

# Re: Remote launch fails with E\_ACCESSDENIED

---

*Source:*

<http://www.tech-archive.net/Archive/Development/microsoft.public.win32.programmer.ole/2008-04/msg00089.html>

---

- *From:* "Paul Baker [MVP, Windows Desktop Experience]" <[paulrichardbaker@xxxxxxxxxxxxxxxxxxxx](mailto:paulrichardbaker@xxxxxxxxxxxxxxxxxxxx)>
  - *Date:* Tue, 29 Apr 2008 10:15:07 -0400
- 

As you would probably guess, CoGetClassObject also fails with E\_ACCESSDENIED.

If I replace my EXE with a copy of Notepad.exe, it also fails to launch that.

Paul

"Paul Baker [MVP, Windows Desktop Experience]" <[paulrichardbaker@xxxxxxxxxxxxxxxxxxxx](mailto:paulrichardbaker@xxxxxxxxxxxxxxxxxxxx)> wrote in message news:...

Gerry,

Thanks. That script works (Agent Version 7.0.6000.381).

I am not sure what details of my DCOM server are relevant. I will offer some further details and maybe you have specific questions?

These are 32-bit systems and 32-bit Win32 processes and the server is out of process. It has a user interface, but can also function through a command line interface or DCOM server in a non-interactive window station. I have recently moved some interactive functions from the server to the client to facilitate running the server non-interactively without loss of functionality. DCOM does not even launch the process, so I won't go into the details of classes, interfaces, etc. at this point.

Paul

"Gerry Hickman" <[gerry666uk@xxxxxxxxxxxxxxxxxxxx](mailto:gerry666uk@xxxxxxxxxxxxxxxxxxxx)> wrote in message news:[OyPGhnXqIHA.4876@xxxxxxxxxxxxxxxxxxxxxxxxxxxx](mailto:OyPGhnXqIHA.4876@xxxxxxxxxxxxxxxxxxxxxxxxxxxx)

Hi Paul,

Before going any further, can you try running this simple code from the client – look out for word wrap!

```
// TestDCOM.js
// Get the Agent Version
```

Re: Remote launch fails with E\_ACCESSDENIED

```
var strCmp = "YourServer";  
var oAgentInfo = new ActiveXObject("Microsoft.Update.AgentInfo",  
strCmp);  
var strVersion = oAgentInfo.GetInfo("ProductVersionString");  
WScript.Echo("Agent Version " + strVersion);
```

If this works, it means both computers are configured for DCOM and remoting and points to a problem specific to the object you are trying to launch – in that case we'd need full details of the object.

Paul Baker [MVP, Windows Desktop Experience] wrote:

CoCreateInstance with a remote server name is returning E\_ACCESSDENIED before it even launches the remote server process. The client is Windows XP SP2 and the server is Windows Server 2003. I am not calling CoInitializeSecurity. The user is an administrator on both client and server and the DCOM configuration and security settings are default, as far as I know, and nothing looks wrong.

It works if I configure the AppID on both the client and server to run as the "interactive user" (use DCOMCNFG) and the user is logged on to the client and server console sessions. However, I get E\_ACCESSDENIED if I configure the AppID on both the client and server to run as "the launching user", which is what I want (to avoid having to logon to the console session or expose it to an interactive user).

Can anyone offer a suggestion on how to troubleshoot this?

Thanks,

Paul

--  
Gerry Hickman (London UK)

Re: Remote launch fails with E\_ACCESSDENIED