
Subject: BUG: plugins\zip
Posted by [kov_serg](#) on Wed, 20 Jan 2010 10:46:37 GMT
[View Forum Message](#) <> [Reply to Message](#)

I found simple bug in zip plugin for Upp. Please fix it.

Symptoms

Testing code:

```
void test_code() {  
    FileStream strm("test.zip",FileStream::CREATE);  
    Zip zip(strm);  
    zip.WriteFile("information","file1.txt");  
    zip.WriteFile("text 1","file2.txt");  
}
```

If we check archive we'll see damaged zip file

How to fix it:

In file uppsrc\plugin\zip\Zip.cpp

```
void Zip::WriteFile(const void *ptr, int size, const char *path, Gate2<int, int> progress, Time tm)  
{  
    File& f = file.Add();  
    f.path = UnixPath(path);  
    StringStream ss;  
    MemReadStream ms(ptr, size);  
    dword crc;  
    ZCompress(ss, ms, size, progress, true, &crc);  
    String data = ss.GetResult();  
    const void *r = ~data;  
    int csize = data.GetLength();  
    zip->Put32le(0x04034b50);  
    zip->Put16le(20);  
    zip->Put16le(0);  
    if(data.GetLength() >= size) {  
        r = ptr;  
        csize = size;  
        zip->Put16le(f.method = 0);  
    }  
    else  
        zip->Put16le(f.method = 8);  
    zip->Put32le(f.time = (tm.day << 16) | (tm.month << 21) | ((tm.year - 1980) << 25) |  
        (tm.hour << 11) | (tm.minute << 5) | (tm.second >> 1));  
    zip->Put32le(f.crc = crc);  
}
```

```

zip->Put32le(f.csize = csize);
zip->Put32le(f.usize = size);
zip->Put16le(strlen(f.path));
zip->Put16le(0);
zip->Put(f.path);
zip->Put(r, csize);
//done += 5 * 2 + 5 * 4 + data.GetCount() + f.path.GetCount(); // -patch
done += 5 * 2 + 5 * 4 + csize + f.path.GetCount(); // +patch
}
...
void Zip::Finish()
{
    if(!zip)
        return;
    dword off = done;
    dword rof = 0; // +patch
    for(int i = 0; i < file.GetCount(); i++) {
        File& f = file[i];
        zip->Put32le(0x02014b50);
        zip->Put16le(20);
        zip->Put16le(20);
        zip->Put16le(0); // general purpose bit flag
        zip->Put16le(f.method);
        zip->Put32le(f.time);
        zip->Put32le(f.crc);
        zip->Put32le(f.csize);
        zip->Put32le(f.usize);
        zip->Put16le(f.path.GetCount());
        zip->Put16le(0); // extra field length          2 bytes
        zip->Put16le(0); // file comment length        2 bytes
        zip->Put16le(0); // disk number start           2 bytes
        zip->Put16le(0); // internal file attributes    2 bytes
        zip->Put32le(0); // external file attributes    4 bytes
        //zip->Put32le(0); // relative offset of local header 4 bytes // -patch
        zip->Put32le(rof); // relative offset of local header 4 bytes // +patch
        rof+=5 * 2 + 5 * 4 + f.csize + f.path.GetCount(); // +patch
        zip->Put(f.path);
        done += 7 * 4 + 9 * 2 + f.path.GetCount();
    }
    zip->Put32le(0x06054b50);
    zip->Put16le(0); // number of this disk
    zip->Put16le(0); // number of the disk with the start of the central directory
    zip->Put16le(file.GetCount()); // total number of entries in the central directory on this disk
    zip->Put16le(file.GetCount()); // total number of entries in the central directory
    zip->Put32le(done - off); // size of the central directory
    zip->Put32le(off); //offset of start of central directory with respect to the starting disk number
    zip->Put16le(0);
    zip = NULL;
}

```

}