



HISPASEC SISTEMAS

SEGURIDAD Y TECNOLOGÍAS
DE LA INFORMACIÓN

**White paper:
Multiple vulnerabilities in IrfanView 2.4**

January 2009

Matthew "j00ru" Jurczyk
matthew@hispasec.com

1. Introduction

IrfanView is a widely spread, freeware graphic viewer for Windows operating systems. It supports a great number of various file formats, including TIFF (Tagged Image File Format).

Several vulnerabilities have been found in the application, that might be exploited by a remote attacker to cause a denial of service, and potentially lead to arbitrary code execution.

The issues are caused by a heap based buffer overflow that could lead to denial of service (application crash) or arbitrary code execution if a user opens a specially crafted image file.

2. Vulnerability details

This section shows the detailed analysis of the sample code vulnerable to the heap overflow:

```
.text:00411CA6      mov     ecx, edi
.text:00411CA8      imul   ecx, [esp+104h+var_CC]
.text:00411CAD      test   ecx, ecx
.text:00411CAF      mov     [esp+104h+var_9C], ecx
.text:00411CB3      jbe    short loc_411CFB
.text:00411CB5      mov     eax, ecx
.text:00411CB7      xor     edx, edx
.text:00411CB9      div    [esp+104h+var_CC]
.text:00411CBD      cmp     eax, edi
.text:00411CBF      jnb    short loc_411CFB
.text:00411CC1      lea    eax, [ecx+ebx*4+28h]
.text:00411CC5      push   eax                ; dwBytes
.text:00411CC6      push   40h                ; uFlags
.text:00411CC8      call   ds:GlobalAlloc
```

The code is responsible for allocating a properly-sized buffer for the image data. Before the GlobalAlloc call, it calculates the number of bytes needed to safely store the entire data. When the value is finally present in the ECX register (at 0411CAD address), it is checked against an Integer Overflow attack:

```
.text:00411CB9      div    [esp+104h+var_CC]
.text:00411CBD      cmp     eax, edi
.text:00411CBF      jnb    short loc_411CFB
```

A simple division is made using the multiplied value to ensure that $(A*B)/B$ is still A , which would not be true in case of $(A*B)$ expression causing an overflow. This seems to be a reasonable protection, except for one little detail:

```
.text:00411CC1      lea    eax, [ecx+ebx*4+28h]
```

Before the allocation is done, a 28h (40d) value is added to the calculation result, with no overflow check after it. The idea is to find the width and height values that

do not cause an overflow by themselves, but that cause it as the result of adding the magic 28h value to their multiplication result. Some exemplary pairs follow:

21846 65532

24553 58308

24939 57404

24940 57404

25914 55245

26215 54610

26216 54610

... and others ...

There are 53 pairs of width/height values that can be used to cause a heap corruption. Such attack could potentially lead to code execution with the privileges of the IrfanView user.

3. Denial of service when processing TIFF files

IrfanView application fails to properly sanitize the user-defined values in TIFF files, such as:

- ColorMap tag: the image palette offset.
Located in offset 320 (hex 0x0140).
- Strip Offset tag: one or more of the 32bit values pointing to particular strips' image data.
Located in offset 273 (hex 0x0111).
- First Image File Directory offset: a DWORD value pointing to the first IFD structure.

Setting these values to, for example 0xFFFFFFFF (-1) value, is very likely to generate an Access Violation exception while trying to read unmapped memory address lying before the file data, and consequently lead to the application crash.

4. Denial of service via IFD Offset in TIFF files

The TIFF file consists of multiple Image File Directory entries, each describing one specific image. At the end of the IFD's structure, there's a DWORD value - offset of the next directory, or 0 in case of the last one.

Setting the first directory's offset so as to make it point to itself, makes the IrfanView image parser get into an infinite loop, denying access to legitimate users.

5. Status y conclusions

The vulnerabilities has been confirmed for IrfanView v.4.20 but older versions could also be affected.

IrfanView v.4.23 is not affected.

6. Websites

Hispasec Sistemas

<http://hispasec.com>

IrfanView

<http://www.irfanview.com/>

Multiple web browser image based information leak

http://www.hispasec.com/laboratorio/vulnerabilidad_firefox_en.pdf