTF？みんなやってるよ！
えっ、お金だって？無料だから、
試しにやってごらん？怖くないって。
お兄さんもやってるし、
参加している大学生とかも多いんだ！

CTFとは？



Capture The Flag（旗取りゲーム）の略
情報セキュリティの技術を競う競技・ゲーム

隠された答え（Flag）をセキュリティのスキルを用いて探し、答えをサーバへ送信するクイズ形式が多い

基本ルール

制限時間以内に得点を多く獲得したチームが勝利

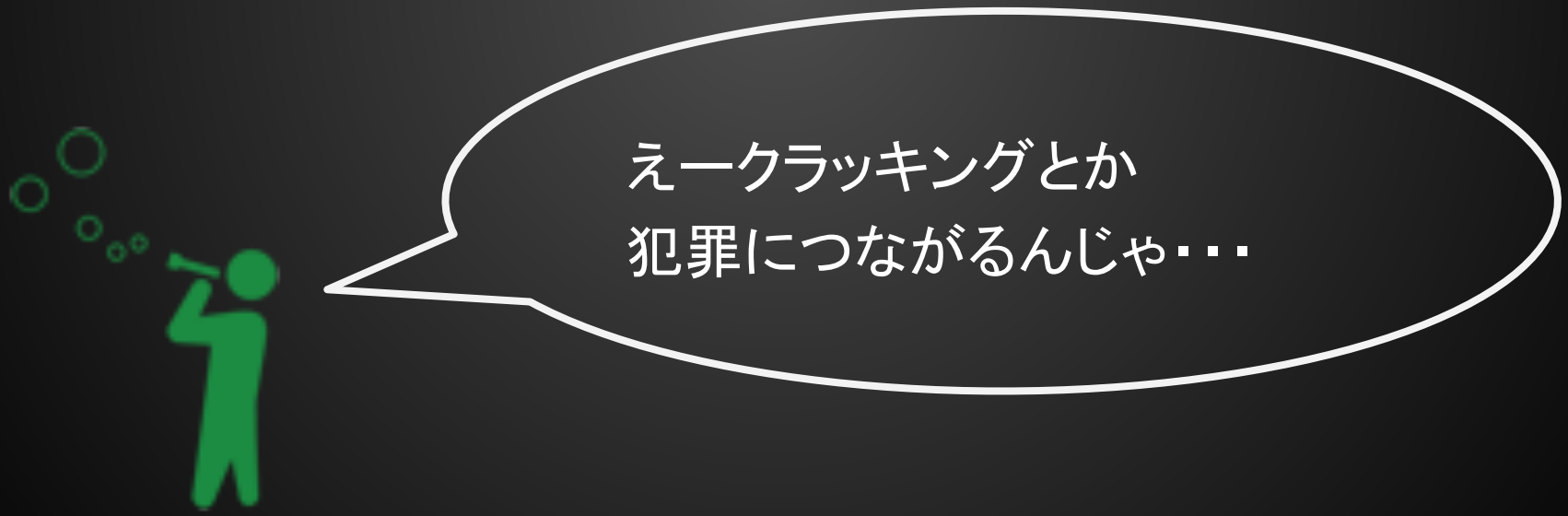
- 制限時間：だいたい12h~48h(大会による)
- チームメンバー制限:なし
- 検索:オンライン，オフライン問わず可

禁止事項

- 競技時間内の他チームとのフラグ・解法の共有

具体的には何をするの？

得られた知識は
どこで役に立つの？



主な出題分野

- リバースエンジニアリング
- ネットワーク
- フォレンジクス
- Pwnable(脆弱性調査)
- Web
- その他
  - 暗号
  - プログラミング

リバースエンジニアリング

ソースコードのないプログラムを解析する
デバッガによる実行
機械語からアセンブリ言語へ変換

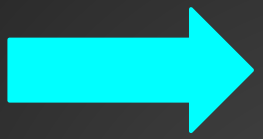
例えばマルウェア解析
悪意のあるプログラムの動作を特定する

```c
#include <stdio.h>

int main()
{
 printf("Hello World\n");

return 0;

}
```

hello.c

ソースコード

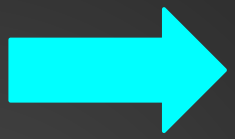
コンパイル



a.out

実行ファイル

デバッグ!!



```
root@kali:~/Desktop# gdb a.out
GNU gdb (GDB) 7.4.1-debian
Copyright (C) 2012 Free Software F
License GPLv3+: GNU GPL version 3
This is free software: you are fre
There is NO WARRANTY, to the exten
and "show warranty" for details.
This GDB was configured as "x86_64
For bug reporting instructions, pl
<http://www.gnu.org/software/gdb/b
Reading symbols from /root/Desktop
(gdb)
```

逆アセンブル!!

```
0000000000040050c <main>:
  40050c:       55                      push   rbp
  40050d:       48 89 e5                mov    rbp,rsp
  400510:       bf dc 05 40 00          mov    edi,0x4005dc
  400515:       e8 c6 fe ff ff          call   4003e0 <puts@plt>
  40051a:       b8 00 00 00 00          mov    eax,0x0
  40051f:       5d                      pop    rbp
  400520:       c3                      ret
  400521:       90                      nop
  400522:       90                      nop
```

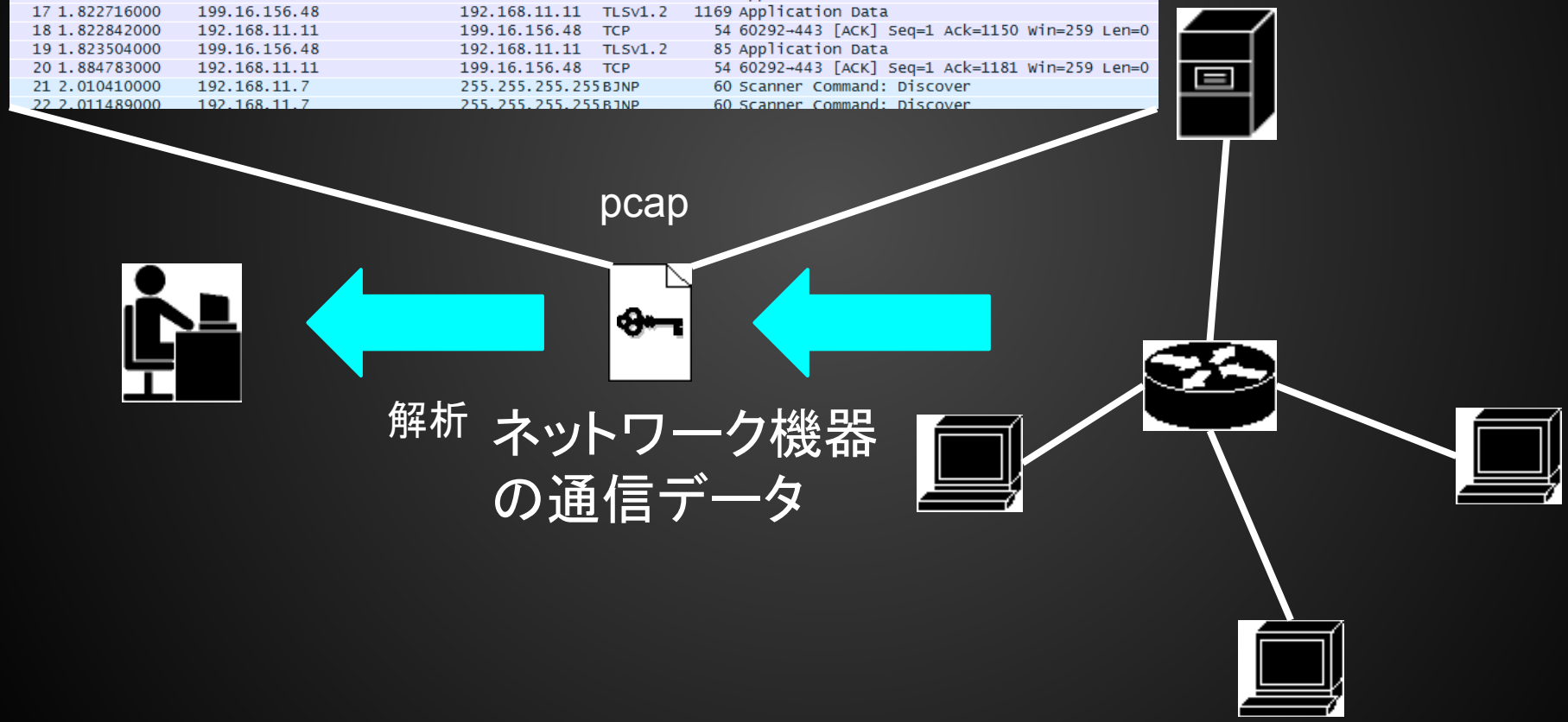
アセンブリ言語

ネットワーク

ネットワークを流れるトラフィックを
キャプチャして、それを分析・解析する

例えばネットワーク管理
ネットワーク機器の設定に必要な知識
ネットワークを理解することで
トラブル発生時に、原因調査ができる！

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	Micro-St_1e:56:9d	Broadcast	ARP	60	Who has 192.168.11.4? Tell 192.168.11.7
2	0.426493000	192.168.11.100	239.255.255.250	SSDP	388	NOTIFY * HTTP/1.1
3	0.752725000	199.16.156.48	192.168.11.11	TLSv1.2	86	Application Data
4	0.753368000	199.16.156.48	192.168.11.11	TLSv1.2	91	Application Data
5	0.753472000	192.168.11.11	199.16.156.48	TCP	54	60285→443 [ACK] Seq=1 Ack=70 Win=257 Len=0
6	0.754027000	199.16.156.48	192.168.11.11	TLSv1.2	85	Application Data
7	0.791596000	192.168.11.7	255.255.255.255	DB-LSP-[	166	Dropbox LAN sync Discovery Protocol
8	0.793855000	192.168.11.7	192.168.11.255	DB-LSP-[	166	Dropbox LAN sync Discovery Protocol
9	0.795741000	192.168.11.7	255.255.255.255	DB-LSP-[	166	Dropbox LAN sync Discovery Protocol
10	0.809261000	192.168.11.11	199.16.156.48	TCP	54	60285→443 [ACK] Seq=1 Ack=101 Win=257 Len=0
11	0.999932000	Micro-St_1e:56:9d	Broadcast	ARP	60	Who has 192.168.11.4? Tell 192.168.11.7
12	1.099289000	192.168.11.100	239.255.255.250	SSDP	323	NOTIFY * HTTP/1.1
13	1.133594000	192.168.11.100	239.255.255.250	SSDP	378	NOTIFY * HTTP/1.1
14	1.211036000	192.168.11.100	239.255.255.250	SSDP	314	NOTIFY * HTTP/1.1
15	1.515997000	fe80::9946:c375:2692:15c8	ff02::c	SSDP	208	M-SEARCH * HTTP/1.1
16	1.821511000	199.16.156.48	192.168.11.11	TLSv1.2	88	Application Data
17	1.822716000	199.16.156.48	192.168.11.11	TLSv1.2	1169	Application Data
18	1.822842000	192.168.11.11	199.16.156.48	TCP	54	60292→443 [ACK] Seq=1 Ack=1150 Win=259 Len=0
19	1.823504000	199.16.156.48	192.168.11.11	TLSv1.2	85	Application Data
20	1.884783000	192.168.11.11	199.16.156.48	TCP	54	60292→443 [ACK] Seq=1 Ack=1181 Win=259 Len=0
21	2.010410000	192.168.11.7	255.255.255.255	BJNP	60	Scanner Command: Discover
22	2.011489000	192.168.11.7	255.255.255.255	BJNP	60	Scanner Command: Discover



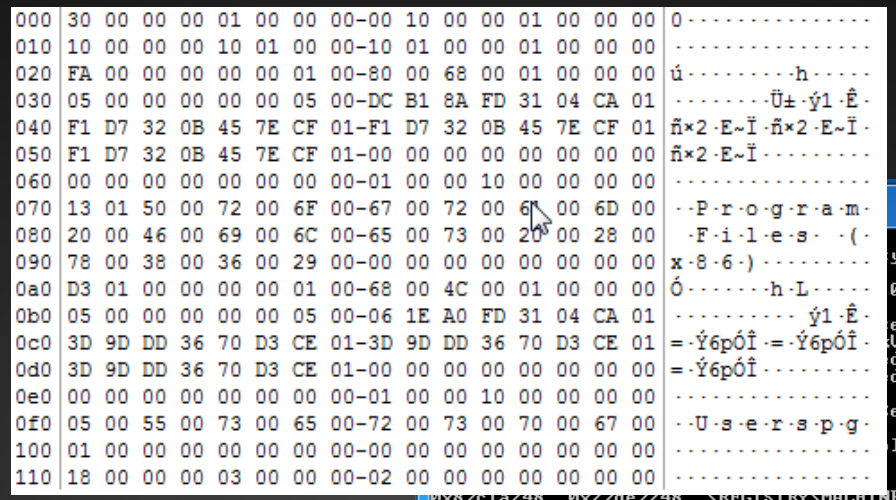
フォレンジクス

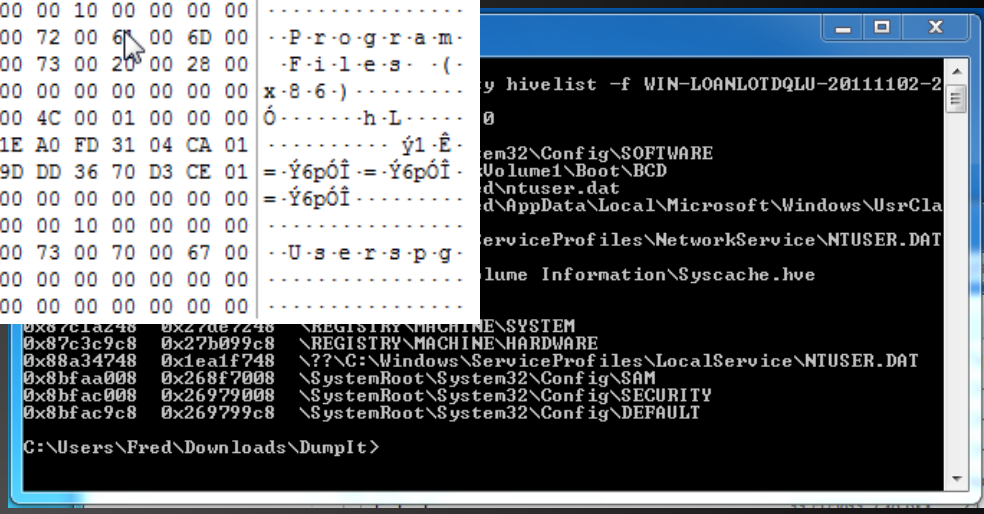
HDDやUSBメモリ，物理メモリのイメージファイルを解析し，必要な情報を得る

例えば犯罪捜査

不正アクセスや機密情報漏洩が起こったとき

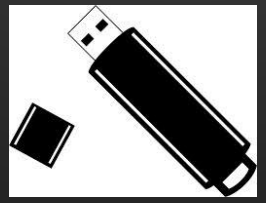
- 消えたファイルを特定
- 一部が壊れたデータの復元
- ファイルのタイムスタンプの調査

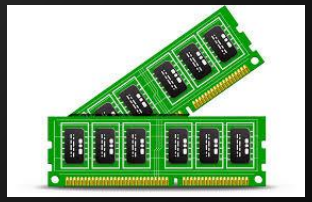




ファイルの特定・復元
パスワード解析

HDDやメモリの
データ





Pwnable

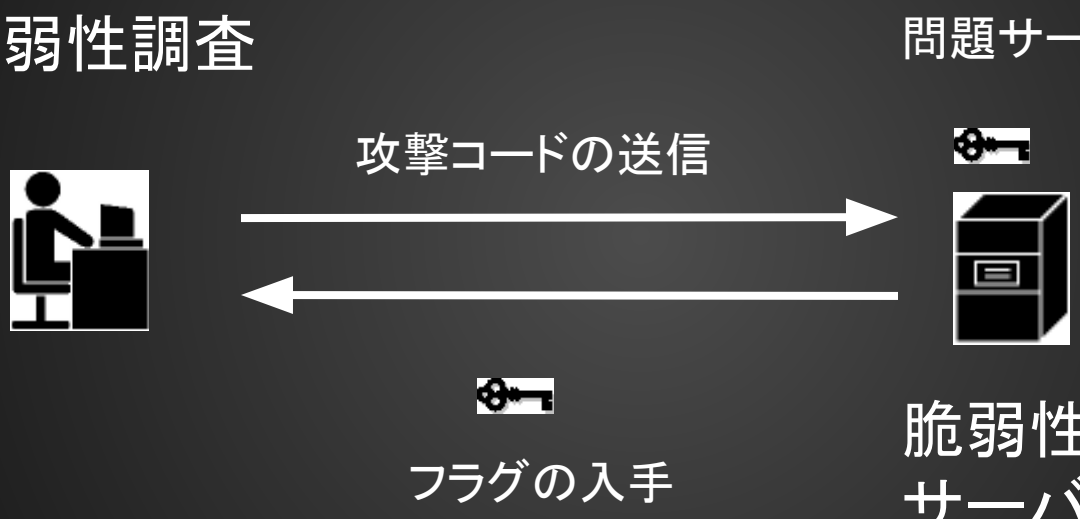
OSやプログラムの脆弱性(セキュリティ上の弱点)
を見つけて攻撃を行う

例えばサーバ管理

インターネットに公開しているサーバは安全?

依頼されたシステムがセキュアかどうかを
診断する仕事もある

サーバで動作するプログラムの脆弱性を突く



キーワード

OSコマンドインジェクション
バッファーオーバーフロー
ROP(Return-oriented Programming)など

Web

Webアプリケーションの脆弱性を探しだして
攻撃する

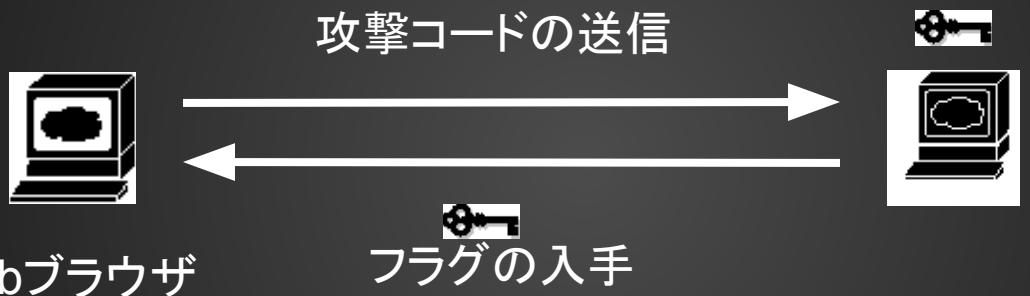
例えばWebエンジニア

Webアプリケーションを作成する人がセキュリティを知らないと大変なことに・・・

攻撃方法を知ることで安全なアプリケーション
を作れる！

基本的にはPwnableと同じ
対象がWebアプリケーションに変わる

脆弱性調査



脆弱性を持った
Webアプリケーション

キーワード
XSS(クロスサイトスクリプティング)
SQLインジェクション
CSRF(クロスサイトリクエストフォージェリ)

DEMO

フォレンジクス

SECCON 2014夏オンライン予選 Forensic 100

問題名: 879,394bytes
ファイル名: Filesystem001.bin

CTFでは基本的に問題文がない
→出題ジャンル・問題名・ファイル名がヒント!!

フォレンジクス

Seccon 2014夏【Forensic 100】⟵ フォレンジクスダヤッター!

問題名: 879,394bytes ⟵ 何かのデータのサイズ？
ファイル名: Filesystem001.bin

ファイルシステムって書いてるしそれ系の知識が必要そう

バイナリファイルに立ち向かうツール

- fileコマンド
  ファイルの形式を判定

- stringsコマンド
  内部のASCII文字列を抽出

- バイナリエディタ
  実際に中身を見る

Google「求めよ、さらば与えられん」

見えたことをとにかく先生に聞いてみる

stringsの出力に注目

- CHRYSA~1JPG
- YDRAN~1JPG
- TULIPS JPG
- PANDA JPG

Google「求めよ、さらば与えられん」

見えたことをとにかく先生に聞いてみる

stringsの出力に注目

- CHRYSA~1JPG
- YDRAN~1JPG
- TULIPS JPG
- PANDA JPG

「短い　ファイル名　大文字」

「ファイルシステム　~1」あたりでググる

Google「求めよ、さらば与えられん」

見えたことをとにかく先生に聞いてみる

stringsの出力に注目

- CHRYSA~1JPG
- YDRAN~1JPG
- TULIPS JPG
- PANDA JPG

「短い　ファイル名　大文字」

「ファイルシステム　~1」あたりでググる

→「FATの8.3形式」がみつかる

FAT?

90年代のWindowsでデフォルトだった
ファイルシステムのこと。

USBメモリとかでは現役。

ファイルシステムとは
　0と1の並びしか表現できないHDD上に
　ファイル/フォルダを実現するための仕組み
　　例) NTFS、HFS+、Btrfs

8.3形式？

- 古いFATではファイル名の長さに制限

- 長い名前が使えるようになってからも
  互換性のために短い名前も勝手につく

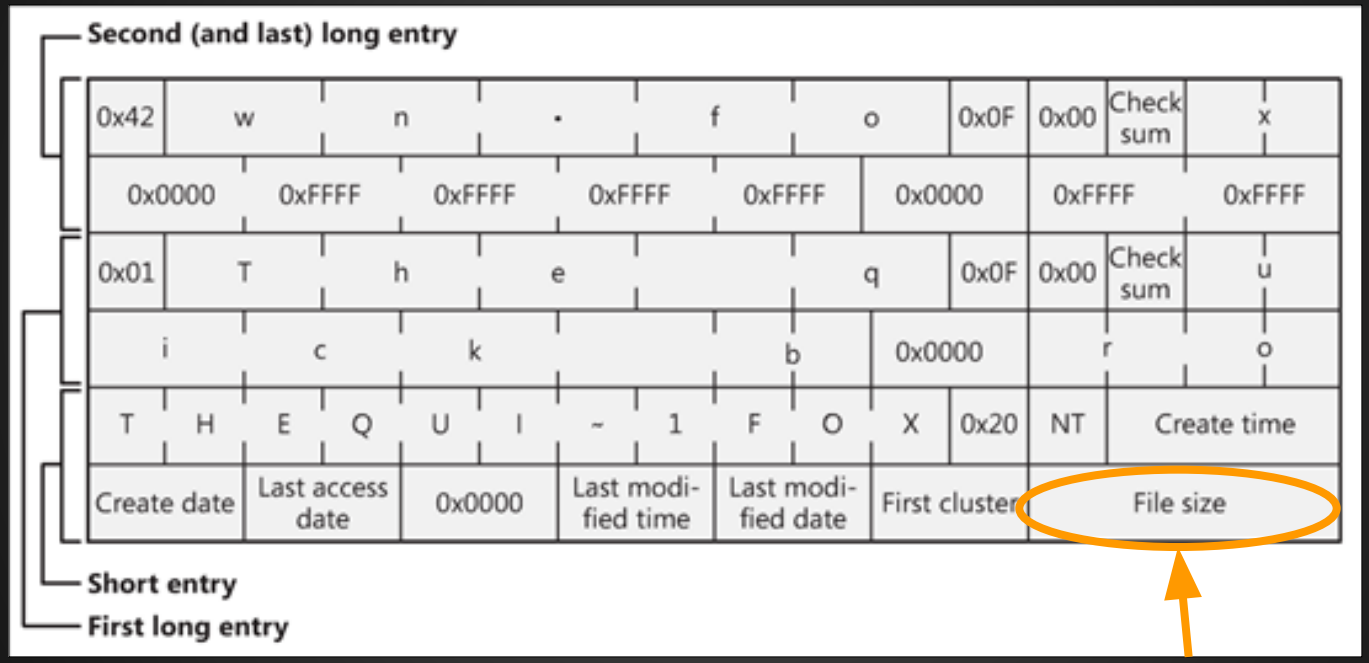
ネタバレ

Filesystem001.binの中には
ファイルのメタデータが並んでいる。
（ディレクトリエントリ）

この問題でやること

1. サイズが879,394bytesのファイルを探す
2. 元の長いファイル名が何か調べる

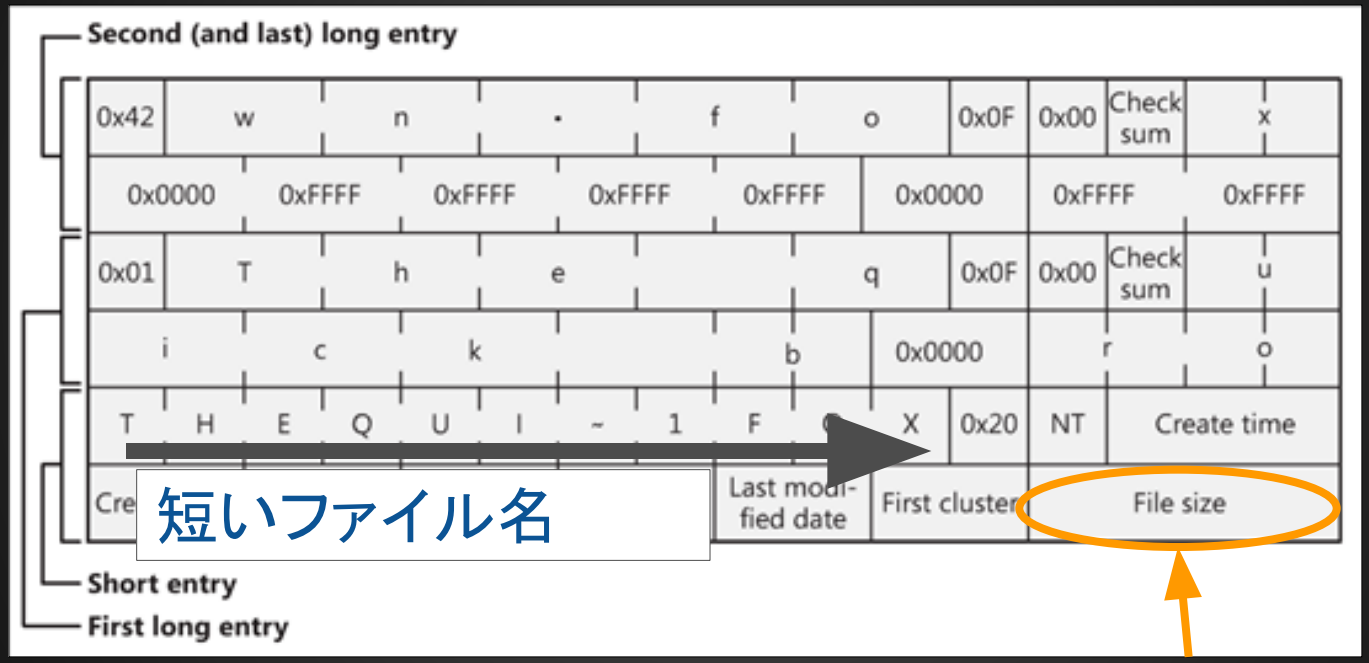
FATのディレクトリエントリを調査



ファイルサイズ

http://social.technet.microsoft.com/wiki/contents/articles/6771.the-fat-file-system.aspx

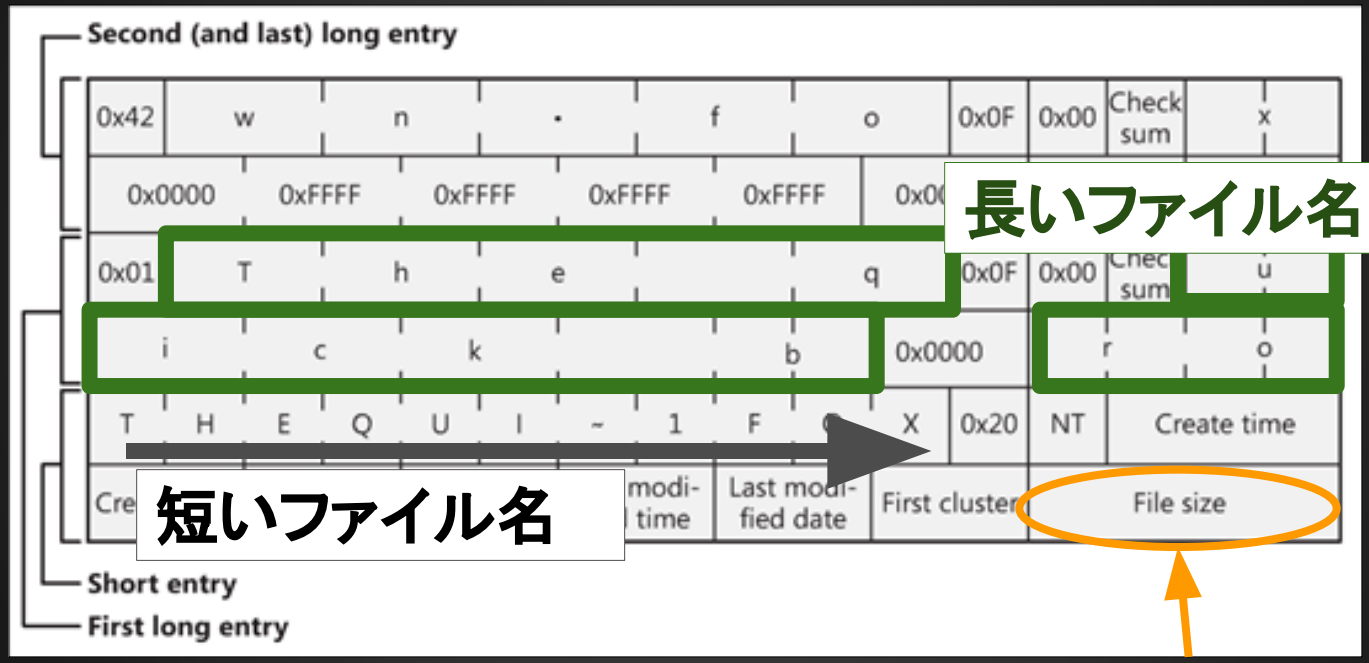
FATのディレクトリエントリを調査



ファイルサイズ

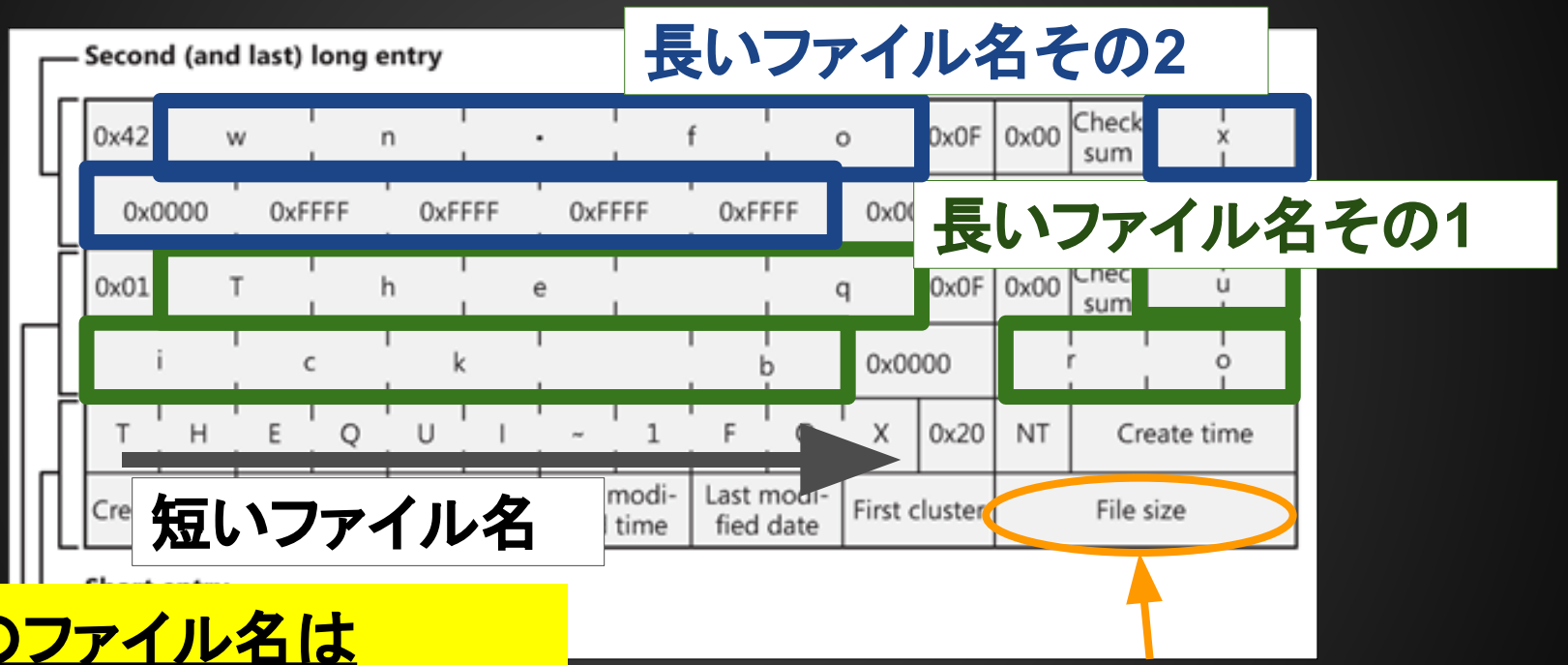
http://social.technet.microsoft.com/wiki/contents/articles/6771.the-fat-file-system.aspx

FATのディレクトリエントリを調査



http://social.technet.microsoft.com/wiki/contents/articles/6771.the-fat-file-system.aspx

FATのディレクトリエントリを調査



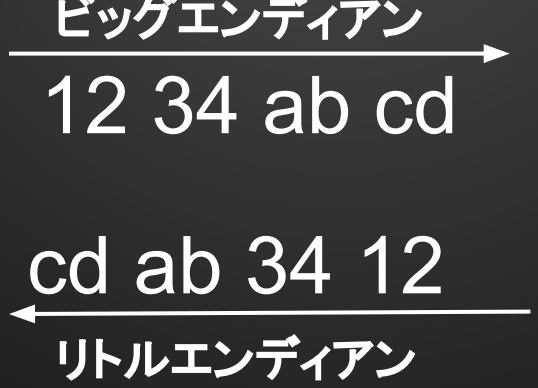
http://social.technet.microsoft.com/wiki/contents/articles/6771.the-fat-file-system.aspx

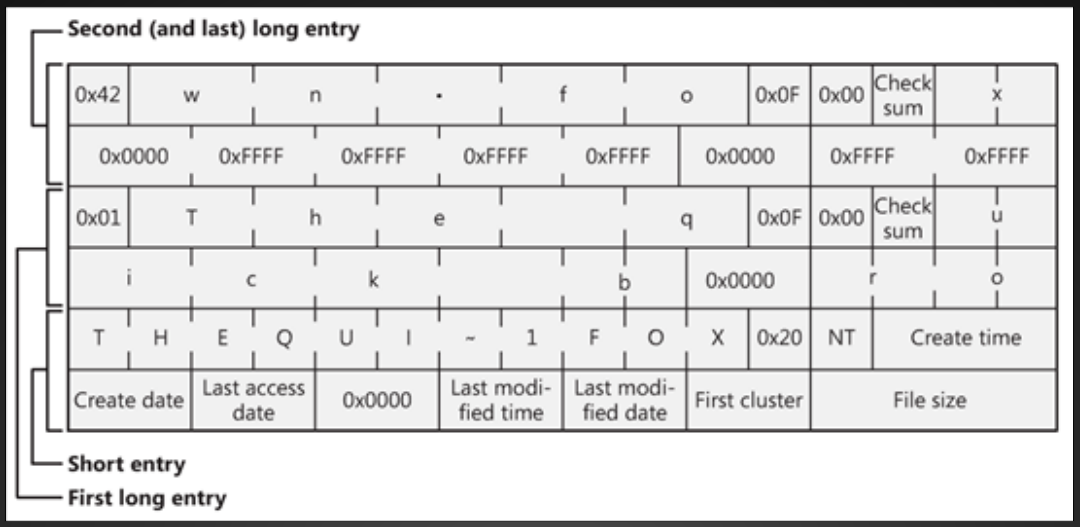
FATという仮定でバイナリをみる

```
0000000: 2e20 2020 2020 2020 2020 2010 0059 dc58  .           ..Y.X
0000010: 9b40 9b40 0000 dd58 9b40 0500 0000 0000  .@.@...X.@......
0000020: 2e2e 2020 2020 2020 2020 2010 0059 dc58  ..          ..Y.X
0000030: 9b40 9b40 0000 dd58 9b40 0000 0000 0000  .@.@...X.@......
0000040: 422e 006a 0070 0067 0000 000f 00c7 ffff  B..j.p.g........
0000050: ffff ffff ffff ffff ffff 0000 ffff ffff  ................
0000060: 0143 0068 0072 0079 0073 000f 00c7 6100  .C.h.r.y.s....a.
0000070: 6e00 7400 6800 6500 6d00 0000 7500 6d00  n.t.h.e.m...u.m.
0000080: 4348 5259 5341 7e31 4a50 4720 001d e358  CHRYSA~1JPG ...X
0000090: 9b40 9b40 0000 8d6e ee3a 0600 226b 0d00  .@.@...n.:.."k..
00000a0: e544 0065 0073 0065 0072 000f 00e3 7400  .D.e.s.e.r....t.
00000b0: 2e00 6a00 7000 6700 0000 0000 ffff ffff  ..j.p.g.........
00000c0: e545 5345 5254 2020 4a50 4720 0020 e358  .ESERT  JPG . .X
00000d0: 9b40 9b40 0000 8d6e ee3a dd00 75e8 0c00  .@.@...n.:..u...
00000e0: e567 0000 00ff ffff ffff ff0f 001d ffff  .g..............
00000f0: ffff ffff ffff ffff ffff 0000 ffff ffff  ................
0000100: e548 0079 0064 0072 0061 000f 001d 6e00  .H.y.d.r.a....n.
0000110: 6700 6500 6100 7300 2e00 0000 6a00 7000  g.e.a.s.....j.p.
0000120: e559 4452 414e 7e31 4a50 4720 0022 e358  .YDRAN~1JPG ."X
0000130: 9b40 9b40 0000 8d6e ee3a ac01 5415 0900  .@.@...n.:..T...
0000140: e54a 0065 006c 006c 0079 000f 006f 6600  .J.e.l.l.y...of.
0000150: 6900 7300 6800 2e00 6a00 0000 7000 6700  i.s.h...j...p.g.
0000160: e545 4c4c 5946 7e31 4a50 4720 0022 e358  .ELLYF~1JPG ."X
```

879,394bytesのファイルを検索

- 879394は16進数で0xd6b22
- FATのディレクトリエントリでFileSizeを確認
  - FileSizeは4バイト=00 0d 6b 22
- FATはリトルエンディアン
  - 22 6b 0d 00というバイナリ列を探す





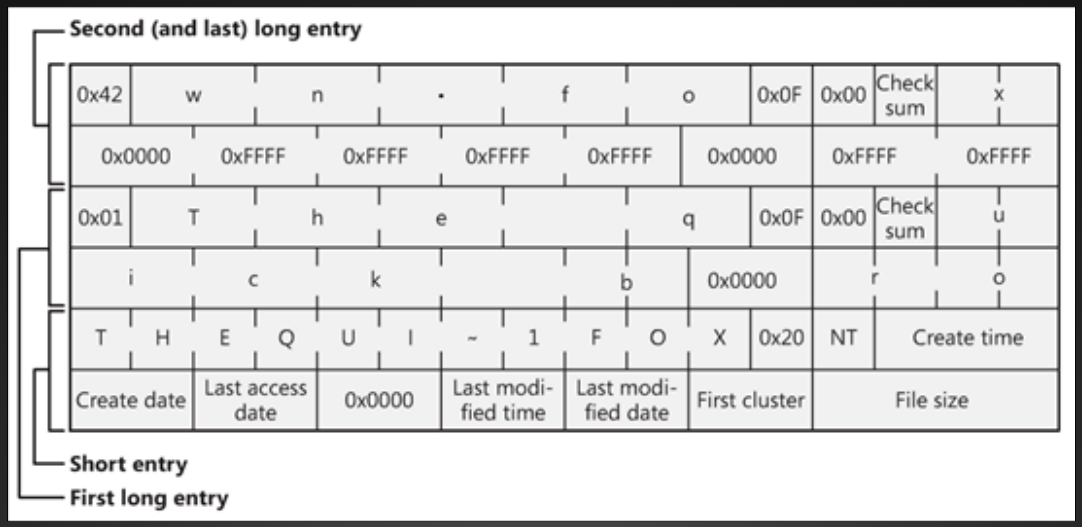
long file name

UTF-16
(2バイトで一文字)

```
0000040: 422e 006a 0070 0067 0000 000f 00c7 ffff  B..j.p.g.......
0000050: ffff ffff ffff ffff ffff 0000 ffff ffff  ................
0000060: 0143 0068 0072 0079 0073 000f 00c7 6100  .C.h.r.y.s.....a.
0000070: 6e00 7400 6800 6500 6d00 0000 7500 6d00  n.t.h.e.m...u.m.
0000080: 4348 5259 5341 7e31 4a50 4720 001d e358  CHRYSA~1JPG ...X
0000090: 9b40 9b40 0000 8d6e ee3a 0600 226b 0d00  .@.@...n.:.."k..
```

short file name

File size



long file name

UTF-16
(2バイトで一文字)

```
0000040: 422e 006a 0070 0067 0000 000f 00c7 ffff  B..j.p.g.......
0000050: ffff ffff ffff ffff ffff 0000 ffff ffff  ................
0000060: 0143 0068 0072 0079 0073 000f 00c7 6100  .C.h.r.y.s.....a.
0000070: 6e00 7400 6800 6500 6d00 0000 7500 6d00  n.t.h.e.m...u.m.
0000080: 4348 5259 5341 7e31 4a50 4720 001d e358  CHRYSA~1JPG ...X
0000090: 9b40 9b40 0000 8d6e ee3a 0600 226b 0d00  .@.@...n.:.."k..
```

short file name

File size

答えは”Chrysanthemum.jpg”

問題を解くには知識が必要！

- Linuxコマンド
- ファイルシステム
  - 歴史
  - ディレクトリエントリ
- エンディアン
- 文字コード

これらを全て暗記する必要はない

どうやって調べれるか何を調べればいいかを知っておけばいい

DEMO2

ネットワーク

SECCON 2014 Winter NW100のアレンジ
問題概要:
HTTPのBASIC認証を行った際の通信を
記録したファイルがある。
このファイルからFlagを見つけ出せ

前提知識

HTTP:
Webページを見る時に使う通信方式
基本的には暗号化されない

BASIC認証:
“user_name:password”という形式の文字列を
BASE64というもので変換したものを使って
認証を行う方式

で、この問題ではどうするの？

この問題では通信を記録したファイルがpcapという形式。

これはWiresSharkというソフトで扱えるので、WireSharkで開いてみる。

実演

ということで…

Flagは”`BASICAuthIsNotSecure`”

この問題からは…

- HTTPは通信が暗号化されていない
- BASIC認証は認証情報から元の情報が復元出来てしまう

ということが分かった

結びに

CTFをすればこんないいことが！

- 情報セキュリティ企業への就職に役立つ？
  - 24万人の情報技術者不足

  -http://www.nhk.or.jp/kaisetsu-blog/100/202598.html
- セキュリティに限らない広範囲の知識がつく
- コンピュータの基本原理を理解できる
- トラブルシューティングのカンがつく

CTFをやる上での注意

許可なくやると違法となる行為もある
だけど、CTFでは、
出題者がそもそも解析を前提に
公開しているのでOK

逆に言えば、こんな機会でもないと
こんなことを（合法的に）出来ないよね

CTFの勉強方法

とりあえず過去問を得く！

http://captf.com/

http://shell-storm.org/repo/CTF/

https://github.com/ctfs

わからなければ[大会 問題名 Writeup]で検索!

ex) CSAW CTF 2014 binary100 writeup

他の人の解法を参考に勉強する

これを見てCTFに興味が湧いた方へ

CTF for ビギナーズ

- 2015年10月 3日(土)
  CTF for ビギナーズ 2015 滋賀 (立命館大学BKC)
- 2015年10月17日(土)
  CTF for ビギナーズ 2015 奈良
  (奈良先端科学技術大)
- 2015年11月 7日(土)
  CTF for ビギナーズ 2015 大阪 (大阪南港 ATC)

SECCON

日本最大のCTF大会

SECCON 2015 オンライン予選

12月 5日(土)～6(日)

http://2015.seccon.jp/

予選はオンラインなので、
ひとまず予選に参加してみよう！

オンラインの常設CTF

- ksnctf
  - http://ksnctf.sweetduet.info/
- akictf
  - http://ctf.katsudon.org/

最後に

最後に

- 最近色々な脆弱性とかで騒がれている
- 安全のためにセキュリティはやっぱり不可欠
- でもハードルが高そう…という印象はある

- CTFとかを通じてセキュリティを身近に感じて頂ければ幸いです