

Ruby - Bug #19384

ASCII 128..154 characters in IO.popen or %x output do not reflect the proper encoding in Windows

01/26/2023 09:17 PM - stringsn88keys (Thomas Powell)

Status:	Open	
Priority:	Normal	
Assignee:		
Target version:		
ruby -v:	3.1.3	Backport: 2.7: UNKNOWN, 3.0: UNKNOWN, 3.1: UNKNOWN, 3.2: UNKNOWN

Description

Operating systems: Windows 10 and Windows Server 2022 (likely all recent versions of Windows)
Ruby: confirmed on 2.7.7 through 3.1.3

On macOS and Linux I can create a file named "ÇüéääââçëèëïîĀĀÉæÆôöòûÿÖÜ" and then do a directory listing via IO.popen or %x and find the file name in the output string.

In Windows, while the encoding is reported as [#Encoding:UTF-8](#), I have to .force_encoding on the output to be able to find the string in the output:

%x|dir tmp|

output encoding: [#Encoding:UTF-8](#)
Output can be made to match by forcing the following encodings:
IBM437
CP850
IBM865

IO.popen(dir tmp).read

output encoding: [#Encoding:UTF-8](#)
Output can be made to match by forcing the following encodings:
IBM437
CP850
IBM865

But on macOS or Linux:
❏ ruby directory_test.rb

%x|ls tmp|

output encoding: [#Encoding:UTF-8](#)
output matches without forcing encoding
Output can be made to match by forcing the following encodings:
UTF-8
UTF8-MAC
CESU-8
UTF8-DoCoMo
UTF8-KDDI
UTF8-SoftBank

IO.popen(ls tmp).read

output encoding: [#Encoding:UTF-8](#)
output matches without forcing encoding
Output can be made to match by forcing the following encodings:
UTF-8
UTF8-MAC
CESU-8
UTF8-DoCoMo

UTF8-KDDI
UTF8-SoftBank

Note:
The example is contrived because the actual IO.popen output is from a customer system with umlaut characters. However, I have found creating a filename with these characters will adequately reproduce the issue.
Also, I'm only using ASCII/IBM437 as an encoding to create a contiguous set of characters, "ÇüéääåçêëèïîĂĂÉæÆôðòûÿÖÜ" as a contrived example.

Files			
directory_test.rb	1.14 KB	01/26/2023	stringsn88keys (Thomas Powell)