Article

[John Murray](#) · Mar 6, 2016   2m read

Who does Windows think I am?

When my COS code is executing in a Caché process it might want to interact with the host operating system. For the purpose of this post I'm focusing on a Windows host, but much of it applies to other host OS platforms as well.

A common example of host OS interaction is when my process wants to read from or write to a file. What credentials will apply when Windows is checking whether or not to allow me access to the file?

To answer that we need to consider another question. How did our process start?

If we're in an interactive terminal session, perhaps we launched this from the Caché cube on the server's desktop. In such a case we will already have authenticated to Windows to get onto the server's desktop, so our Caché process can run with those Windows credentials.

Or our interactive terminal session might be a telnet one, where we're using a telnet client app such as Caché Telnet. The telnet client communicates with the Caché server's telnet service over TCP/IP. The server has no idea what Windows credentials you're running the telnet client with. Indeed, you might not be running the telnet client on Windows at all. So your Caché server process runs under the same Windows account as the Caché instance started up with. Caché runs as a Windows service, and part of a service definition is what account the service should "log on as".

When Caché is installed using the Minimal security option, its service is set up to run under the Local System account of Windows. While this may be adequate, one key limitation is that this account can never access Windows networking fileshares, e.g. \\FileServer\Team\ProjectSpec.xml.

So if your code works when run from a terminal session on the server, but fails when run from a telnet session, or from a Studio add-in, or from a web application, you might be running up against the issue of "Who does Windows think I am?".

UPDATE: Starting at 2015.2 (I think), it is important to use the cinstall.exe utility (located in your installation's bin subdirectory) to change which account the service runs as:

```
cinstall setserviceusername <InstanceName> <username> <password>
```

If you don't do this but instead only change the setting in the Log On tab of the Windows service, then in certain circumstances the $ZF() functions may return a -1 failure code. See http://docs.intersystems.com/latest/csp/docbook/DocBook.UI.Page.cls?KEY=GCNVC147006 for more details.

[#Caché](#) [#Terminal](#) [#Ensemble](#) [#Callout](#) [#Microsoft Windows](#)