Week 8

MICT 1

Groep: Disco Ninja’s

Robin Chel 1241141

Jeroen van den Hurk 1071149

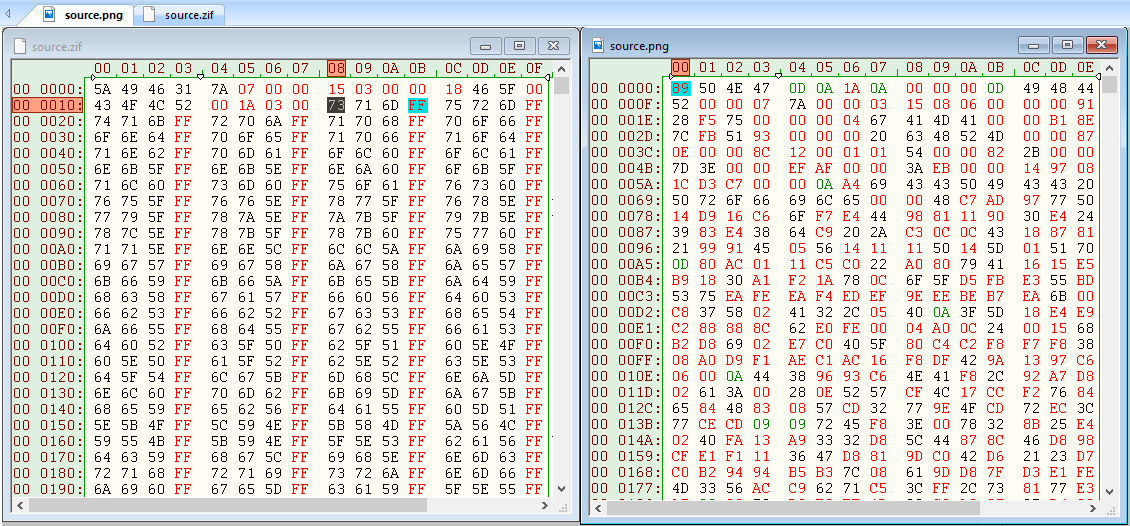
Luc Janssens 1070649

Kevin de Waal 1072153

# What we found

We zijn begonnen met kijken welke bytes in het bestand we herkennen als we hem vergelijken met de originele PNG format. Het eerst wat ons opviel is dat de eerste 4 bytes van het document aangeven dat het een ZIF file is.

Volgens de opdracht zouden de source.zif en de source.png dezelfde afbeelding bevatten. De vergelijking van de twee bestanden laat zien dat er grote verschillen zijn in de opbouw van de bestanden.



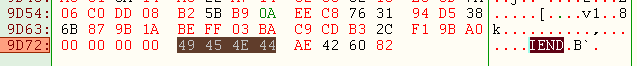
Daarbij staat de ZIF file in de verkeerde volgorde geschreven. Er moet een conversie naar little endian plaatsvinden om de opbouw van een Pixel waarde in BMP te laten kloppen.

In de originele PNG file begint de data van het plaatje nadat de waarde 49 44 41 54 (IDAT) is voorgekomen.

In de ZIF file hebben we wel een kleine vergelijking gevonden. Hierin staat 49 44 41 (IDA) wel aangegeven.

De regels waar IDA voorkomt is volgestopt met de waarde FF. Hierdoor is waarschijnlijk de T van IDAT overschreven. Als de hexadecimale waarde FF toegevoegd zijn aan het document zou de letter T later in het document voor moeten komen. Dis is niet het geval. Het is ons niet gelukt om een manier te vinden om de FF uit het bestand te verwijderen of terug te draaien naar originele waarde.

Een PNG file format eindigt ook met de data chunck IEND, deze is niet aanwezig in de ZIF file.



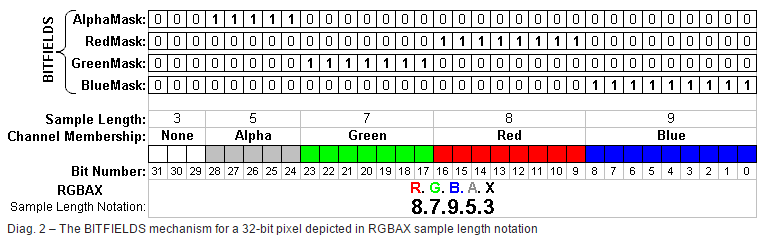
Dit deed ons sterk vermoeden dat het ZIF file format niet gebaseerd is op een PNG file format maar meer op BMP.

De PNG file is daarna geconverteerd naar alle bestandsindelingen die gebruik maken van een bitmap indeling. Helaas kwam er geen overeenkomsten van bestanden uit die lijken op de ZIF file format.

Verder is ons opgevallen dat adres: 00 00 0C t/m 00 00 13 aangeven dat er iets van een kleur bepaling aan komt en zijn we er vanuit gegaan dat 00 00 14 t/m 00 00 17 de waarde van deze kleur bepaling is. We hebben verder niet kunnen vinden met wat dit overeen komt in de PNG file.

# Where we found it

De grote verschillen tussen de twee bestandstypen zijn de opbouw van de data. Waar bij de ZIF file veel doet denken aan een BMP indeling. Er zijn steeds 32 bits die een pixel kleur aangeven. 



Zoals eerder vermeld moet de ZIF file in little endian weergeven worden om de opbouw te laten kloppen.

IDAT waarde in source.png

Address: 00 0C 7F

Waarde: IDAT

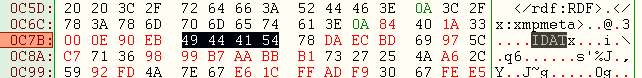


Figure 1 Source.png

IDA waarde in source.zif

Address: 00 E8 40

Waarde: IDA

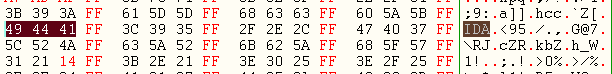


Figure Source.zif

ZIF file format aanduiding:

De eerste 4 bytes is de intro van het document

Address: 00 00 00

Waarde: 31 46 49 5A

Daarna kwamen we erachter dat de volgende 8 bytes de width en height van het plaatje aangeven. Deze byte stonden echter omgedraaid in het ZIF formaat.

De width van het plaatje

Address: 00 00 04

Waarde: 00 00 07 7A

De height van het plaatje

Address: 00 00 08

Waarde: 00 00 03 15

# How we found it

Na wat verder onderzoek vonden we de constructie veel lijken op een bmp file. De source.png file hebben we toen als BMP 24bit opgeslagen en hier de header en ender uit gehaald. Deze hebben we toen vervangen met de header van de ZIF file. Dit had wel invloed op de afbeelding, helaas was er geen afbeelding in te zien.

