Cybozu Labs

# Disassemble Flash Lite 3.0 SWF file

(How to protect your ActionScript source code)

TAKESAKO

@32bit.in

<takesako@gmail.com>

Cybozu Labs

## Japan is a birthplace of "K-ON!" and "K-TAI"

- Japanese cellular phones are called "K-TAI"
  - Japanese cellular phones have made original evolution because the communication method isn't an universal standard and so on
    - those have many functions, such as infrared ray, pictographs, electronic money, and television,
    - Therefore, In Japan, there aren't NOKIA's
- Japanese people want smart phone now
  - But, many old K-TAI still remain in Japan, and
    - many people are playing “Flash Lite” game on K-TAI!
      - GREE, mobage, mixi …


CybozuLabs

## SWF File Format Specification

- SWF Technology Center | Adobe Developer Connection
  - http://www.adobe.com/devnet/swf.html
    - VERSION 10
- Alexis' SWF Reference
  - http://sswf.sourceforge.net/SWFalexref.html
    - Flash VERSION 1, 2, 3, 4, 5, 6, …
      - Flash 1.0 ～ Flash 4.0 （FlashLite 1.1）
      - 2001～



CybozuLabs

## Alexis' SWF Reference

http://sswf.sourceforge.net/SWFalexref.html

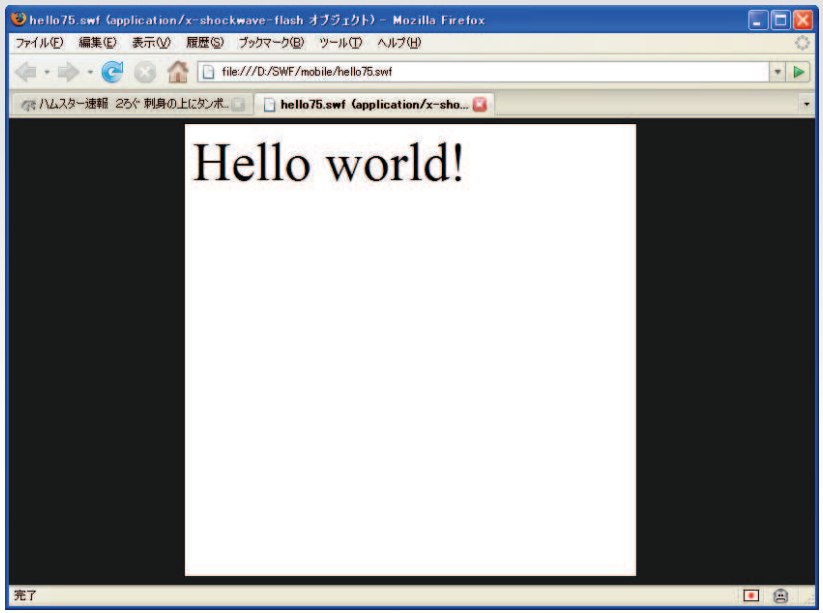
| Name | Number | Type | Comments |
|---|---|---|---|
| SWF Version 1.0 | | | |
| *File Header* | *None* | *Format* | We can consider the file header as being a tag though it isn't a tag per say. |
| END | 0 | *Format* | Mark the end of the file. It can't appear anywhere else but the end of the file. |
| SHOWFRAME | 1 | Define | Display the current display list and pauses for 1 frame as defined in the file header. |
| DEFINESHAPE | 2 | Define | Define a simple geometric shape. |
| FreeCharacter *Unknown encoding* | 3 | Define | Release a character which won't be used anymore. |
| PLACEOBJECT | 4 | Display | Place the specified object in the current display list. |
| REMOVEOBJECT | 5 | Display | Remove the specified object at the specified depth. |
| DEFINEBITSJPEG | 6 | Define | Define a JPEG bit stream. |
| DEFINEBUTTON | 7 | Define | Define an action button. |
| JPEGTABLES | 8 | Define | Define the tables used to compress/decompress all the SWF 1.0 JPEG images (See also DEFINEBITSJPEG). |
| SETBACKGROUNDCOLOR | 9 | Display | Change the background color. |
| DEFINEFONT | 10 | Define | List shapes corresponding to glyphs. |
| DEFINETEXT | 11 | Define | Defines a text of characters displayed using a font. This definition doesn't support any transparency. |
| : | : | : | : |

# SWF

## Flash Lite 1.1+ “Hello world!”

- hello.swf (75 byte)

```
46 57 53 04 4b 00 00 00 60 00 3f c0 00 3f c0 00
0c 02 00 43 02 33 33 33 17 03 96 12 00 00 6f 00
00 48 65 6c 6c 6f 20 77 6f 72 6c 64 21 0a 00 1d
00 4d 09 01 00 60 0a 3e 80 0a 3e 80 60 08 6f 00
05 01 01 00 01 00 00 40 00 00 00
```






## > swfdump -D hello.swf

```
[HEADER]        File version: 4
[HEADER]        File size: 75
[HEADER]        Frame rate: 12.000000
[HEADER]        Frame count: 2
[HEADER]        Movie width: 102.00
[HEADER]        Movie height: 102.00
[009]         3 SETBACKGROUNDCOLOR (33/33/33)
[00c]        23 DOACTION
                (   18 bytes) action: Push String:"o" String:"Hello world!¥n"
                (    0 bytes) action: SetVariable
                (    0 bytes) action: End
[025]        13 DEFINEEDITTEXT defines id 0001 variable "o"
[004]         5 PLACEOBJECT places id 0001 at depth 0001

                | Matrix            | CXForm    r    g    b    a
                | 1.000 0.000   0.00 | mul     1.0  1.0  1.0  1.0
                | 0.000 1.000   0.00 | add       0    0    0    0
[001]         0 SHOWFRAME 1 (00:00:00,000)
[000]         0 END
```



## 1. How to read SWF flie

- SWF File magic (4byte)

```
46 57 53 04 4b 00 00 00 60 00 3f c0 00 3f c0 00
0c 02 00 43 02 33 33 33 17 03 96 12 00 00 6f 00
00 48 65 6c 6c 6f 20 77 6f 72 6c 64 21 0a 00 1d
00 4d 09 01 00 60 0a 3e 80 0a 3e 80 60 08 6f 00
05 01 01 00 01 00 00 40 00 00 00
```

```
struct swf_header {
        unsigned char           f_magic[3];     'FWS' or 'CWS'
        unsigned char           f_version;
        unsigned long           f_file_length;
}
```

## 2. SWF File length (4byte)

- 32bit integer (Little Endian format)

```
46 57 53 04 4b 00 00 00 60 00 3f c0 00 3f c0 00
0c 02 00 43 02 33 33 33 17 03 96 12 00 00 6f 00
00 48 65 6c 6c 6f 20 77 6f 72 6c 64 21 0a 00 1d
00 4d 09 01 00 60 0a 3e 80 0a 3e 80 60 08 6f 00
05 01 01 00 01 00 00 40 00 00 00
```

```
struct swf_header {
        unsigned char           f_magic[3];     'FWS' or 'CWS'
        unsigned char           f_version;
        unsigned long           f_file_length;
}
                                                57 = 4b 00 00 00
```





## 3. swf_header_movie (swf_rect)

- swf_rect (variable length)

```
46 57 53 04 4b 00 00 00 60 00 3f c0 00 3f c0 00
0c 02 00 43 02 33 33 33 17 03 96 12 00 00 6f 00
00 48 65 6c 6c 6f 20 77 6f 72 6c 64 21 0a 00 1d
00 4d 09 01 00 60 0a 3e 80 0a 3e 80 60 08 6f 00
05 01 01 00 01 00 00 40 00 00 00
```

```
struct swf_header_movie {
        swf_rect                f_frame_size;
        unsigned short fixed    f_frame_rate;
        unsigned short          f_frame_count;
};
```





## 3.1. swf_rect (variable length) format

- E.g. Decode “60 00 3f c0 00 3f c0”


```
| ssss sxxx | xxxx xxxx | xXXX XXXX | XXXX Xyyy | yyyy yyyy | yYYY YYYY | YYYY Y000 |
| 0110 0000 | 0000 0000 | 0011 1111 | 1100 0000 | 0000 0000 | 0011 1111 | 1100 0000 |
|  6   0    |  0   0    |  3   f    |  c   0    |  0   0    |  3   f    |  c   0    |
```

```
f_size = sssss(5bit) = 011000 = 12
f_x_min = xxxxxxxxxxxx(12bit) =     0 twips
f_x_max = XXXXXXXXXXXX(12bit) = +2040 twips (104px)
f_y_min = yyyyyyyyyyyy(12bit) =     0 twips
f_y_max = YYYYYYYYYYYY(12bit) = +2040 twips (104px)
```

2^12 = -2047〜+2047

```
struct swf_rect {
        char align;
        unsigned                f_size : 5;
        signed twips            f_x_min : f_size;
        signed twips            f_x_max : f_size;
        signed twips            f_y_min : f_size;
        signed twips            f_y_max : f_size;
};
```





## 3.2. swf_header_movie (f_frame_rate)

- Frame Rate (2byte) = 12.0 frame/sec

```
46 57 53 04 4b 00 00 00 60 00 3f c0 00 3f c0 00
0c 02 00 43 02 33 33 33 17 03 96 12 00 00 6f 00
00 48 65 6c 6c 6f 20 77 6f 72 6c 64 21 0a 00 1d
00 4d 09 01 00 60 0a 3e 80 0a 3e 80 60 08 6f 00
05 01 01 00 01 00 00 40 00 00 00
```

```
struct swf_header_movie {
        swf_rect                f_frame_size;
        unsigned short fixed    f_frame_rate;
        unsigned short          f_frame_count;
};
```

8.8 bit fixed-point integer 12.0 → 00 0c

## 3.3. swf_header_movie (f_frame_count)

- Frame count (16bit integer)

```
46 57 53 04 4b 00 00 00 60 00 3f c0 00 3f c0 00
0c 02 00 43 02 33 33 33 17 03 96 12 00 00 6f 00
00 48 65 6c 6c 6f 20 77 6f 72 6c 64 21 0a 00 1d
00 4d 09 01 00 60 0a 3e 80 0a 3e 80 60 08 6f 00
05 01 01 00 01 00 00 40 00 00 00
```

```
struct swf_header_movie {
        swf_rect              f_frame_size;
        unsigned short fixed  f_frame_rate;
        unsigned short        f_frame_count;
};
          16bit integer       automatically-calculated
```



## (Are you) still with me?

- SWF header is finished!

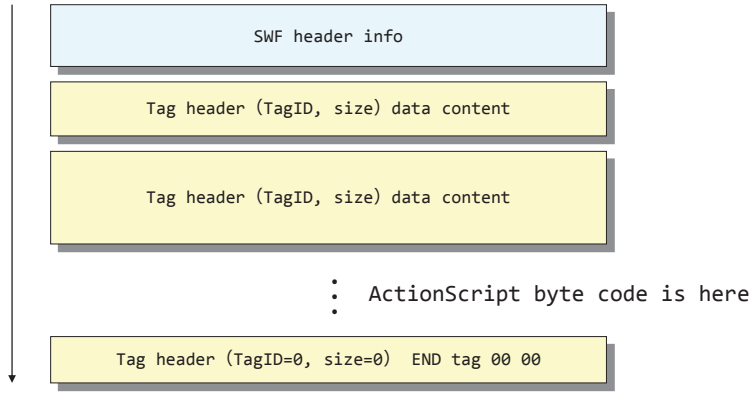
```
46 57 53 04 4b 00 00 00 60 00 3f c0 00 3f c0 00
0c 02 00 43 02 33 33 33 17 03 96 12 00 00 6f 00
00 48 65 6c 6c 6f 20 77 6f 72 6c 64 21 0a 00 1d
00 4d 09 01 00 60 0a 3e 80 0a 3e 80 60 08 6f 00
05 01 01 00 01 00 00 40 00 00 00
```

Go to Next Stage!



## SWF File Format Overview

- Block image






## 4.1. SWF tag (variable length) format

```
46 57 53 04 4b 00 00 00 60 00 3f c0 00 3f c0 00
0c 02 00 43 02 33 33 33 17 03 96 12 00 00 6f 00
00 48 65 6c 6c 6f 20 77 6f 72 6c 64 21 0a 00 1d
00 4d 09 01 00 60 0a 3e 80 0a 3e 80 60 08 6f 00
05 01 01 00 01 00 00 40 00 00 00
```

Bit operation!

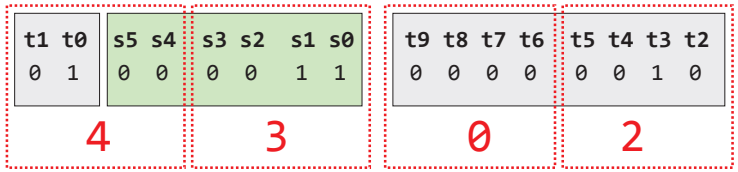
```
struct swf_tag {
        unsigned short          f_tag_and_size;
        f_tag = f_tag_and_size >> 6;
        f_tag_data_size = f_tag_and_size & 0x3F;
        if(f_tag_data_size == 63) {
                unsigned long   f_tag_data_real_size;
        }
        else {
                f_tag_data_real_size = f_tag_data_size;
        }
};
```

43 02 ↓ TagID:09 Size:3

SetBackgroundColor(TagID:09) → RR GG BB (3byte)

## 4.2. SWF Tag header (2 byte)

1. If Size < 63 byte




TagID = t0*2^0 + t1*2^1 + t2*2^2 + t3*2^3 + ⋯ t9*2^9 = t0*1 +t3*8 = 1 + 8 = 9

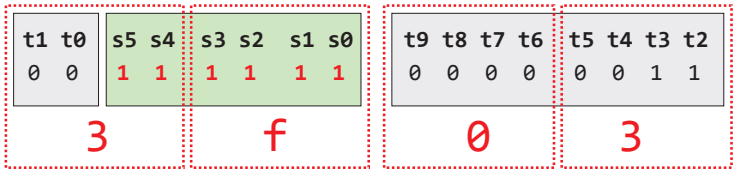
Size = s0*2^0 + s1*2^1 + s2*2^2 + s3*2^3 + ⋯ t5*2^5 = s0*1 +s1*2 = 1 + 2 = 3





## 4.3. SWF Tag header (6 byte)

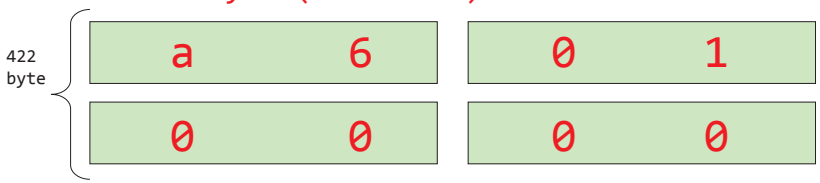
2. If Size > 62 byte




TagID = t0*2^0 + t1*2^1 + t2*2^2 + t3*2^3 + ⋯ t9*2^9 = t2*4 +t3*8 = 4 + 8 = 12

Size = 2^0 + 2^1 + 2^2 + 2^3 + 2^4 + 2^5 = 63 (0x3f) ← magic number

Next 4byte (32bit int) is real size data








## Lesson 1. (calculate SWF Tag header)

- For instance, the data of tag is converted into the byte sequence of such feeling

3byte

442byte

```
SetBgColor    (TagID= 9, Size=3) → 43 02 RR GG BB
doAction      (TagID=12,Size=422)→  3f 03 a6 01 00 00 xx ⋯ xx
PlaceObject   (TagID= 4, Size=7) → 07 01 xx xx xx xx xx xx xx
PlaceObject2  (TagID=26, Size=8) → 88 06 xx xx xx xx xx xx xx xx
ShowFrame     (TagID= 1, Size=0) → 40 00
End           (TagID= 0, Size=0) → 00 00
```



## 5. doAction Tag

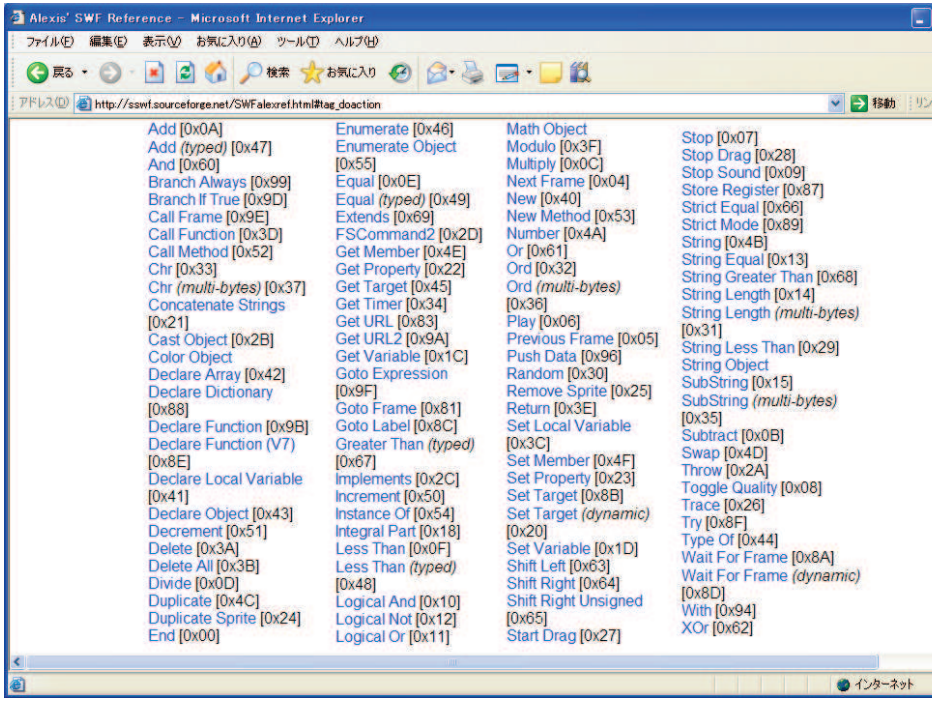
```
46 57 53 04 4b 00 00 00 60 00 3f c0 00 3f c0 00
0c 02 00 43 02 33 33 33 17 03 96 12 00 00 6f 00
00 48 65 6c 6c 6f 20 77 6f 72 6c 64 21 0a 00 1d
00 4d 09 01 00 60 0a 3e 80 0a 3e 80 60 08 6f 00
05 01 01 00 01 00 00 40 00 00 00
```

**TagID** **Size** 17 03 = TagID:12 Size:23byte

[00c] 23 DOACTION

```
(   18 bytes) action: Push String:"o" String:"Hello world!¥n"
(    0 bytes) action: SetVariable
(    0 bytes) action: End
```

**Action Script Byte Code (explain later)**




CybozuLabs

## Flasm

http://flasm.sourceforge.net/

```
Flasm 1.62 build Jun  9 2007

(c) 2001 Opaque Industries, (c) 2002-2007 Igor Kogan, (c) 2005 Wang Zhen
All rights reserved. See LICENSE.TXT for terms of use.

Usage: flasm [command] filename

Commands:
   -d     Disassemble SWF file to the console
   -a     Assemble Flasm project (FLM)
   -u     Update SWF file, replace Flasm macros
   -b     Assemble actions to __bytecode__ instruction or byte sequence
   -z     Compress SWF with zLib
   -x     Decompress SWF

Backups with $wf extension are created for altered SWF files.

To save disassembly or __bytecode__ to file, redirect it:
flasm -d foo.swf > foo.flm
flasm -b foo.txt > foo.as

Read flasm.html for more information.
```

CybozuLabs

## 6. DefinedEditText Tag

```
46 57 53 04 4b 00 00 00 60 00 3f c0 00 3f c0 00
0c 02 00 43 02 33 33 33 17 03 96 12 00 00 6f 00
00 48 65 6c 6c 6f 20 77 6f 72 6c 64 21 0a 00 1d
00 4d 09 01 00 60 0a 3e 80 0a 3e 80 60 08 6f 00
05 01 01 00 01 00 00 40 00 00 00
```

```
TagID    Size
[025]      13 DEFINEEDITTEXT defines id 0001 variable "o"
[004]       5 PLACEOBJECT places id 0001 at depth 0001
                | Matrix             | CXForm    r    g    b    a
                | 1.000 0.000   0.00 | mul     1.0  1.0  1.0  1.0
                | 0.000 1.000   0.00 | add       0    0    0    0
[001]       0 SHOWFRAME 1 (00:00:00,000)
[000]       0 END
```



CybozuLabs

## 7. PlaceObject Tag

```
46 57 53 04 4b 00 00 00 60 00 3f c0 00 3f c0 00
0c 02 00 43 02 33 33 33 17 03 96 12 00 00 6f 00
00 48 65 6c 6c 6f 20 77 6f 72 6c 64 21 0a 00 1d
00 4d 09 01 00 60 0a 3e 80 0a 3e 80 60 08 6f 00
05 01 01 00 01 00 00 40 00 00 00
```

```
TagID    Size
[025]      13 DEFINEEDITTEXT defines id 0001 variable "o"
[004]       5 PLACEOBJECT places id 0001 at depth 0001
                | Matrix             | CXForm    r    g    b    a
                | 1.000 0.000   0.00 | mul     1.0  1.0  1.0  1.0
                | 0.000 1.000   0.00 | add       0    0    0    0
[001]       0 SHOWFRAME 1 (00:00:00,000)
[000]       0 END
```

## swf_matrix (variable length)

```
struct swf_matrix {
        char align;
        unsigned                f_has_scale : 1;
        if (f_has_scale) {
                unsigned        f_scale_bits : 5;
                signed fixed    f_scale_x : f_scale_bits;
                signed fixed    f_scale_y : f_scale_bits;
        }
        unsigned                f_has_rotate : 1;
        if (f_has_rotate) {
                unsigned        f_rotate_bits : 5;
                signed fixed    f_rotate_skew0 : f_rotate_bits;
                signed fixed    f_rotate_skew1 : f_rotate_bits;
        }
        unsigned                f_translate_bits : 5;
        signed                  f_translate_x : f_rotate_bits;
        signed                  f_translate_y : f_rotate_bits;
};
```





## 8. ShowFrame Tag

```
46 57 53 04 4b 00 00 00 60 00 3f c0 00 3f c0 00
0c 02 00 43 02 33 33 33 17 03 96 12 00 00 6f 00
00 48 65 6c 6c 6f 20 77 6f 72 6c 64 21 0a 00 1d
00 4d 09 01 00 60 0a 3e 80 0a 3e 80 60 08 6f 00
05 01 01 00 01 00 00 40 00 00 00
```

```
TagID     Size
[025]       13 DEFINEEDITTEXT defines id 0001 variable "o"
[004]        5 PLACEOBJECT places id 0001 at depth 0001
                | Matrix              | CXForm    r    g    b    a
                | 1.000 0.000   0.00 | mul     1.0  1.0  1.0  1.0
                | 0.000 1.000   0.00 | add       0    0    0    0
[001]        0 SHOWFRAME 1 (00:00:00,000)
[000]        0 END
```



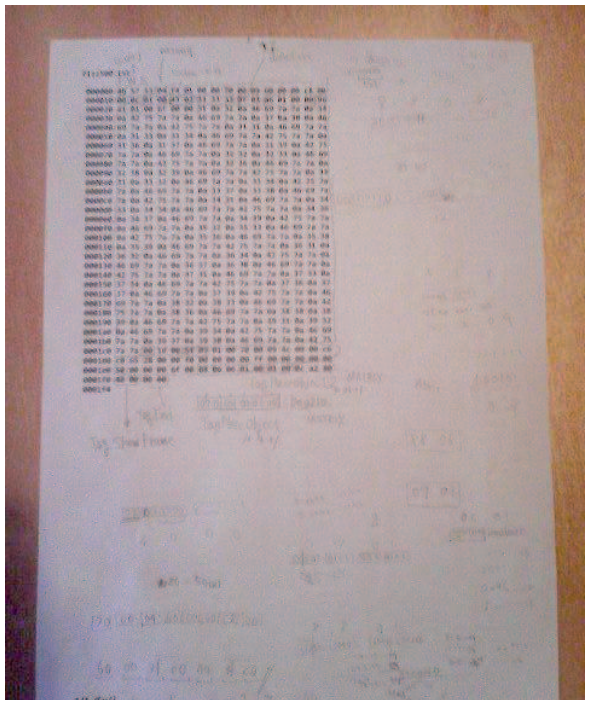


## 9. End Tag

```
46 57 53 04 4b 00 00 00 60 00 3f c0 00 3f c0 00
0c 02 00 43 02 33 33 33 17 03 96 12 00 00 6f 00
00 48 65 6c 6c 6f 20 77 6f 72 6c 64 21 0a 00 1d
00 4d 09 01 00 60 0a 3e 80 0a 3e 80 60 08 6f 00
05 01 01 00 01 00 00 40 00 00 00
```

```
TagID     Size
[025]       13 DEFINEEDITTEXT defines id 0001 variable "o"
[004]        5 PLACEOBJECT places id 0001 at depth 0001
                | Matrix              | CXForm    r    g    b    a
                | 1.000 0.000   0.00 | mul     1.0  1.0  1.0  1.0
                | 0.000 1.000   0.00 | add       0    0    0    0
[001]        0 SHOWFRAME 1 (00:00:00,000)
[000]        0 END
```

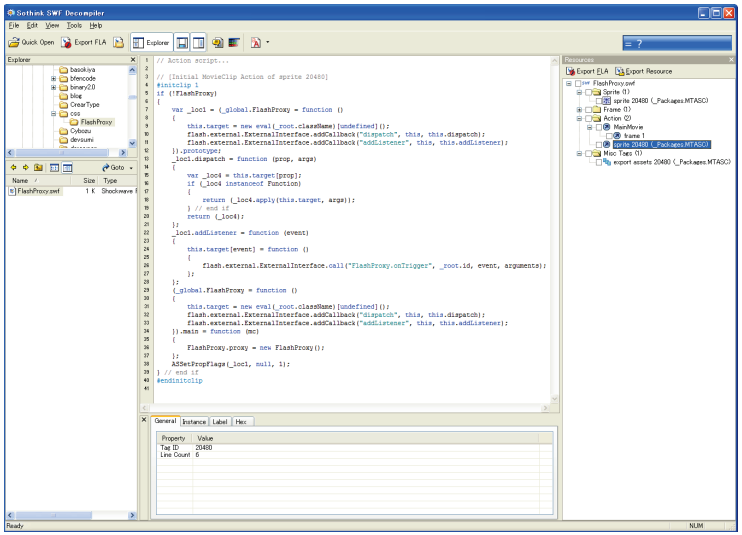




# ActionScript Decompiler



## (1) Sothink SWF Decompiler



http://www.sothink.com/product/flashdecompiler/

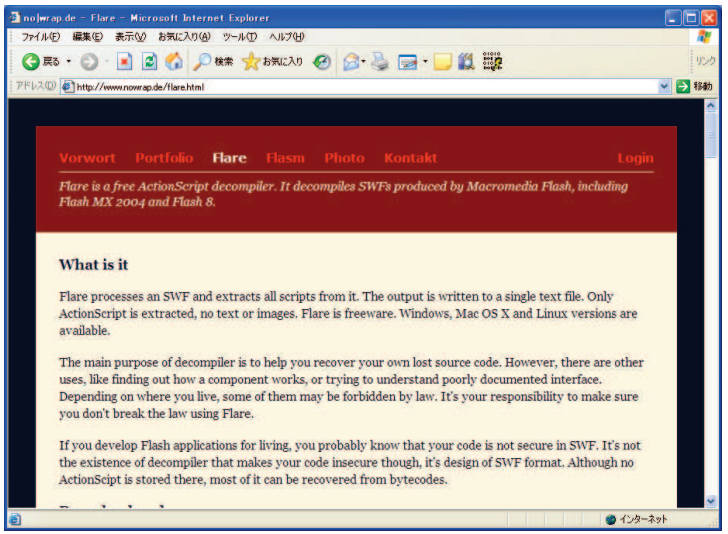


## (1) SWF → ActionScript (Sothink SWF Decompiler)

```
// Action script...

// [Initial MovieClip Action of sprite 20480]
#initclip 1
if (!FlashProxy)
{
    var _loc1 = (_global.FlashProxy = function ()
    {
        this.target = new eval(_root.className)[undefined]();
        flash.external.ExternalInterface.addCallback("dispatch", this, this.dispatch);
        flash.external.ExternalInterface.addCallback("addListener", this, this.addListener);
    }).prototype;
    _loc1.dispatch = function (prop, args)
    {
        var _loc4 = this.target[prop];
        if (_loc4 instanceof Function)
        {
            return (_loc4.apply(this.target, args));
        } // end if
        return (_loc4);
    };
    _loc1.addListener = function (event)
    {
        this.target[event] = function ()
        {
```



## (2) Flare http://www.nowrap.de/flare.html



## [Demo] flare nicovideo.swf → nicovideo.flr

```
    _global.styles.TextArea.setStyle('borderStyle', 'solid');
    _global.styles.TextArea.setStyle('backgroundColor', '0xFFFFEE');
    _global.styles.ComboBox.setStyle('rollOverColor', '0xF0FFF0');
    var arr = _url.split('/');
    var U = 'http://' + strReplace(arr[2], 'res', (country_code == undefined) ?
'www' : country_code) + ((arr[3].charAt(0) == '~') ? '/' + arr[3] : '') + '/';
    var NICOVIDEO_URL = 'http://' + strReplace(arr[2], 'ext', (country_code ==
undefined) ? 'www' : country_code) + ((arr[3].charAt(0) == '~') ? '/' +
arr[3] : '') + '/';
    var PLAYER_VERSION = '200808211900';
    var B = U + 'api/';
    system.useCodepage = true;
    Stage.scaleMode = 'noScale';
    Stage.align = 'TL';
    System.security.allowDomain('www.nicovideo.jp');
    System.security.allowDomain('www.dev.nicovideo.jp');
    System.security.allowDomain('res.nicovideo.jp');
    System.security.allowDomain('dwango.co.jp');
```

# [Demo]

# swf2protect



## swf2protect (How to use)

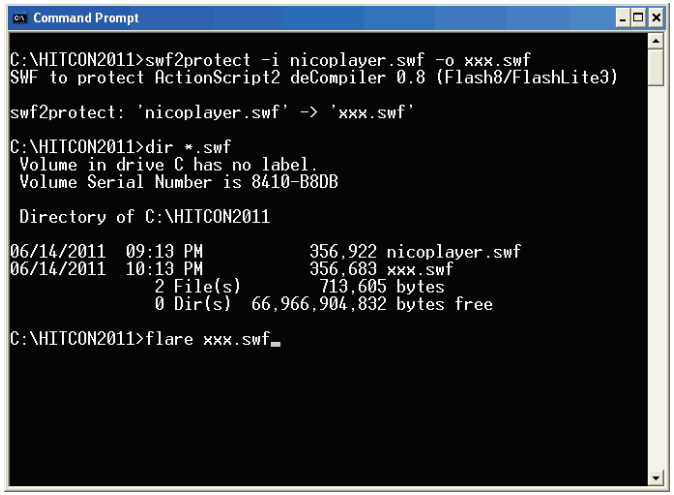
```
Command Prompt
C:\HITCON2011>swf2protect -h
SWF to protect ActionScript2 deCompiler 0.8 (Flash8/FlashLite3)

Usage: swf2protect [-h] [-i] in.swf [[-o] out.swf]

Commands:
   -i in.swf  (input  SWF file) require
   -o out.swf (output SWF file) optional
   -c (compress) optional
   -d (decompress) optional

C:\HITCON2011>dir
 Volume in drive C has no label.
 Volume Serial Number is 8410-B8DB

 Directory of C:\HITCON2011

06/14/2011  10:02 PM    <DIR>          .
06/14/2011  10:02 PM    <DIR>          ..
02/09/2006  11:08 PM            78,848 flare.exe
06/14/2011  09:13 PM           356,922 nicoplayer.swf
06/14/2011  10:09 PM            93,138 swf2protect.exe
               3 File(s)        528,908 bytes
               2 Dir(s)  66,967,273,472 bytes free

C:\HITCON2011>_
```



## swf2protect (anti-decompiling)

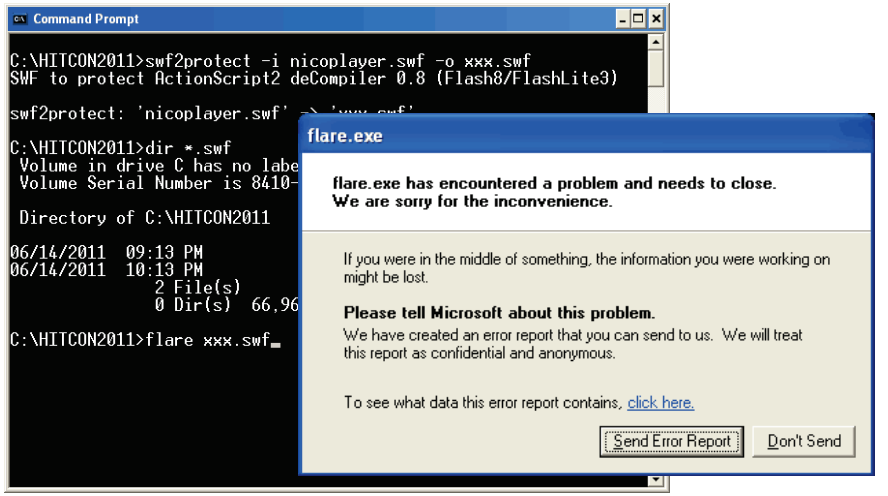
```
Command Prompt

C:\HITCON2011>swf2protect -i nicoplayer.swf -o xxx.swf
SWF to protect ActionScript2 deCompiler 0.8 (Flash8/FlashLite3)

swf2protect: 'nicoplayer.swf' -> 'xxx.swf'

C:\HITCON2011>dir *.swf
 Volume in drive C has no label.
 Volume Serial Number is 8410-B8DB

 Directory of C:\HITCON2011

06/14/2011  09:13 PM           356,922 nicoplayer.swf
06/14/2011  10:13 PM           356,683 xxx.swf
               2 File(s)        713,605 bytes
               0 Dir(s)  66,966,904,832 bytes free

C:\HITCON2011>_
```

# swf2protect (Demo)



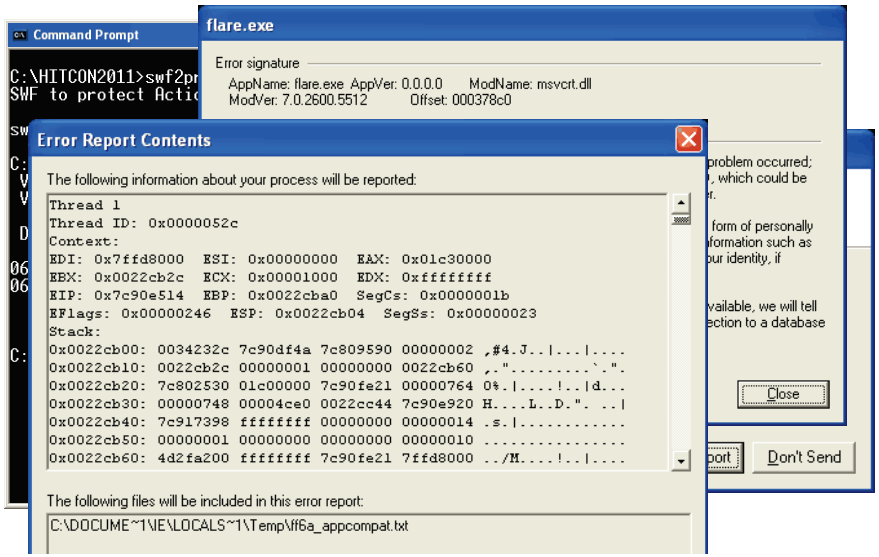



# Result: Flare failed to decompile (SEGV)






# Result: Flare failed to decompile (SEGV)






# Flash Player vs. ActionScript Decompiler

- Boundary jump (ActionScript byte code)