

Re: How to set up DCOM properly to allow server and client connect remotely

Source:

<http://www.tech-archive.net/Archive/Development/microsoft.public.win32.programmer.ole/2006-11/msg00043.html>

- *From:* "Alexander Nickolov" <agnickolov@xxxxxxxx>
 - *Date:* Wed, 8 Nov 2006 17:43:16 -0800
-

Well, frankly your problem is not using a domain. Win NT has a feature where two machines can cross-authenticate their accounts if said accounts have the same username and password. This feature is disabled by default on Win XP. Change the following local security policy setting: "Network access: Sharing and Security Model for Local Accounts" to Classic mode. Then make sure your accounts on the two machines are indeed the same with regard to their username and password.

=====
Alexander Nickolov
Microsoft MVP [VC], MCS
email: agnickolov@xxxxxxxx
MVP VC FAQ: <http://www.mvps.org/vcfaq>
=====

"Atiz" <fleetfoot18@xxxxxxxx> wrote in message
news:1163034619.673494.323270@xx

Hi all,

I need to know how to set up DCOM properly so the OPC server and client can connect to each other.

Details:

- server and client on separate pcs.
- pcs not in any domain.
- pcs installed with windows XP SP2.
- both server and client are .exe files.
- both are assigned same local user accounts

What I have done so far:

server side:

Re: How to set up DCOM properly to allow server and client connect remotely

Gave exception to server.exe, DCOM port 135 in Windows firewall. in dcomcnfg, I found the server's id, and changed the properties as well as for "my computer" .

I have set default authentication level to connect, impersonation level to identify.

For both access and launch permissions, I have set the limits and defaults to include administrators, users, anonymous logon, system and local user (the user assigned specially for the server)

client side:

Gave exception to client.exe, DCOM port 135 in Windows firewall. in dcomcnfg, is also the same as the above. Only difference is that the properties are set for "my computer".

Btw, when I edit say, access permissions and I want to add the assigned user account used for the client, when I click on the location button, I can only find the pc's own local user acct. It will show as pc1_name/username.

Similarly for the server side, I can only add pc2_name/username.

The server and the client will only deal with the username and NOT the pc's name when trying to connect right?

So where have I gone wrong? Btw, in my client, I have the following code (some snippets):

I have set the coAuthIdentity to only have the local user acct and not the domain(since my pcs are not in any domain).

```
CoInitializeSecurity(NULL, -1, NULL, NULL, RPC_C_AUTHN_LEVEL_CONNECT,
RPC_C_IMP_LEVEL_IMPERSONATE, NULL, EOAC_NONE, NULL);
```

```
COAUTHINFO coAuthInfo;
COSERVERINFO remoteServerInfo;
COAUTHIDENTITY coAuthIdentity;
HKEY remoteRegHandle;
HKEY keyHandle;
char classIdString[100];
CLSID clsid;
MULTI_QI reqInterface;
CString keyName = "SOFTWARE\\Classes\\" + serverName + "\\Clsid";
```

```
result = RegConnectRegistry(hostName, HKEY_LOCAL_MACHINE,
&remoteRegHandle);
```

```
if (SUCCEEDED(result)){
result = RegOpenKeyEx(remoteRegHandle, keyName, 0, KEY_READ,
&keyHandle);
if (SUCCEEDED(result)){
DWORD entryType;
```

```
unsigned bufferSize = 100;
result = RegQueryValueEx(keyHandle, NULL, 0, &entryType,
```

Re: How to set up DCOM properly to allow server and client connect remotely

```
(LPBYTE)&classIdString, (LPDWORD)&bufferSize);
if (FAILED(result)){
printf("here");
}else{
USES_CONVERSION;
LPOLESTR classId = A2W(classIdString);
if (CLSIDFromString(classId,&clsid) != S_OK){
printf("Failed");
}
}
}
}

ZeroMemory(&coAuthIdentity, sizeof(coAuthIdentity));
coAuthIdentity.Flags = SEC_WINNT_AUTH_IDENTITY_UNICODE;
coAuthIdentity.User = (USHORT*)L"user";
coAuthIdentity.UserLength = wcslen(L"user");
coAuthIdentity.Password = (USHORT*)L"password";
coAuthIdentity.PasswordLength = wcslen(L"password");

ZeroMemory(&coAuthInfo, sizeof(COAUTHINFO));
coAuthInfo.dwAuthnLevel = RPC_C_AUTHN_LEVEL_CONNECT;
coAuthInfo.dwAuthnSvc = RPC_C_AUTHN_WINNT;
coAuthInfo.dwAuthzSvc = RPC_C_AUTHZ_NONE;
coAuthInfo.dwCapabilities = EOAC_NONE;
coAuthInfo.dwImpersonationLevel = RPC_C_IMP_LEVEL_IMPERSONATE;
coAuthInfo.pAuthIdentityData = &coAuthIdentity;
coAuthInfo.pwszServerPrincName = NULL;
```

When client is runned, the "Failed" message will appear. Which I think means the client does not have permission to access the server.

I need help in this area, hope someone can help me, thanks.

Atiz