




sun.misc.BASE64Encoder加密jdk1.8中Base64解密报错分析

原创

技术分子  于 2019-12-09 19:06:19 发布  960  收藏

分类专栏: [java基础原理](#) 文章标签: [Base64](#) [jdk1.8](#) [Base64](#)

版权声明: 本文为博主原创文章, 遵循 [CC 4.0 BY-SA](#) 版权协议, 转载请附上原文出处链接和本声明。

本文链接: <https://blog.csdn.net/u014205434/article/details/103463076>

版权



[java基础原理](#) 专栏收录该内容

5 篇文章 0 订阅

订阅专栏

sun.misc.BASE64Encoder一行不能超过76字符, 超过则添加回车换行符。

源码

```
public void encode(InputStream var1, OutputStream var2) throws IOException {
    byte[] var5 = new byte[this.bytesPerLine()];
    this.encodeBufferPrefix(var2);

    while(true) {
        int var4 = this.readFully(var1, var5);
        if (var4 == 0) {
            break;
        }

        this.encodeLinePrefix(var2, var4);

        for(int var3 = 0; var3 < var4; var3 += this.bytesPerAtom()) {
            if (var3 + this.bytesPerAtom() <= var4) {
                this.encodeAtom(var2, var5, var3, this.bytesPerAtom());
            } else {
                this.encodeAtom(var2, var5, var3, var4 - var3);
            }
        }

        if (var4 < this.bytesPerLine()) {
            break;
        }

        this.encodeLineSuffix(var2);
    }
}
```

```
//每次循环步长为3
protected int bytesPerAtom() {
    return 3;
}
//判断需要换行值
protected int bytesPerLine() {
    return 57;
}
//每次往var2中添加4个字节
this.encodeAtom
//println() 在Windows 回车换行符 (\r\n) ， 而Linux只有换行 (\n) ， Mac只有回车 (\r) )
var2字节等于76 (57/3*4) 会添加回车换行符

如果我们使用的是sun.misc.BASE64Decoder解密没有问题。
```

jdk1.8 Base64

使用方法

```
Base64.getDecoder().decode(str)
getEncoder()使用的是 RFC4648
public static Encoder getEncoder() {
    return Encoder.RFC4648;
}
static final Encoder RFC4648 = new Encoder(false, null, -1, true);最大长度-1没有分隔符。

我们使用 Base64.getDecoder().decode(str)解密将会报错
```

```

private int decode0(byte[] src, int sp, int sl, byte[] dst) {
    int[] base64 = isURL ? fromBase64URL : fromBase64;
    int dp = 0;
    int bits = 0;
    int shiftto = 18;    // pos of first byte of 4-byte atom
    while (sp < sl) {
        int b = src[sp++] & 0xff;
        if ((b = base64[b]) < 0) {
            if (b == -2) {    // padding byte '='
                // = shiftto==18 unnecessary padding
                // x= shiftto==12 a dangling single x
                // x to be handled together with non-padding case
                // xx= shiftto==6&&sp==sl missing last =
                // xx=y shiftto==6 last is not =
                if (shiftto == 6 && (sp == sl || src[sp++] != '=') ||
                    shiftto == 18) {
                    throw new IllegalArgumentException(
                        "Input byte array has wrong 4-byte ending unit");
                }
                break;
            }
        }
        if (isMIME) // skip if for rfc2045
            continue;
        else
            throw new IllegalArgumentException(
                "Illegal base64 character " +
                Integer.toString(src[sp - 1], 16));
    }
}

```

解密是会报错误

```

throw new IllegalArgumentException(
    "Illegal base64 character " +
    Integer.toString(src[sp - 1], 16));

```

错误提示如下：

java.lang.IllegalArgumentException: Illegal base64 character a