

What exactly causes binary file "gibberish"?

Asked 16 years, 2 months ago Modified 4 years, 11 months ago

Viewed 28k times



25

I haven't found an answer to this particular question; perhaps there isn't one. But I've been wondering for a while about it.



What exactly causes a binary file to display as "gibberish" when you look at it in a text editor? It's the same thing with encrypted files. Are the binary values of the file trying to be converted into ASCII? Is it possible to convert the view to display raw binary values, i.e. to show the 1s and 0s that make up the file?

Finally, is there a way to determine what program will properly open a data file? Many times, especially with Windows, a file is orphaned or otherwise not associated w/ a particular program. Opening it in a text editor sometimes tells you where it belongs but most of the time doesn't, due to the gibberish. If the extension doesn't provide any information, how can you determine what program it belongs to?

binaryfiles

Share

Improve this question

Follow

edited Oct 23, 2015 at 18:18



Brian Tompsett - 汤莱恩

5,875 ● 72 ● 61 ● 133

asked Oct 19, 2008 at 5:46



crystalattice

5,089 ● 12 ● 43 ● 57

7 Answers

Sorted by:

Highest score (default)



- ***Are the binary values of the file trying to be converted into ASCII?***

22



Yes, that's exactly what's happening. Typically, the binary values of the file also include ASCII control characters that aren't printable, resulting in even more bizarre display in a typical text editor.



- ***Is it possible to convert the view to display raw binary values, i.e. to show the 1s and 0s that make up the file?***



It depends on your editor. What you want is a "hex editor", rather than a normal text editor. This will show you the raw contents of the file (typically in hexadecimal rather than binary, since the zeros and ones would take up a lot of space and be harder to read).

- ***Finally, is there a way to determine what program will properly open a data file?***

There is a Linux command-line program called ["file"](#) that will attempt to analyze the file (typically looking for common header patterns) and tell you what sort of file it is (for example text, or audio, or video, or XML, etc). I'm not sure if there is an equivalent program for Windows. Of course, the output of this program is just a guess, but it can be very useful when you don't know what the format of a file is.

Share Improve this answer

edited Dec 21, 2017 at 18:15

Follow



Ivan Aracki

5,301 ● 11 ● 66 ● 84

answered Oct 19, 2008 at 5:57



Ross

9,878 ● 9 ● 38 ● 36

The file command has been ported to Windows; you can find it for instance on Cygwin. – [CesarB](#) Oct 19, 2008 at 15:21

And anyway `file` is surely not a Linux program--it's a *nix program, and may be on other systems as well. Solaris has had it for many years. – [John Zwinck](#) Nov 25, 2008 at 2:42

The (well, one) Windows port of 'file' is here:
gnuwin32.sourceforge.net/packages/file.htm – [sleske](#) Feb 12, 2009 at 9:33



5

A binary file appears as gibberish because the data in it is designed for the machine to read and not for humans. Sadly, some of us get used to interpreting gibberish -



albeit with somewhat specialized tools to help see the data better - but most people should not need to know.



Each byte in the file is treated as a character in the current code set (probably CP1252 on Windows). Byte value 65 is 'A', for example; you can find illustrative examples easily on the web. So, the bytes that make up the binary data are displayed according to the code set - as best as the text editor can. It doesn't try to convert the binary - it doesn't know how (only the original program does).

As to how to detect what program created the file - you may be able to do that sometimes, but not easily and reliably. On Unix (or with Cygwin on Windows) the 'file' program may be able to help. This program looks at the first few bytes to try and guess the program.

Encrypted data is supposed to look like gibberish. If it doesn't look like gibberish, then it probably isn't very well encrypted.

Share Improve this answer

Follow

answered Oct 19, 2008 at 5:58



[Jonathan Leffler](#)

752k ● 145 ● 946 ● 1.3k

7 Ha ha, "some of us get used to interpreting gibberish" - back in the stone age I remember helping a customer with his (4800 baud) modem connection problems. I had him describe, over the phone, the garbage he was seeing on his end and I immediately identified the fix for his problem. Kind of frightening to think about it now.... – [NVRAM](#) Jul 2, 2009 at 17:46



3

The display looks interesting, because a binary file can contain non-printable characters. It is up to the displaying program to replace such characters with something else.



This can be prevented by using a hex editor. Such a program displays each byte from the file as its hexadecimal value. That makes for a nice tabular view of the file, but it is not easy for the average person to decipher this view, because we are not used to look at data that way.



There are a few ways to find out what program a file might belong to. You can look at the beginning of the file and with some knowledge, you might recognize the file type. There are some types that begin with the same characters (RAR, GIF etc.). For other types it might not be as easy.

In Linux you can use the "file" command to help you determine file type. There are probably programs for Windows that will do the same.

answered Oct 19, 2008 at 5:53



HS.

2,615 ● 2 ● 21 ● 29



2



Binary data is often very random. Encrypted data in particular, by definition. Each byte can be represented by one of 256 characters (leaving Unicode out of the equation). ASCII only covers 128 of these, and only 94 of these are actual printable characters. Outside the ASCII range, you have a number of international characters and strange symbols. There are certainly more than 128 of these, so one must specify a codepage to select a specific set of symbols.

Anyway, since binary files can be represented as a very random assortment of familiar and unfamiliar characters, the file will look like gibberish if you open it in an editor.

You could always open a file (binary or text file, there really is no difference) in a hex editor, and look at the raw binary data.

There is no way to tell which program created a specific file. In particular, if the program has encrypted its data, all hope is lost. Otherwise, it is often easy to recognize certain "signatures."

Follow



Tor Hovland

1,649 ● 2 ● 15 ● 23



2



The reason files that are binary display as gibberish when viewed in standard text editors such as notepad is because when displayed with the encodings commonly used by these types of applications (e.g. ASCII or UTF-8) the data is mapped to characters when it is encoded for display, the output of this process generally makes as little sense to humans as the binary data being mapped, ergo the gibberish you see

As previously mentioned these files make more sense when viewed in a different way such as with a hex editor.

Certain file types can be recognized by data present in all files of a given type, for example all executable files (*.exe) begin with the letters MZ

Share Improve this answer

Follow

edited Jan 10, 2020 at 16:41



Brian FitzGerald

3,109 ● 3 ● 30 ● 39

answered Oct 19, 2008 at 5:59



Crippledsmurf

4,012 ● 1 ● 33 ● 51



0

Yes, Wordpad and Notepad and many other text editors assume that any file you open with it is a text file and will try to display the ASCII characters represented by the bytes in the file.



[Hex Editors](#) are made to view and edit binary files. They usually display each byte as a pair of hexadecimal digits instead of "1s and 0s" because it's easier to read that way.

Share Improve this answer

answered Oct 19, 2008 at 5:54

Follow



[Paige Ruten](#)

176k ● 37 ● 182 ● 199



0



A text editor makes very few assumptions about the data coming into it, besides things like character encodings.

Thus, it will (as you say) read the file's data as ASCII and display it that way. Since binary data doesn't always fall within the alphanumeric range, you get gibberish. As for showing the raw binary values, you need a hex editor like [XVI32](#).

Binary files often have no context outside of the program that uses them. Some binary formats contain a 4-byte magic sequence at the beginning (for example, Java .class files start with "CAFE"), but to recognize them without their program, you need a mapping of those 4-byte sequences. I believe some Linux distros contain this information for a wide variety of binary formats and will examine the beginning of the file to attempt to identify it. Other than that, there's not much you can do.

Share Improve this answer

answered Oct 19, 2008 at 5:56

Follow



Ben Leedom