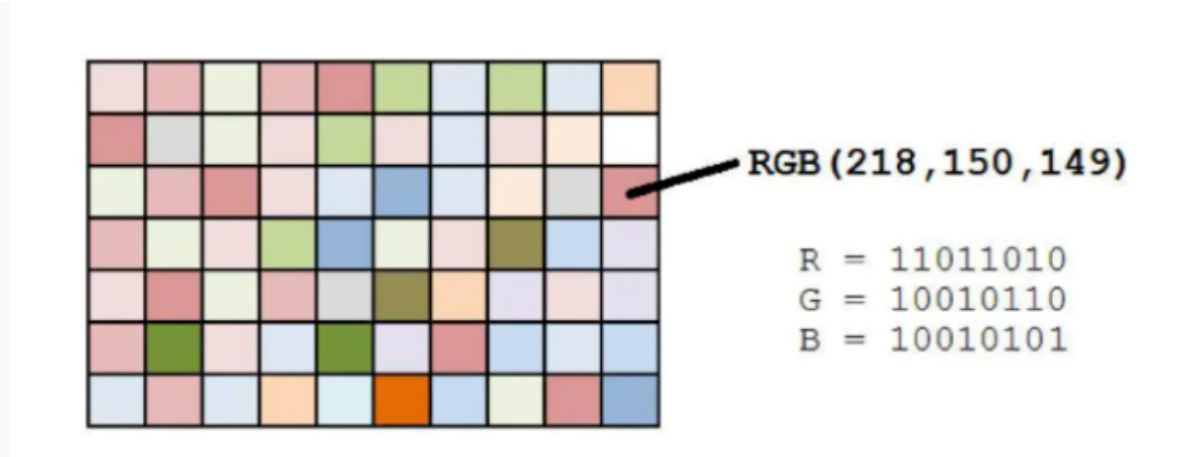





LSB隐写原理

LSB即为最低有效位 (Least Significant Bit, lsb) , 我们知道, 图片中的图像像素一般是由RGB三原色 (红绿蓝) 组成, 每一种颜色占用8位, 取值范围为0x00~0xFF, 即有256种颜色, 一共包含了256的3次方的颜色, 即16777216种颜色。而人类的眼睛可以区分约1000万种不同的颜色, 这就意味着人类的眼睛无法区分余下的颜色大约有6777216种。



LSB隐写就是修改RGB颜色分量的最低二进制位也就是最低有效位 (LSB) , 而人类的眼睛不会注意到这前后的变化, 每个像数可以携带3比特的信息

Color (Green)	Base 10	Binary	Change
	238	11101110	+3
	235	11101011	(base)
	232	11101000	-3

上图我们可以看到, 十进制的235表示的是绿色, 我们修改了在二进制中的最低位, 但是颜色看起来依旧没有变化。我们就可以修改最低位中的信息, 实现信息的隐写。

StegSolve工具

使用stegsolve打开图片, 按右方向键查看各通道显示的图像。

图像处理主要是analyse这个模块, 主要有这四个功能:

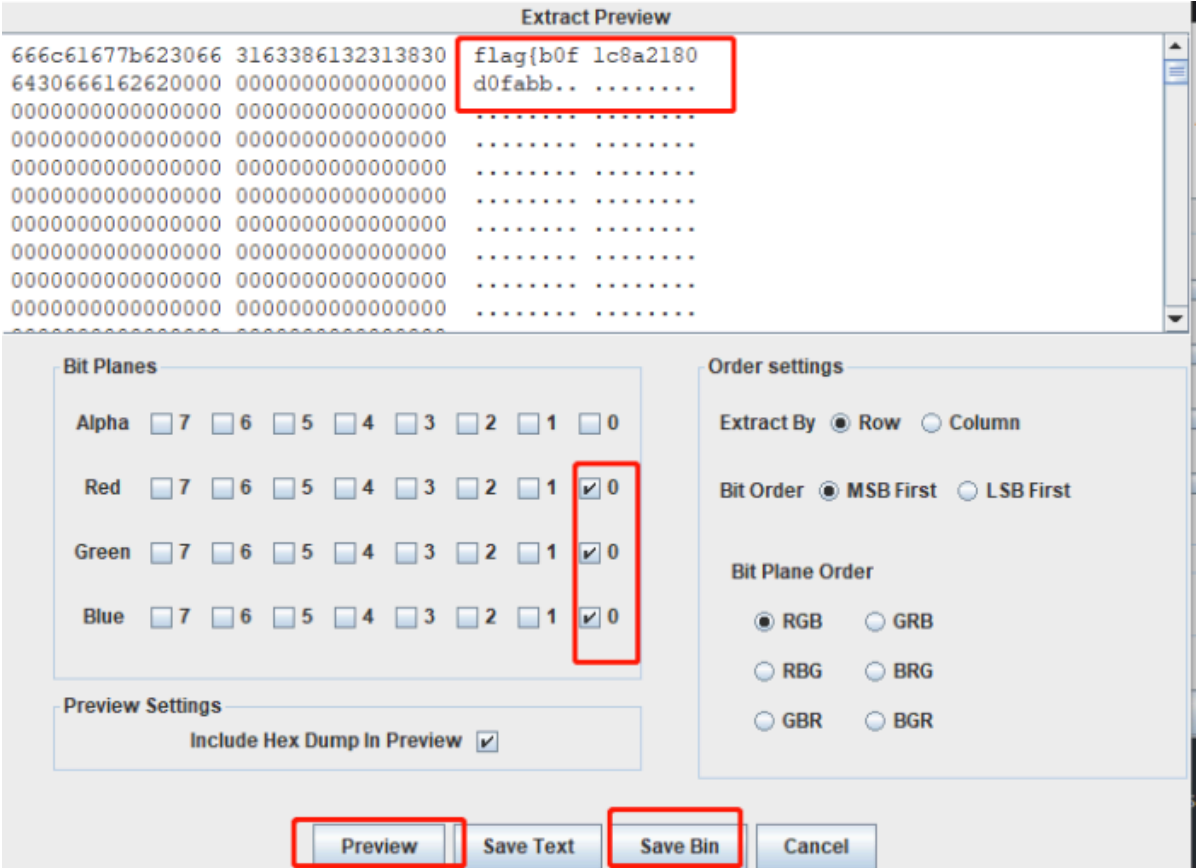
File Format: 文件格式, 查看图片的具体信息

Data Extract: 数据抽取, 提取图片中隐藏数据

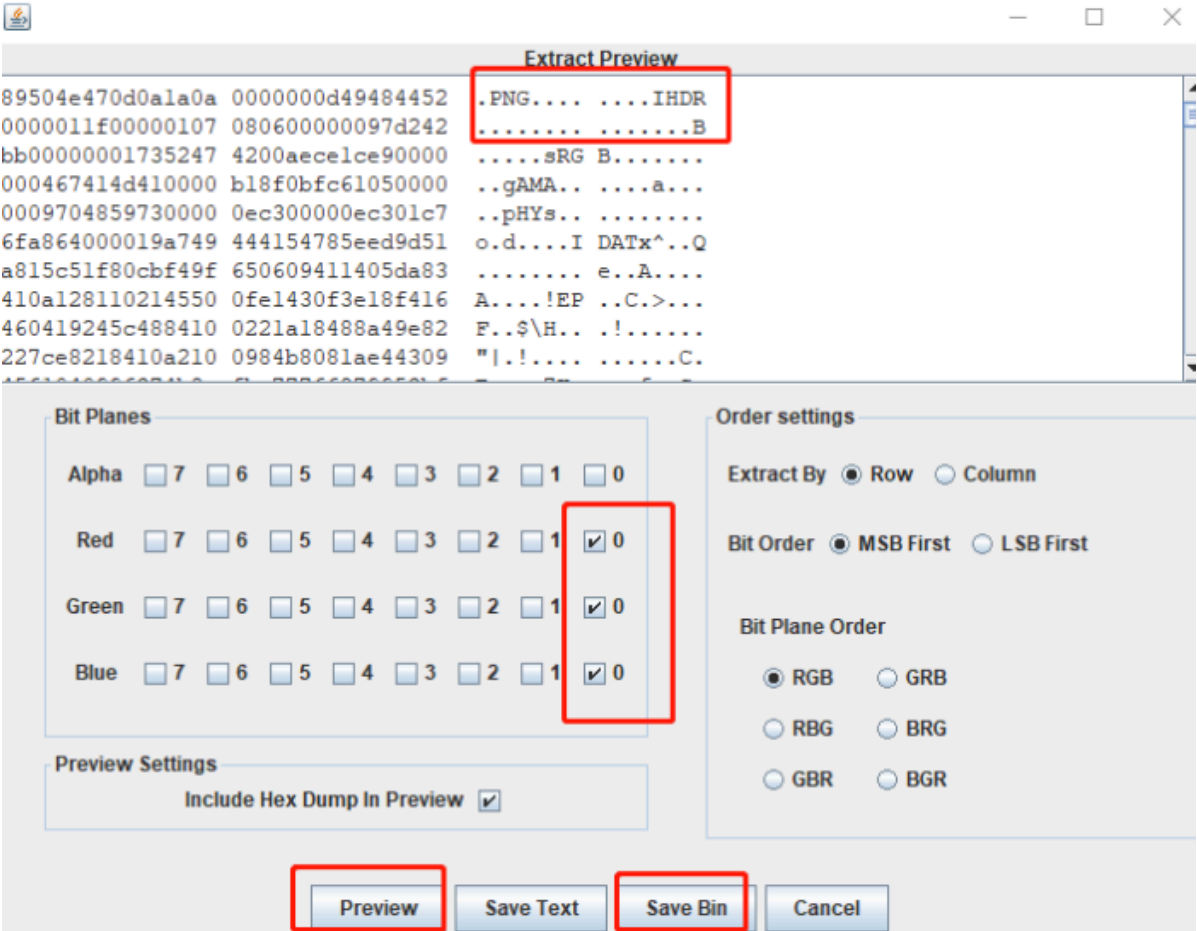
Frame Browser: 帧浏览器, 主要是对GIF之类的动图进行分解, 动图变成一张张图片

Image Combiner: 拼图, 图片拼接

对于LSB隐写的图片，我们用StegSolve打开模块，由于是RGB三原色的最低位隐写，所以在Data Extract模，提取Red，Green，和Blue的0通道信息，在这三个颜色的0通道上打勾，并按下Preview键，当隐写的内容为文本文件时如下所示：



当隐写的内容为图片时如下所示：



由PNG文件头可以看出隐写内容为PNG文件，按 save Bin 键保存为PNG文件

