Matrikon®
Powering Interoperability

Windows Security Configuration for OPC Classic Communication

Matrikon®
Powering Interoperability

Document Revision History

Date	Document Version	Description	Author
2013-01-14	1.0	Initial document.	TS
2013-04-05	1.1	Revisions.	TS
2014-08-18	1.2	Revisions.	TS
2020-07-30	2.0	Revisions and update	CC

DOCUMENT VERSION

Version: 2.0

COPYRIGHT INFORMATION

CONFIDENTIAL

The information contained herein is confidential and proprietary to Matrikon Inc. It may not be disclosed or transferred, directly or indirectly, to any third party without the explicit written permission of Matrikon Inc.

LIMITATIONS

Matrikon has made its best effort to prepare this manual. Matrikon makes no representation or warranties of any kind with regard to the completeness or accuracy of the contents herein and accepts no liability of any kind including without limitation warranties of merchantable quality, satisfactory quality, merchantability and fitness for a particular purpose on those arising by law, statute, usage of trade, course of dealing or otherwise. Matrikon shall not be liable for any losses or damages of any kind caused or alleged to be caused directly or indirectly from this manual.

LICENSE AGREEMENT

This document and the software described in this document are supplied under a license agreement and may only be used in accordance with the terms of that agreement.

TRADEMARK INFORMATION

The following are either trademarks or registered trademarks of their respective organizations:

MatrikonOPC™ is a division of Matrikon™ Inc. Matrikon and MatrikonOPC are trademarks or registered trademarks of Matrikon Inc.

Adobe, the Adobe logo, Acrobat, the Adobe PDF logo, Distiller and Reader are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries.

Matrikon®
Powering Interoperability

Table of Contents

Matrikon®
Powering Interoperability

Table of Figures

Matrikon®
Powering Interoperability

Introduction

The connection between an OPC client and an OPC server is one of the most basic functions of all OPC applications, and yet it is also one of the most common issues experienced in commissioning of a new system, modifying or adding a new component to an existing system, or simply troubleshooting a production system. In order to successfully resolve issues involving application connection it is important to understand the basic components of the connection process and how they affect connection.

All OPC communication is based on the proprietary COM technology integral to the Windows operating system. In fact, the connection between OPC applications is defined not by the OPC specification, but by COM. The connection process is therefore a COM process and is subject to the security apparatus in Windows. As most systems where OPC applications are employed are distributed systems and depend on various networking technologies, these must also be examined to ensure that they are not responsible for the lack of connectivity.

This document presents a process for ensuring that the connection between OPC applications is not compromised by the security configuration of either the Windows platforms on which they are installed or the network apparatus that connects them. For the most part this will involve reducing the various security mechanism to near non-existence. As this can pose serious security risks in itself, we will work from the inside out to ensure that system-wide security is minimally impacted.

Platform Compatibility

This document, and the screenshots included within, apply to the Windows 10/ Windows Server 2012 R2 platform. Although the procedures described are applicable for earlier versions, access to the configuration tools may very slightly. For older versions pre-Windows XP/Server 2003, some of these may not apply. Should that be the case, consult your user documentation, your IT team, or your software vendor's support team.

Group Policy Objects

A Group Policy Object (GPO) is a collection of policy settings that allows administrators to manage a Windows system using Active Directory Directory Services (AD DS) and security group membership. Simply put, it allows administrators to control access to security configuration and resources on a network-wide (domain-wide) basis. These settings will override the Local Security Policy settings on a Domain member when conflict between the two arises.

If you have completed all of the steps outlined in this document and are still unable to connect to your OPC server, there may be GPOs in place that are affecting OPC connectivity. There is, unfortunately, no easy process for troubleshooting this. You need to consult with your IT Department / System Administrator to analyze the GPOs configured on your system to ascertain these effects.

User Account Control

User Account Control (UAC) is an integral component of the Windows Security framework that helps prevent malware from damaging a PC or its operating system. It accomplishes this by limiting the security context of apps running on the system to a non-administrator account unless specifically authorized by an administrator. Elevating an app's security level allows it to perform tasks that require administrator authority.

To force a permission elevation, you can right-click on an icon and choose **Run as administrator**; if the launching user is not part of the Administrators group you will be prompted for administrator credentials.

As MatrikonOPC products require access to protected parts of the filesystem to perform certain tasks, we strongly recommend using Run as administrator to install and license MatrikonOPC software and run MatrikonOPC configuration utilities. Please note that access to all of the utilities and configuration tools described in this document require an account with local administrator privileges.

Identities

Whether configuring who has access to the server, or for determining which permissions the server will have to access other resources, the identities associated with running process play a critical role in Windows and in OPC functionality.

Client Identities

These are special identities in Windows that are added to the security settings on the COM (OPC) servers to determine who has access to the server.

Everyone – all authenticated users on the local machine. This group applies to local users only.
Interactive – any user logged on to the local system. This identity allows only local users to access a resource.
Network – all users logged in through a network connection. This identity allows only remote users to access a resource.
System – a service account used by the operating system.

Any user account or group can be added to the server's security settings. Refer to the Security tab Custom DCOM Permissions section of this document. For additional information on Special Accounts in Windows refer to Microsoft Docs.

Server Identities

These are the accounts used to run the COM (OPC) servers.

Launching User – server runs with the identity of the user that launched the instance of the server. Each user (client) runs its own instance of the server. This can cause issues with server access and resource usage.
Interactive User – server runs with the identity of the logged in user. All clients connect to the same instance of the server, but if no user is logged on the server will not run.
System Account – only available if server is registered as a service. All clients connect to a single instance of the server. Normally a safe choice, unless there are restrictions on the resource the server is attempting to access.
This User – an account specified by the administrator that meets all security requirements of the resource to be accessed.

These identities are configured in the Identity tab of the Custom DCOM Permissions for each server object. For further information on these identities refer to Microsoft Docs.

Server Listing

When a client is launched, it should list all available servers in the target machine. This list is generated either directly from the registry or from a utility from the OPC Foundation called the OPC Server List utility, or OPC Enumerator *(OPC Enum for brevity)*. This is a COM server that manages server name and supported specification information for locally installed OPC servers. If this list is not available, it is possible that the permissions on this utility are not properly configured. To set these permissions, refer to the Custom DCOM Security Settings.

Matrikon®
Powering Interoperability

Security Configuration

Starting with the most basic security settings, this document examines;

1. **Data Execution Prevention** (DEP) which prevents unauthorized code from running in protected system memory areas.
2. The **DCOM Security Settings** that determine which identities have permission to interact with COM (OPC) objects.
3. The **Local Security Policy Options** that control access to the machine in a workgroup environment.
4. The **Windows Firewall**.
5. Security Configuration and Network Components

Data Execution Prevention (DEP)

Data Execution Prevention (DEP) is a set of hardware and software technologies that perform additional checks on memory to help prevent malicious code from running on a system. In Windows versions including and after Windows 7 and Windows Server 2008, DEP is enforced by hardware and software.

DEP will also prevent many installations from running and has been known to cause other software issues. It should be noted that the effects of DEP are known and that many modern applications are no longer affected by DEP. MatrikonOPC software released from late 2006 to July 2009 will detect the DEP setting and, if set to **Turn on DEP for all programs and services except those I select** (On), terminate the installation process. MatrikonOPC software released since August 2009 no longer requires DEP to be turned off. To verify this, or for non-Matrikon software, consult the release notes and user manual for each application.

If DEP is turned ON in your system and you believe that it may indeed be responsible for the bad behaviour exhibited by your applications, the following procedure details how to turn DEP OFF. Please note that for this to be effective, DEP must be turned off during software installation. If DEP is ON and the software has been installed, you will need to turn DEP OFF, restart the machine, uninstall the software, and then perform a fresh install of the software. A machine restart may also be required after the uninstall.

To turn DEP OFF, perform the following steps:

1. Click on the **Start** button, type *computer properties* or right-click on **Computer** (**This PC**) and select **Properties**.
2. On the left side of the dialogue, click on the link for **Advanced system settings**. Accessing this item requires Local Administrator privileges and may trigger a UAC prompt. Respond appropriately.

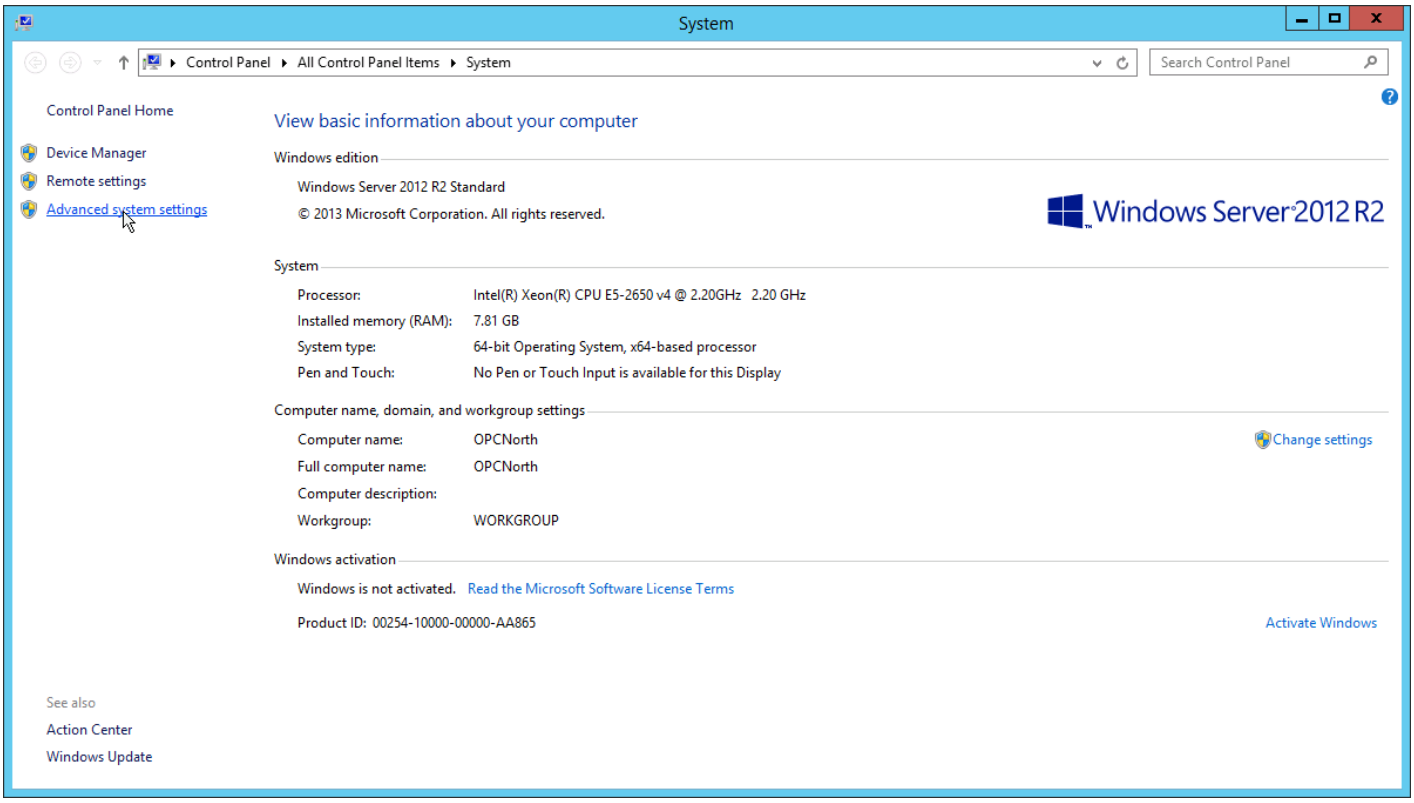

Figure 1 System Properties

3. Select the Advanced tab and click on Settings in the Performance area.

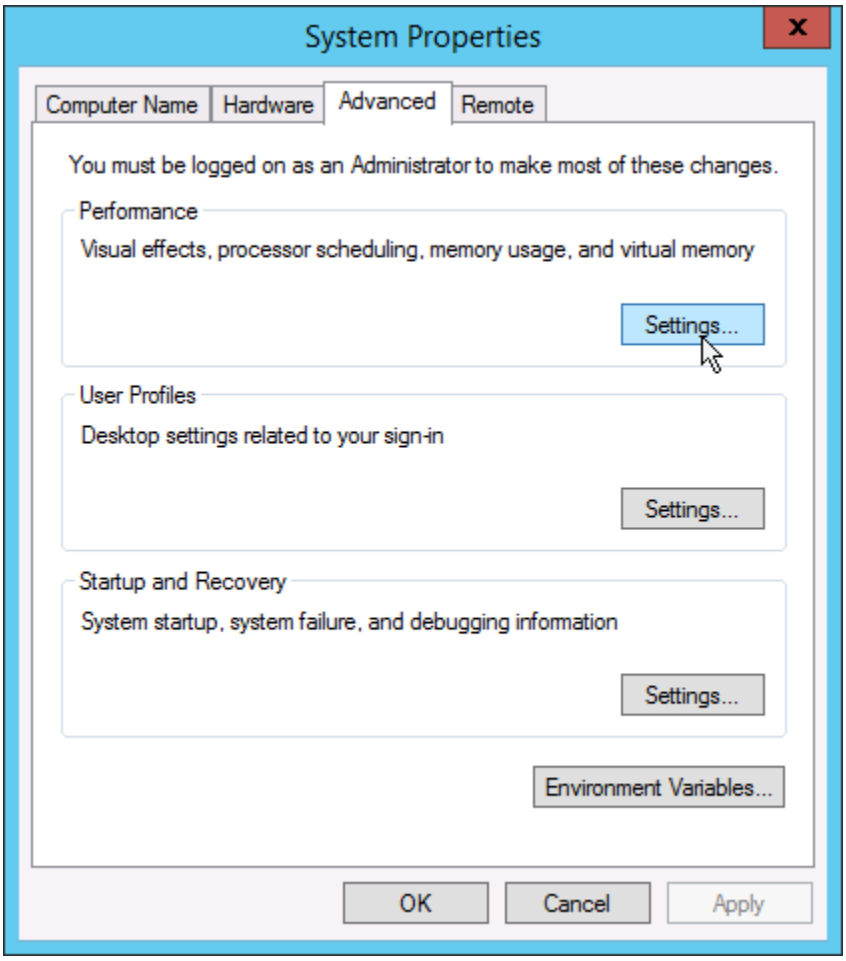

Figure 2 System Properties Advanced Settings

4. In the **Performance Options** dialogue, on the **Data Execution Prevention** tab, select the **Turn on DEP for essential Windows programs and services only** option. This is the setting we refer to as *OFF*.

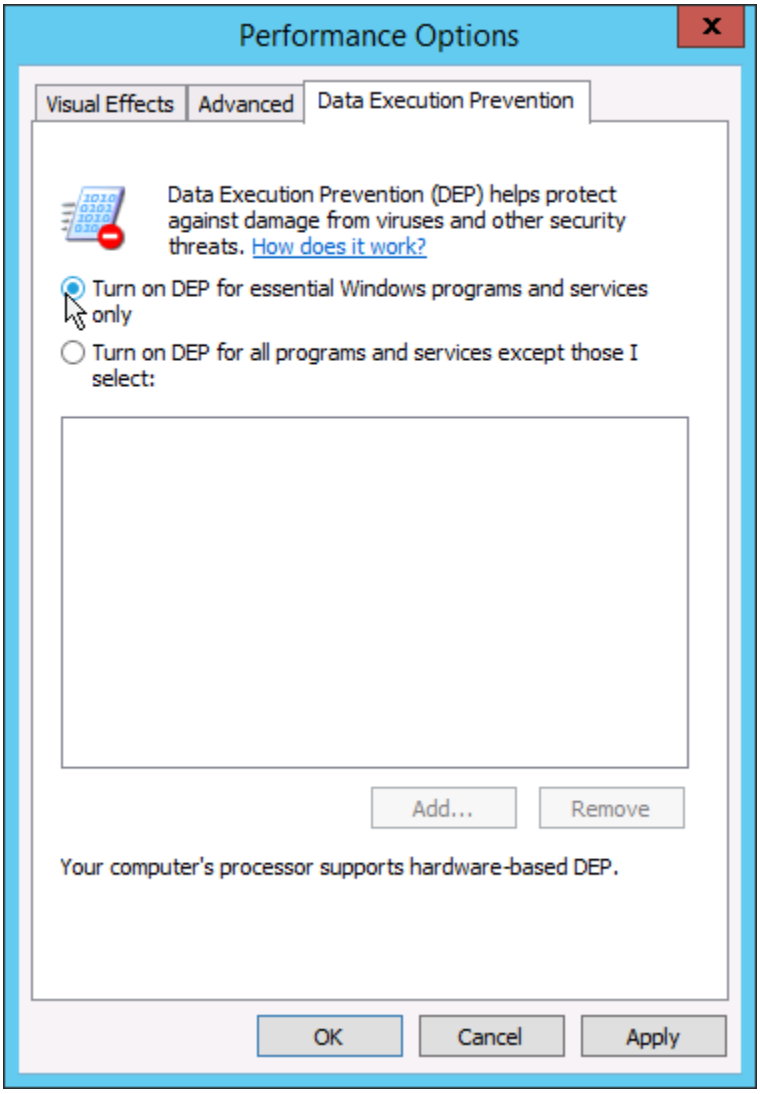

Figure 3 System Properties - DEP Settings

5. Click **OK**. If you changed the setting, it will be necessary to restart the operating system.

DCOM Security Settings

The Windows platform implements a set of permissions affecting all COM applications as a series of Access Control Lists (ACLs). These permissions will be in effect for each OPC client and server application unless it sets its own security settings, or a set of custom security settings is created for it via the ***dcomcnfg*** tool.

In order for an OPC client to connect to and communicate with an OPC server, these security settings must be configured to allow the client identity to interact with the server object. The first step in troubleshooting whether these settings are affecting connectivity is to determine if the server is using the default settings or custom settings. This can be determined by checking the DCOMCNFG properties for the OPC server.

1. Click on the **Start** button and type in *dcomcnfg*.

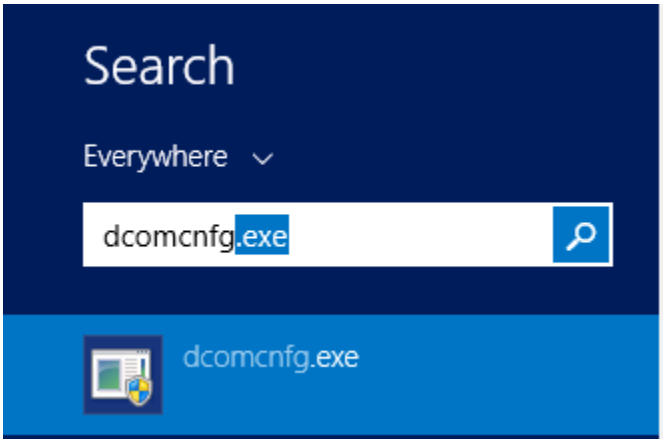

Figure 4 Windows Start Menu Search

2. Right-click on **dcomcnfg.exe** and select the **Run as administrator** option. This utility can also be accessed through the **Control Panel > Administrative Tools > Component Services** link.
3. In the left-hand panel, expand **Component Services** to find the **DCOM config** branch.

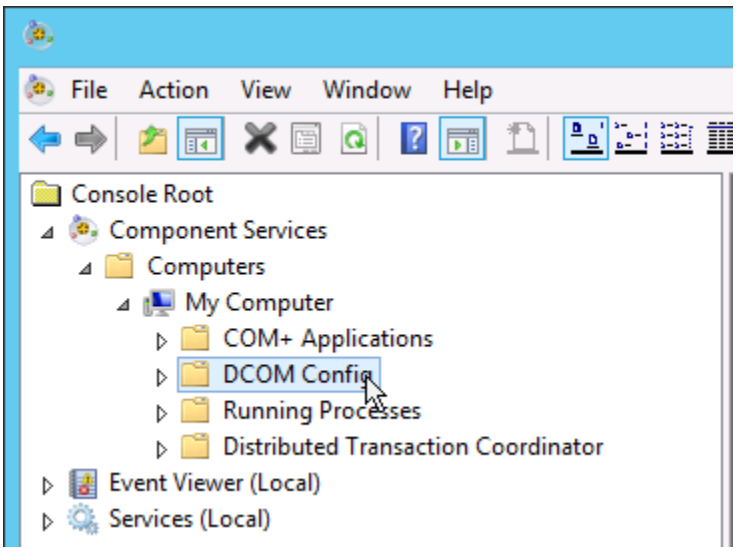

Figure 5 Component Services Directory

4. Selecting DCOM config displays a list of COM server objects in the center panel. Locate the server under investigation, right-click on it and select **Properties** (double-clicking does not function in this utility).

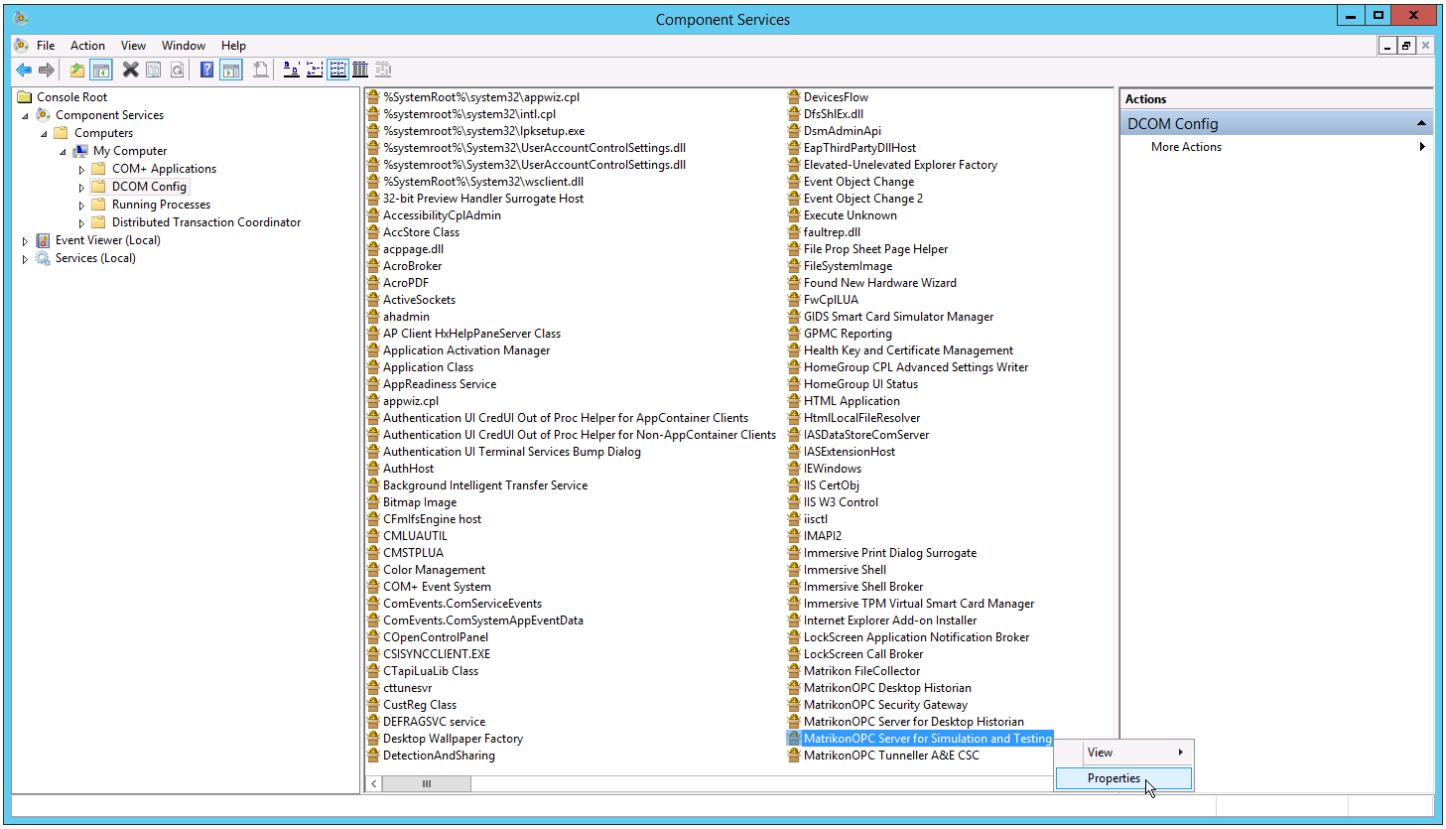

Figure 6 Component Services - DCOMCNFG

5. In the properties dialogue select the **Security** tab. Here you see the settings for three sets of permissions, Launch and **Activation**, **Access**, and **Configuration**. Here you also see **Use Default** or **Customize** for each of these permission sets. For this server, the Customize option is selected. This means that this server has a set of

permissions that differ from the Default settings. It is these Custom settings that will determine which identities are allowed (or denied) access to this server.

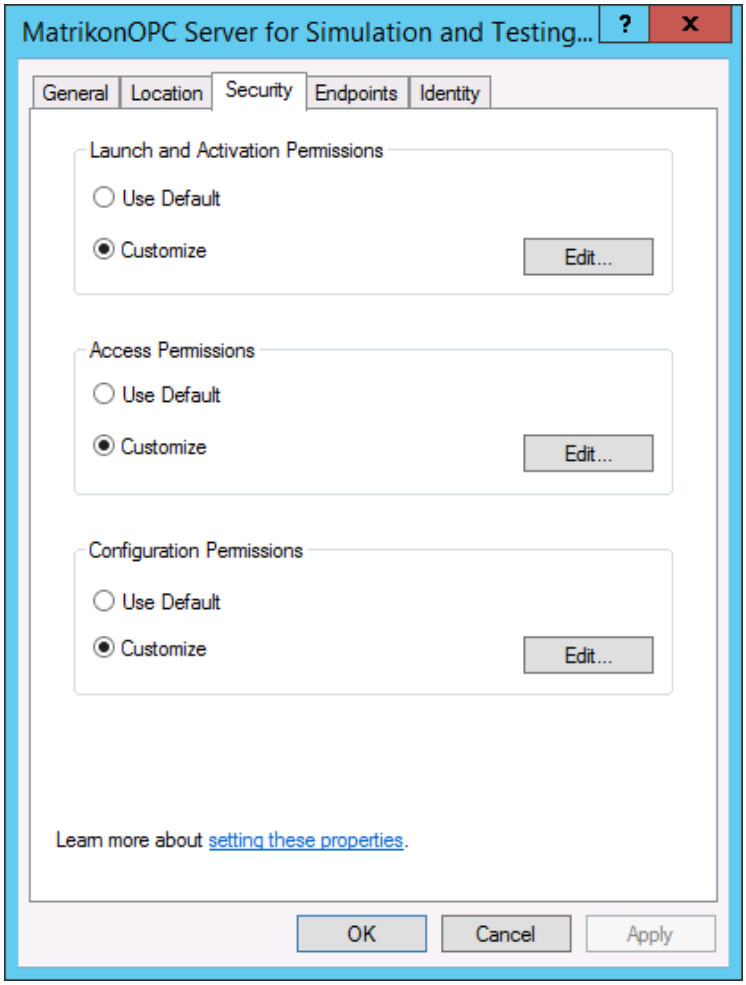

Figure 7 DCOMCNFG Server Properties

6. If the server under investigation has Default selected, proceed to the section on the Default DCOM Permissions.
7. If the server under investigation has Customize selected, proceed to the section on the Custom DCOM Permissions.

Note

There are many mechanisms for modifying the ACL for each COM server. Most of these can have damaging effects in the event of misuse or accidental misconfiguration. They are therefore meant for advanced Windows users. DCOMCNFG is a graphic configuration utility that is the recommended method for making the required changes in these security settings.

Default DCOM Permissions

These permissions will be used by all COM applications on the machine that have not set their own security settings, or had custom settings created for them. Most OPC clients will use these default settings.

In changing these settings, it is important to remember that this will change the DCOM Security settings on most of the COM objects installed on the machine. It is necessary to observe the following precautions.

1. Inform your IT department or System Administrator that you are going to modify these settings. They may have specific precautions that must also be observed.
2. Do not delete, remove, or edit any existing settings unless this document specifically instructs. In this case you must document any changes that have been made so that they can be restored once testing is completed.
3. If the server under investigation has the Customize option selected, do not modify these settings. Proceed to the **Custom Default Permissions** section.

To access/modify the Default DCOM Permissions, perform the following steps:

1. Prior to making any modifications to the existing settings, go through each user's permissions on this server to determine if any permissions are set to **Deny**. If there is a **Deny** setting configured, this may be the cause of your issue. Adding **Allow** permissions for this user, or any group of which this user is a member, will have no effect. Verify that this denied identity is not the identity of any client that is having connection issues with this server before proceeding.
2. In the **Component Services** utility, expand **Component Services** to find the **My Computer** branch. Right-click on **My Computer** and select **Properties**.

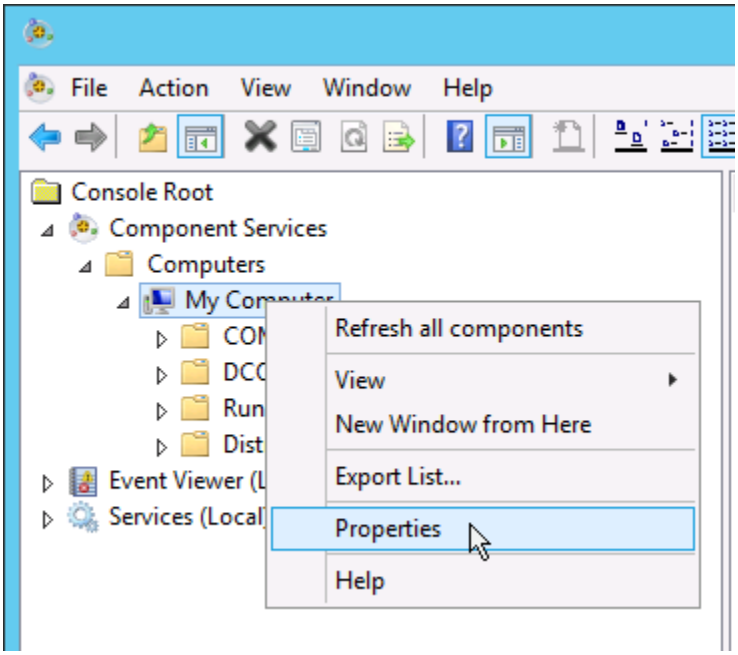

Figure 8 Component Services - My Computer Selected

3. In the **My Computer Properties** window select the **Default Properties** tab. Ensure that
   a. The **Enable Distributed COM on this computer** option is checked

b. The **Default Authentication Le**vel is set to **Connect**.
c. The **Default Impersonation Level** is set to **Identify**.

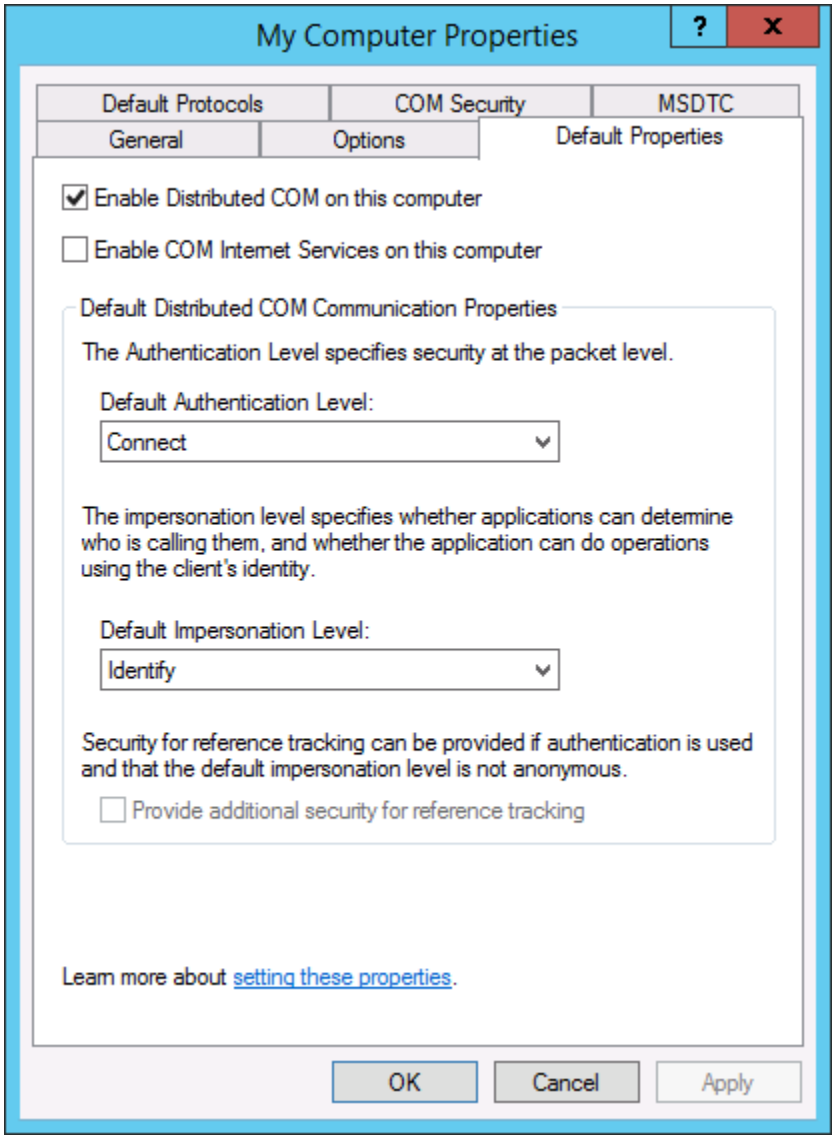

Figure 9 Default DCOM Security Settings 1

4. Select the **COM Security** tab. In each of the **Access Permissions** and the **Launch and Activation Permissions** click on the **Edit Default** button

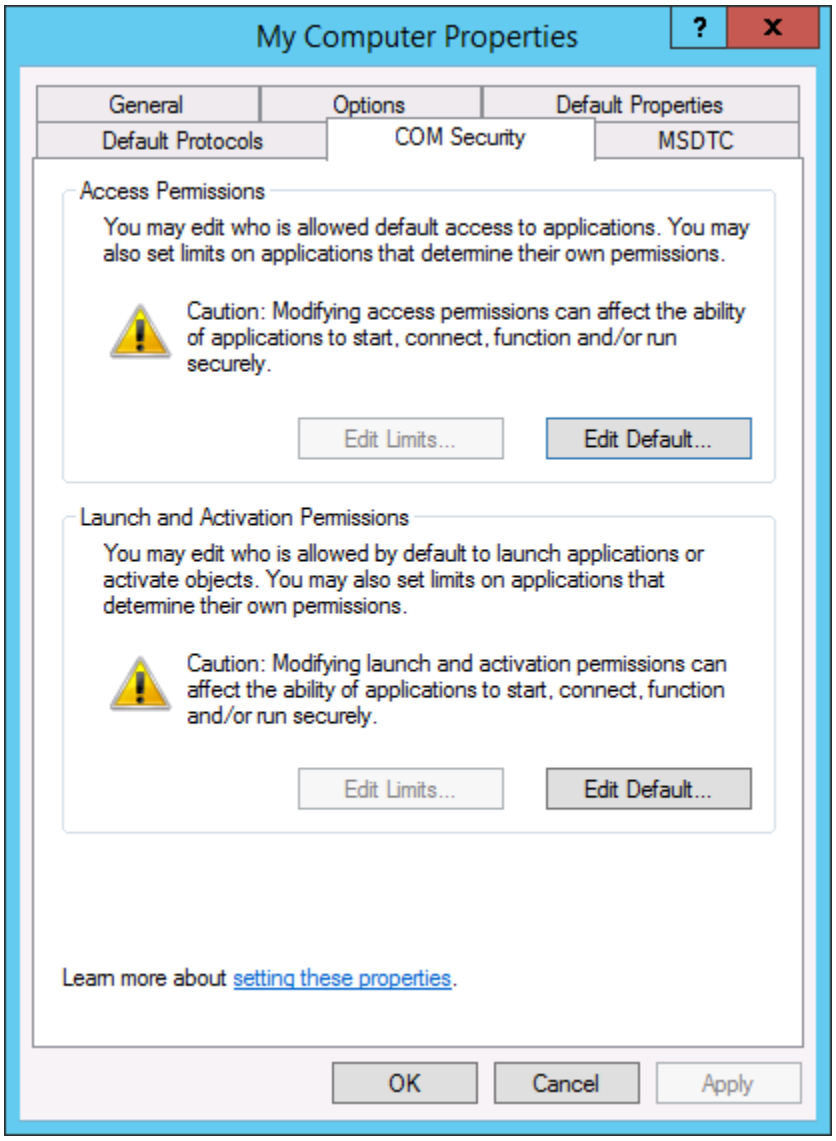

Figure 10 Default DCOM Security Settings 2

5. Add the following users to each permission set and **Allow** both **Local** and **Remote**;
    a. Everyone
    b. Interactive
    c. Network

d. System

6. Click on the **OK** button.

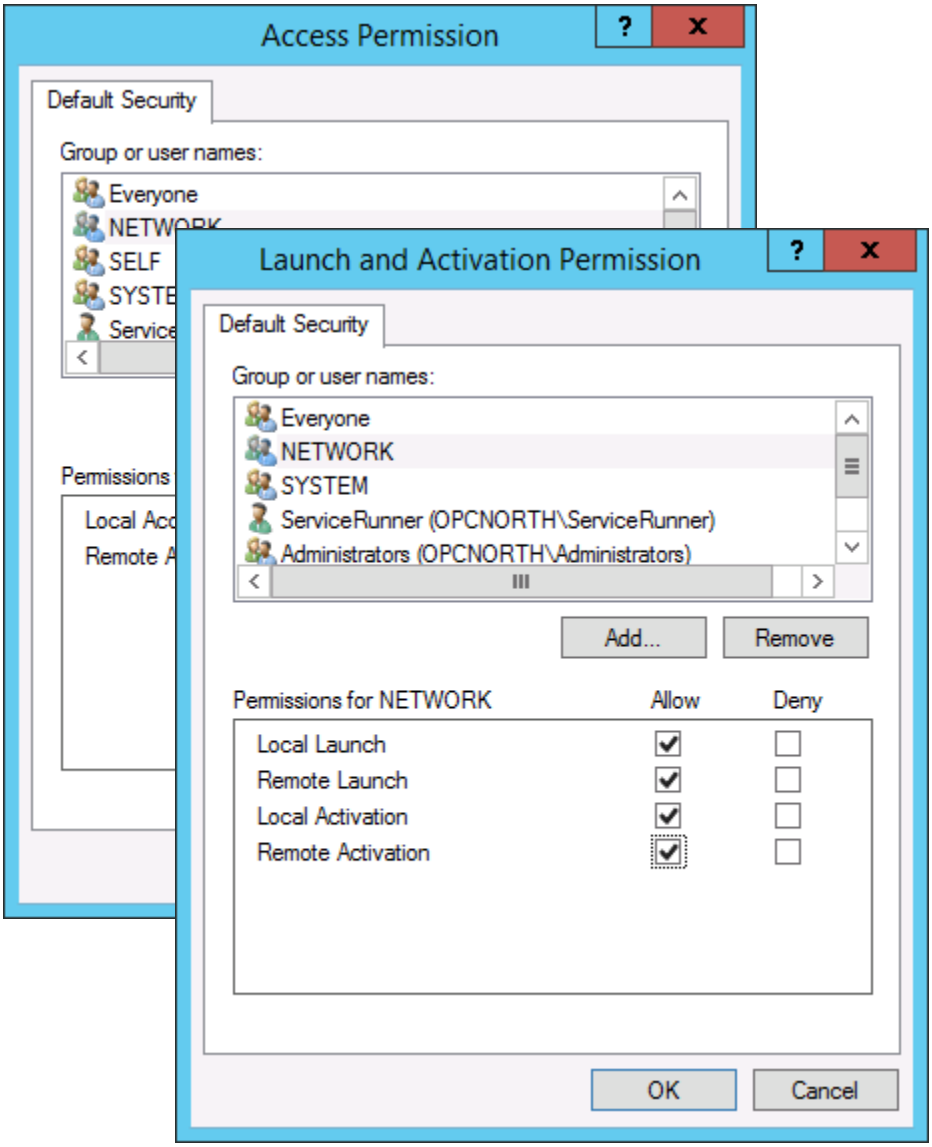

Figure 11 Default DCOM Security Settings 3

7. The **Edit Limits** option in this tab applies machine-wide limits for **Access** and **Launch** permissions. These buttons may be *inactive* (greyed out as shown), in which case no action is required. If they are *active* add the same users as were added to the Default permissions.
8. These settings effectively reduce the default security for the COM objects installed on the machine to minimum levels. If the issue you are investigating is being caused by one of these settings, connection should now be possible. If you are still unable to connect to the OPC server then these settings are not the cause of your issue. Possible next steps include
    a. Create a set of Custom Permissions for this server that specifically include the client identity. Refer to the section on creating/setting **Custom DCOM Permissions** for information on this.
    b. Continue with the other topics in this document, especially **GPO's** and **Network Security**.
9. Once you have completed testing these new settings, return the Default Settings to their previous state prior to implementing a solution or continuing your investigation. This specifically includes
    a. Removing any identities that were added to the existing configuration

b. Returning any settings that were modified to their original values

10. For more information on the Default DCOM settings and configuring system-wide security, refer to Microsoft Docs.

Custom DCOM Permissions

Some COM objects require Security Settings that differ from those of the default configuration. Removing the reliance on the default settings, requiring that access for specific users or groups be allowed (or denied), allowing the server to use a specific identity, these are all reasons why custom permissions are used for certain COM applications.

It should be noted that for Matrikon OPC servers, the installation program registers the server as a service and creates a set of custom permissions identical to the settings implemented in this document. Both of these items are selectable at installation and can be modified once installation is complete.

To access/modify the Custom DCOM Permissions, perform the following steps:

1. In the left-hand panel, expand **Component Services** to find the **DCOM config** branch.

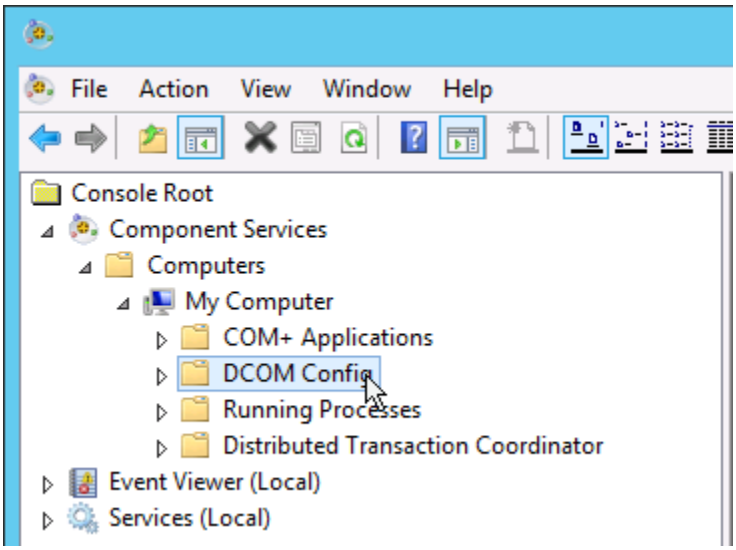

Figure 12 Component Services - DCOM Config Selected

2. Selecting DCOM config displays a list of COM server objects in the center panel. Locate the server under investigation, right-click on it and select **Properties** (double-clicking does not function in this utility).

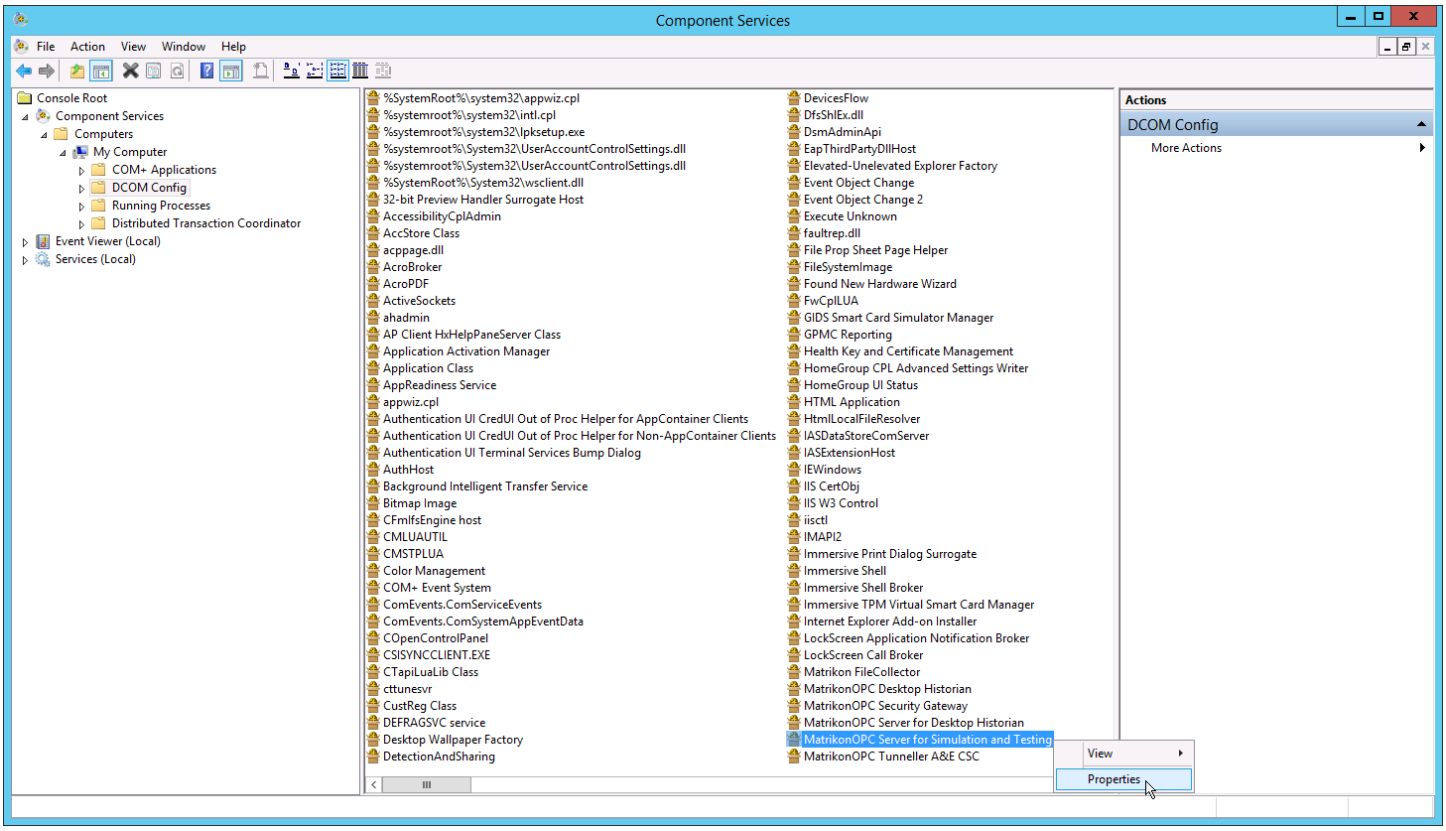

Figure 13 DCOMCNFG Server Selection

3. To enable security for an application, you must set an **Authentication Level** other than **None**. In the server **Properties** window, select the **General** tab (it should appear by default) and ensure that the **Authentication Level** is set to **Connect**. Click on **Apply**.

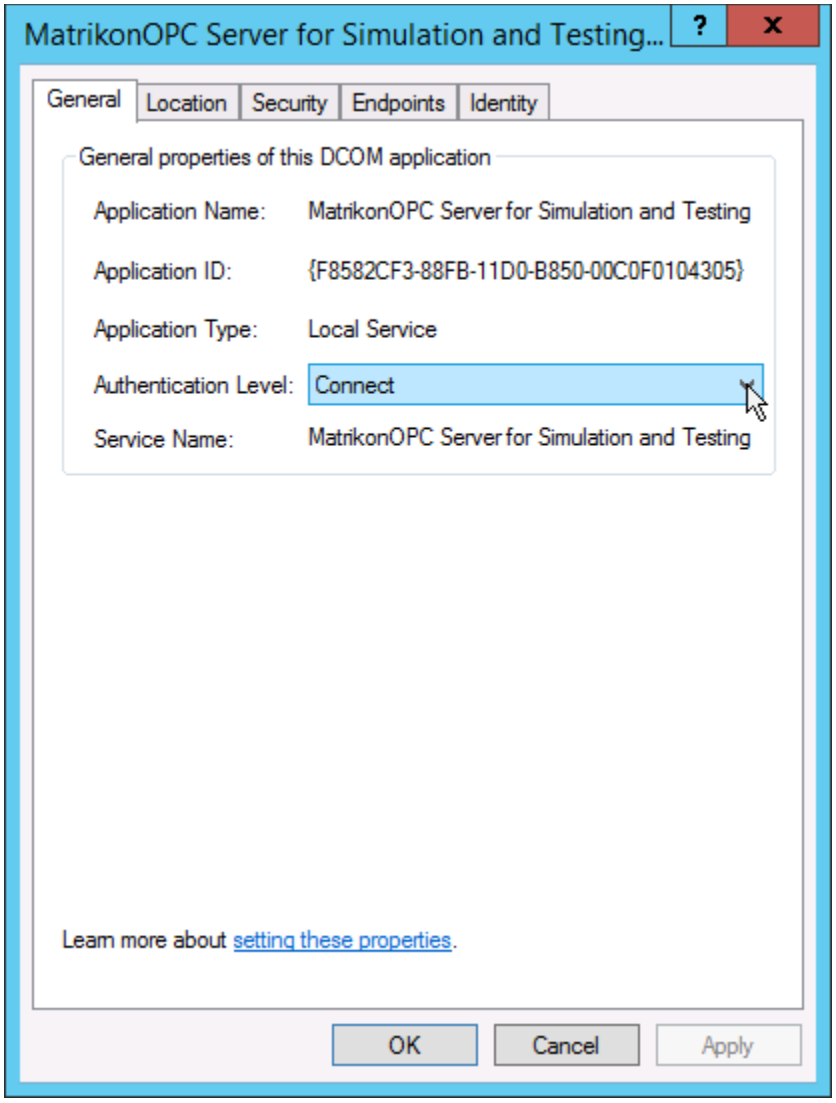

Figure 14 Custom DCOM Security Settings - General

4. In the Location tab, ensure that the Run application on this computer option is selected. Click on **Apply** as required.

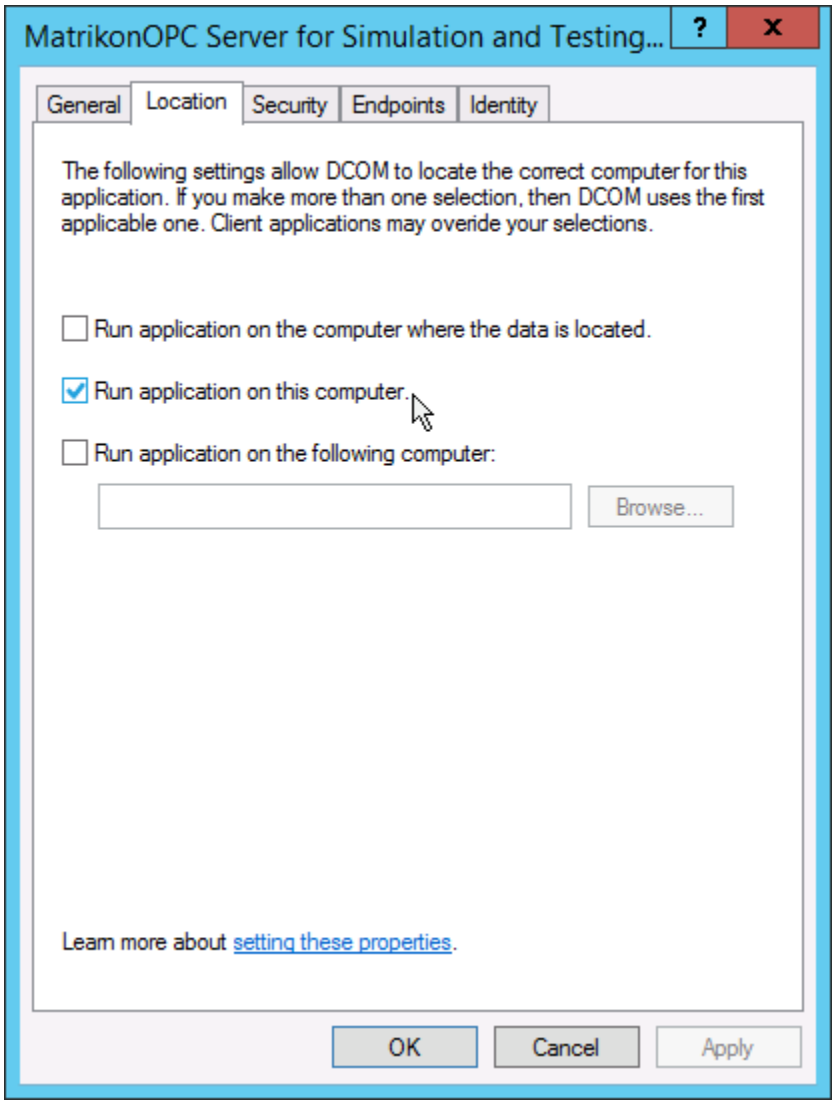

Figure 15 Custom DCOM Security Settings - Location

5. The **Security** tab contains the permissions for the server object in terms of which identities are allowed (or denied) access to the server. Prior to making any changes to these settings, the following precautions/steps should be taken.
    a. Go through each user's permissions on this server to determine if any permissions are set to **Deny**. If there is a **Deny** setting configured, this may be the cause of your issue. Adding **Allow** permissions for this user, or any group of which this user is a member, will have no effect. Verify that this denied identity is not the identity of any client that is having connection issues with this server before proceeding.
    b. Do not delete, remove, or edit any existing settings unless this document specifically instructs. In this case you must document any changes that have been made so that they can be restored once testing is completed.

Note

In this utility we are concerned only with the **Launch and Activation**, and **Access** permissions. Although it is possible to restrict who can modify these settings through the use of the **Configuration Permissions**, this can be risky unless you have specific reasons for restricting this access. Unless this is the case, the **Configuration Permissions** are best left unaltered.

6. In the **Security** tab, ensure that the **Customize** radio button is selected for each of the **Launch and Activation Permissions** and the **Access Permissions**.

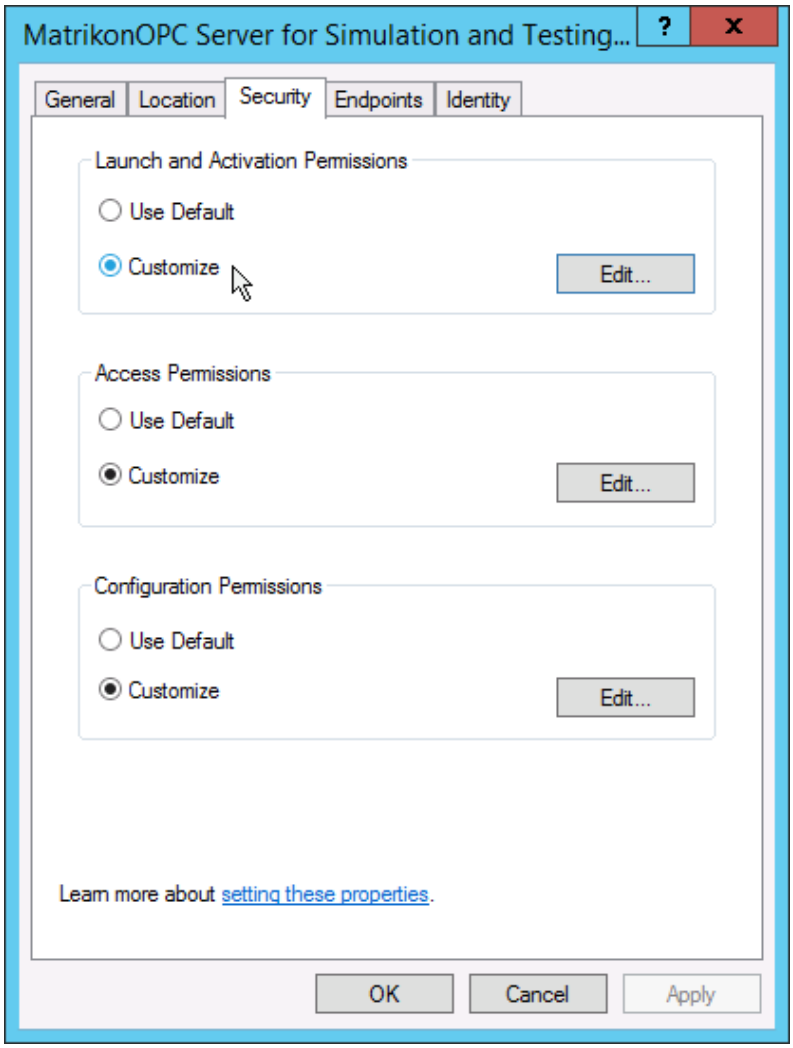

Figure 16 Custom DCOM Security Settings – Security

7. In turn, click on the Edit button for each of these permission sets and add the following identities with Allow set for all Local and Remote settings (refer to Client Identities for more information on these identities);

    a. Everyone
    b. Interactive
    c. Network
    d. System

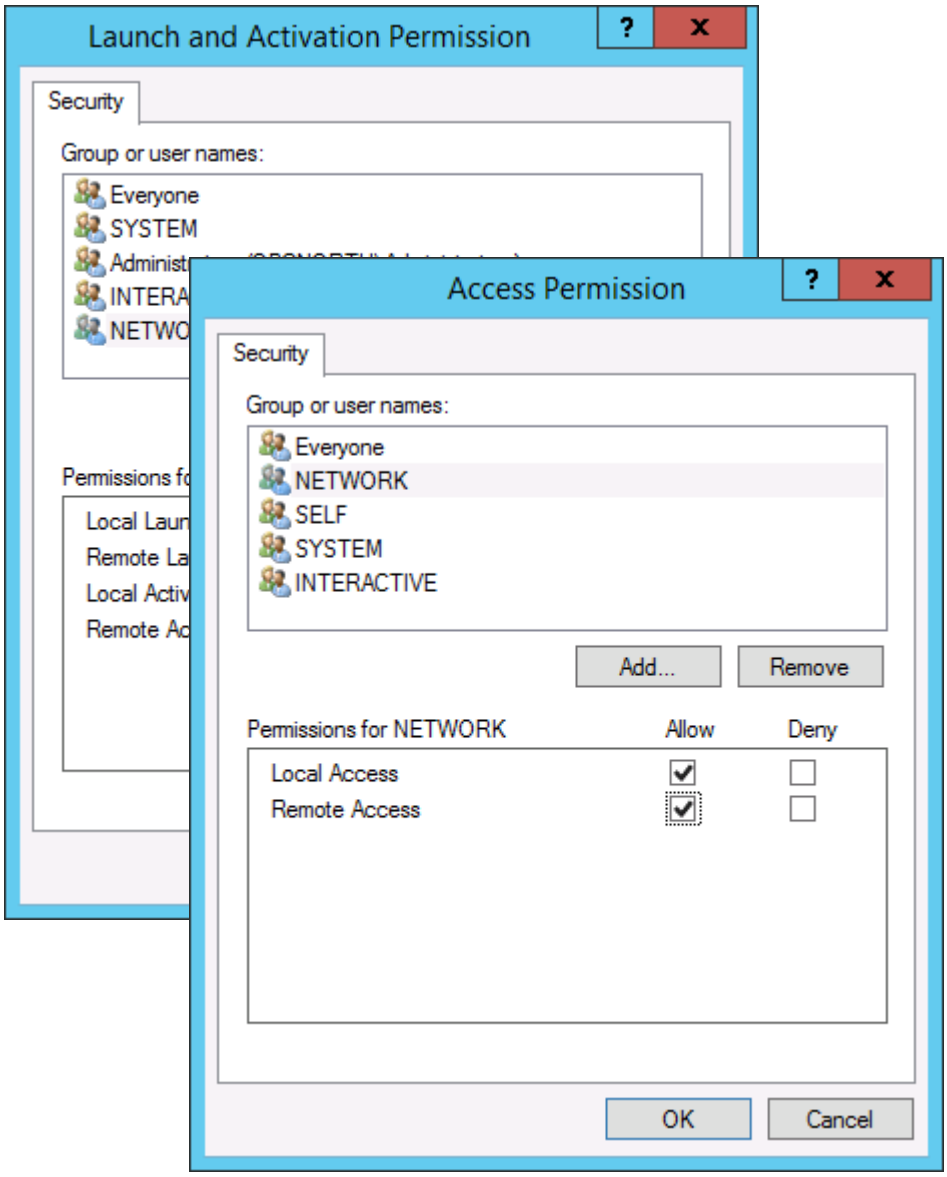

Figure 17 Custom DCOM Security Settings - Server Permissions

8. Click on **OK**, then **Apply**.

9. The Endpoints tab allows configuration of the network protocols and endpoints for this server that are available to the client. For this setting, the default system protocols will normally work as in almost all cases the default will be TCP/IP. If another protocol or set of endpoints is to be used, it can be selected here.

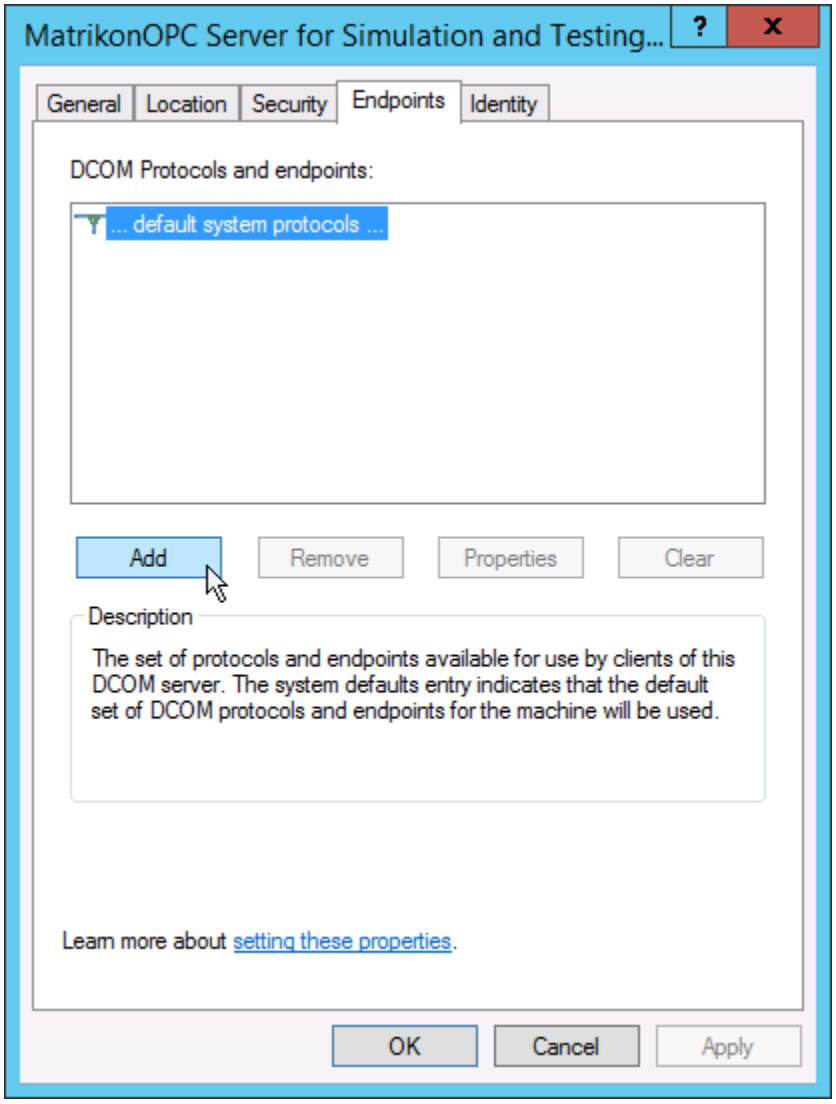

Figure 18 Custom DCOM Security Settings - Endpoints

10. Click on the **Add** button and select the required protocol from the drop-down list. Ensure that the required endpoints are also selected from the available options.

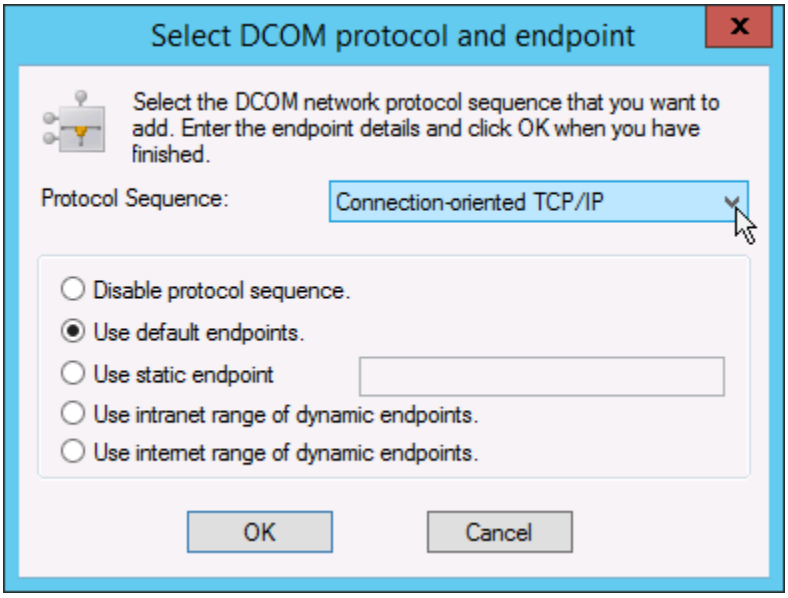

Figure 19 Custom DCOM Security Settings - Endpoint Selection

11. The identity of the OPC server is the account that is used to run the server. This can be configured on the Identity tab. When selecting an identity, it is important to understand whether the server runs as an interactive application or if it is registered as a service. If running as an interactive application, the **system account** will not be available. Of the remaining choices there are serious issues with both the **Interactive User** and the **Launching User** that limit their suitability for OPC applications. Selecting **This user** allows you to use an appropriate account as the identity of the server. When the server is registered as a service, only the system account and This user options will be available. Although the system account is appropriate for almost all situations, if the server requires access to a resource whose own permissions are highly restricted, **This user** once again allows selection of an appropriate account.

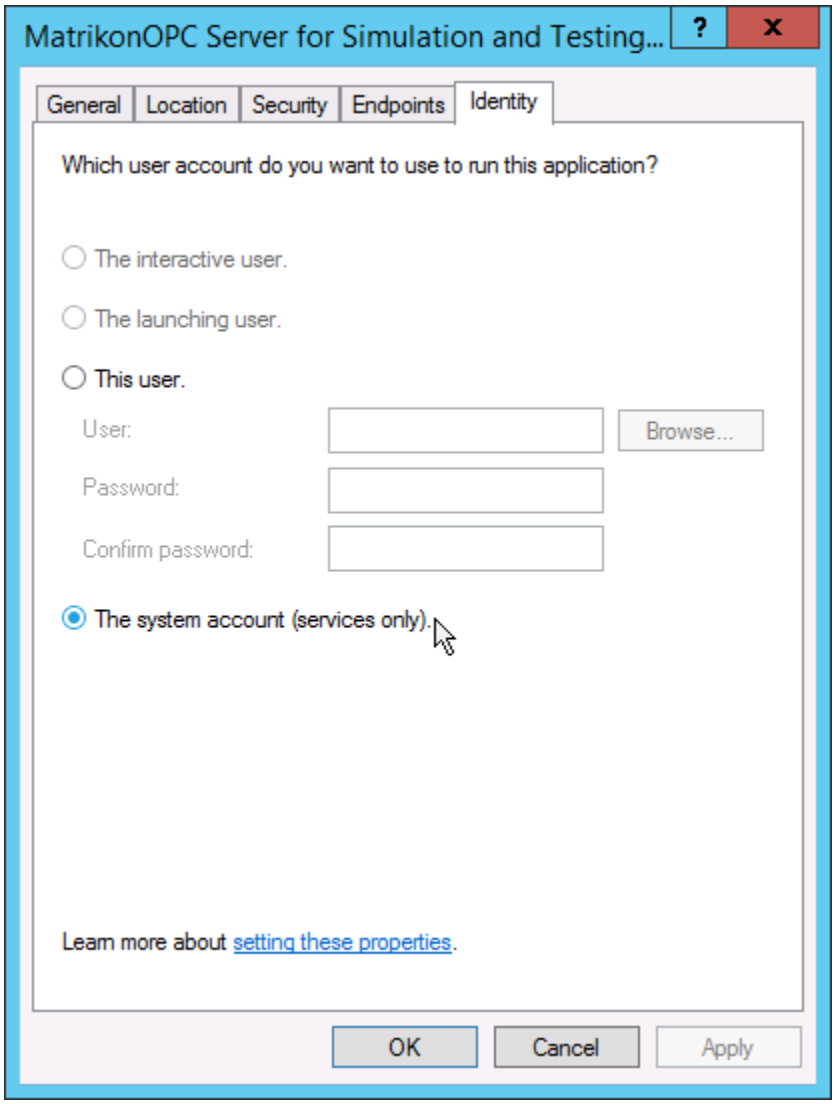

Figure 20 Custom DCOM Security Settings - Identity

12. This setting depends upon the specific requirements of your system. Refer to Server Identities for further information. Select the appropriate identity and then click on the **OK** button.
13. At this point you have completed the configuration of the Custom DCOM Permissions. The settings implemented in this document have reduced the security on the server under investigation to minimum levels, effectively removing security from the server object. Test these settings by attempting to connect your client to your server. If you are now able to create this connection, your initial security settings were at fault and need to be reconfigured. If still unable to connect, these settings are not the issue and additional troubleshooting is required.

14. Regardless of the outcome of the testing, before moving on to the next step in the process it is necessary to return the DCOM Permissions to their initial state.
    a. Remove any identities that were added to the **Launch and Activation Permissions** and the **Access Permissions**
    b. Return any settings that were modified to their initial setting
15. Although we are using these settings to troubleshoot connection issues between OPC applications, it is important to note that these settings are an integral component of the Windows Security Framework. While it may be convenient to use these settings to reduce security on the OPC servers, thereby reducing connectivity issues, this does create gaps in your system security and can provide an experienced unauthorized user with an attack surface to exploit. It is recommended that you consult with your IT Department / System Administrator and software vendor to apply the appropriate level of security. Additional information on setting up the Custom DCOM permissions can be found at Microsoft Docs.

Local Security Policy

The Local Security Policy settings are a set of rules implemented by administrators to protect the resources on the computer. They are, in effect, a set of machine-wide Access Control Lists (ACLs) that restrict access to all COM objects on the machine. Although these are Local settings, they will still be in effect in a Domain environment as they form a subset of the overall security configurations as part of a Group Policy Object (GPO). Where there is a conflict between the Local Policy and the GPO on a Domain, the Domain settings will take precedence. If your system is part of a Domain and you are still experiencing connection issues after configuration of the Local Security Policy settings, refer to the section on **Group Policy Objects**.

The default for these settings is either Blank or Not Defined. If this is the case, there is a default set of permissions in the registry that Windows will use. Depending on the Windows version and network configuration of your system, these default settings may cause communication to some COM applications to fail.

If these settings are already configured, they may be the cause of the connection issue. In this case, prior to modifying these settings, examine these settings to determine if there is an identity that is **Denied** access to the machine. Do not remove or edit existing settings unless explicitly instructed to do so.

To access/modify the Local Security Policy Options, perform the following steps:

1. There are a number of ways to access the Local Security Policy settings, depending on the version of Windows. This is most commonly available through the Administrative Tools in the Control Panel. However, to see the Local Security Policies in their wider context, click the **Start** button and type *gpedit.msc* in the search window. Regardless of the path chosen, access to this tool requires an account with administrator privileges.
2. In the left panel of the **Policy Editor** window, expand the **Security Settings** to find the **Security Options**.

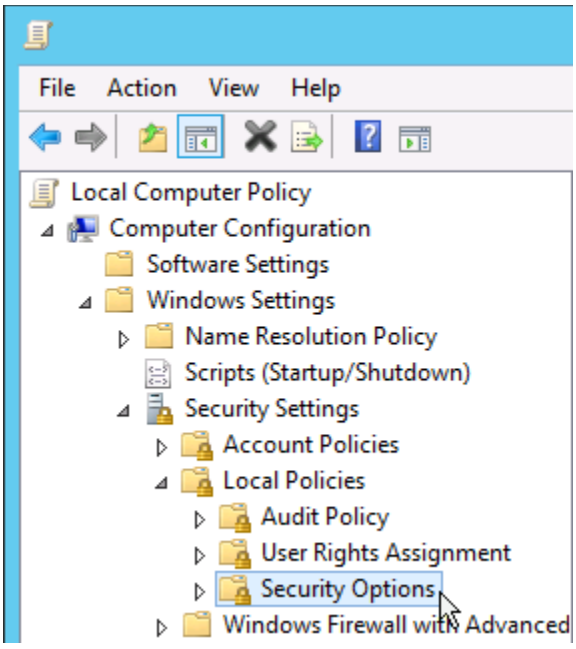

Figure 21 Security Policy Editor

3. The Security Options will be listed in the center panel. To configure the machine-wide settings for COM, you will need to modify two items;
    a. DCOM: Machine Access Restrictions in Security Descriptor Definition Language (SDDL) syntax
    b. DCOM: Machine Launch Restrictions in Security Descriptor Definition Language (SDDL) syntax
4. Double-click, or right-click and select Properties on each in turn to open the Properties dialogue.

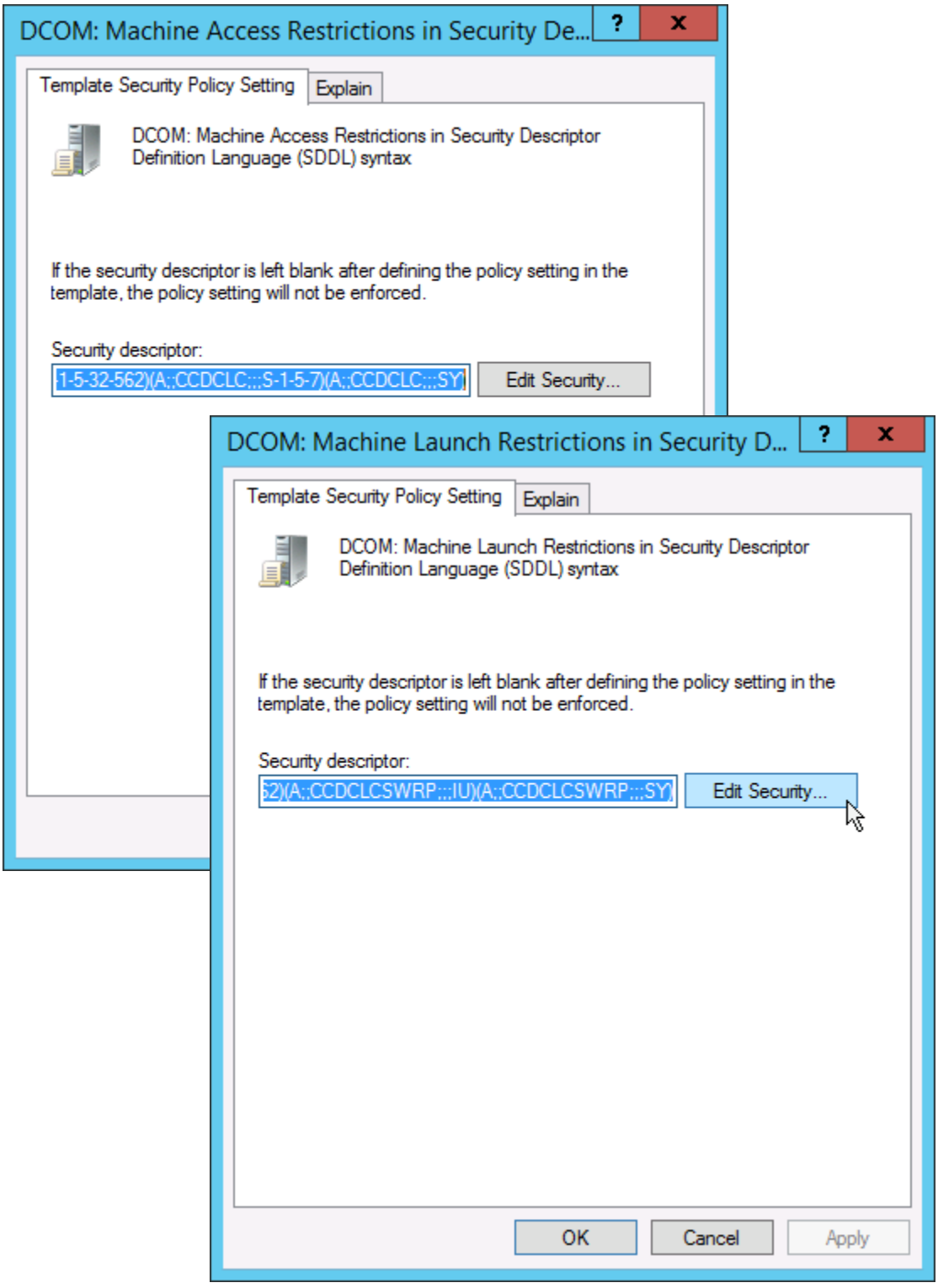

Figure 22 Local DCOM Security Policies

5. Click on the **Edit Security** button.

6. Ensure that the following Users/Groups are added and that all have Local and Remote access allowed (refer to Client Identities for more information on these identities)

    a. Everyone
    b. Interactive
    c. Network
    d. System

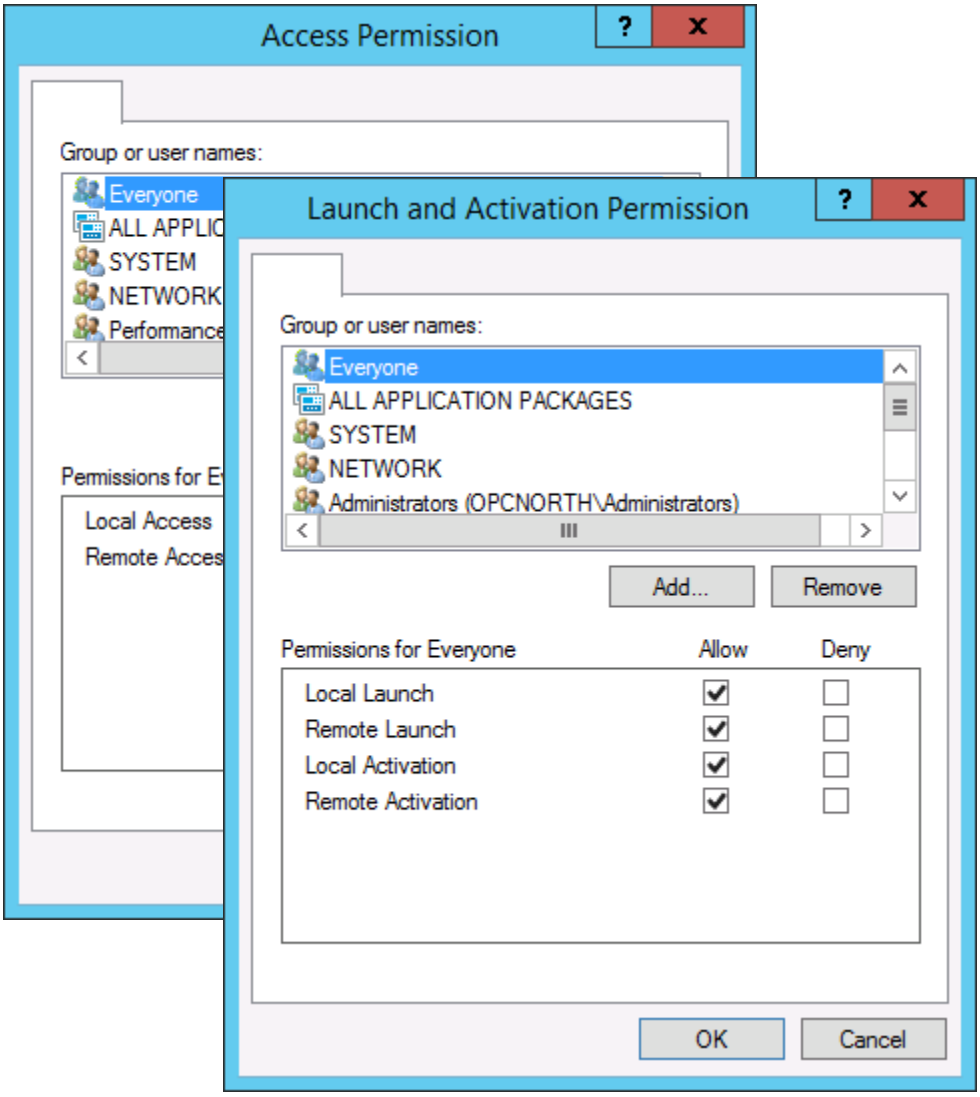

Figure 23 Local DCOM Security Policy Settings

7. Click the **OK** button to return to the **Security Policy** window and select **Network Access: Let Everyone permissions apply to anonymous user**. Double-click the setting to open the dialogue and **Enable** this setting. Click on the **OK** button.

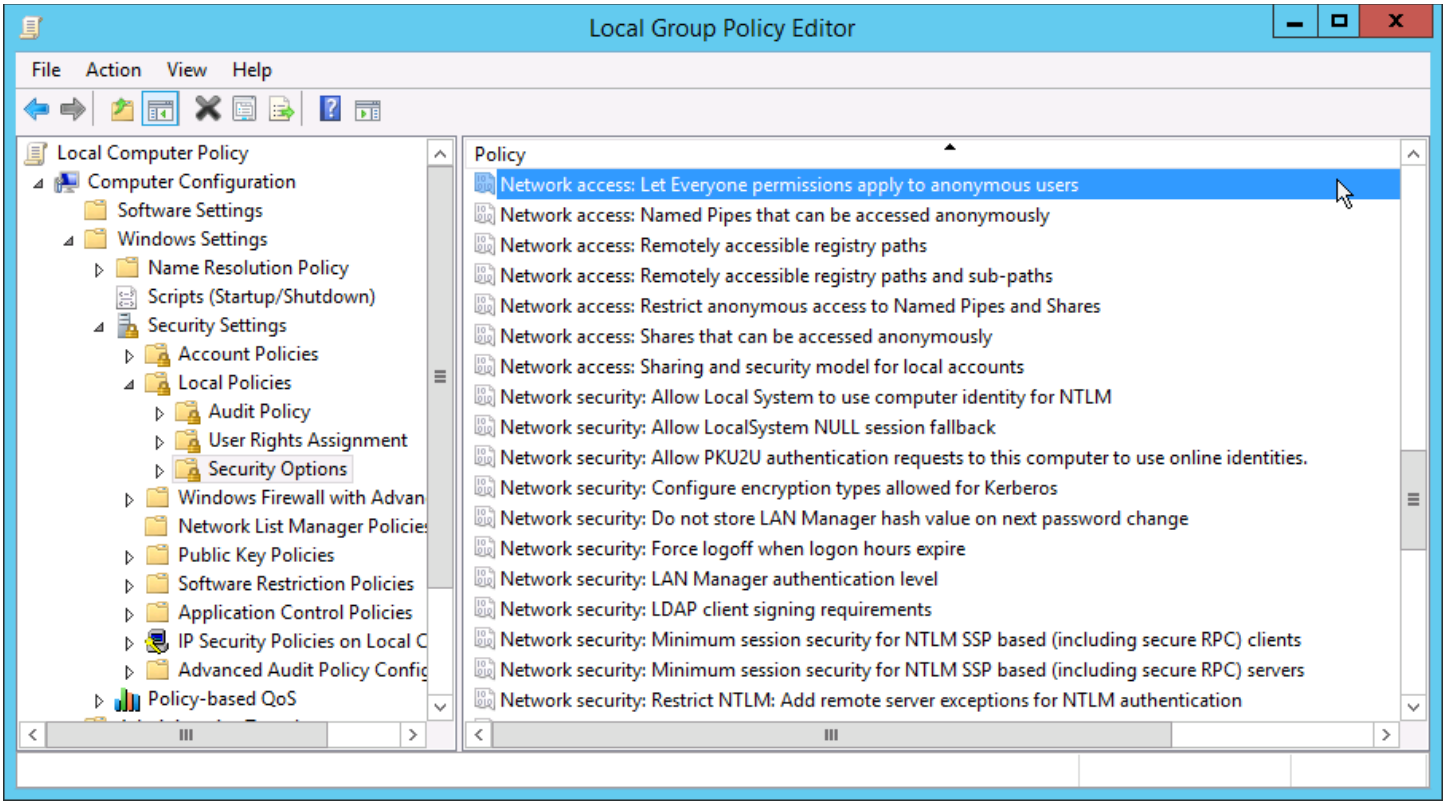

Figure 24 Local Network Access Security Policy 1

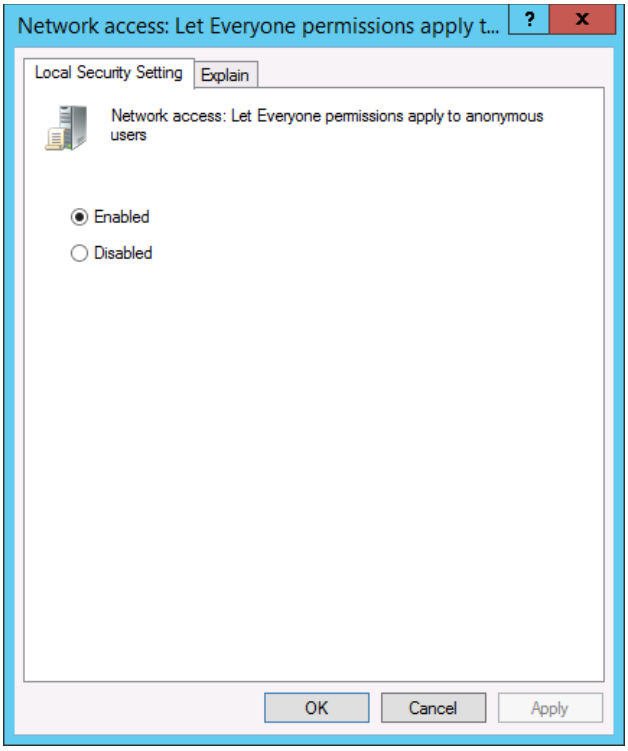

Figure 25 Local Network Access Security Policy Settings 1

8. Select **Network Access: Sharing and security model for local accounts** from the **Security Options**. Double-click this item to open the properties dialogue and select **Classic – local users authenticate as themselves** option from the drop-down list. Click on the **OK** button.

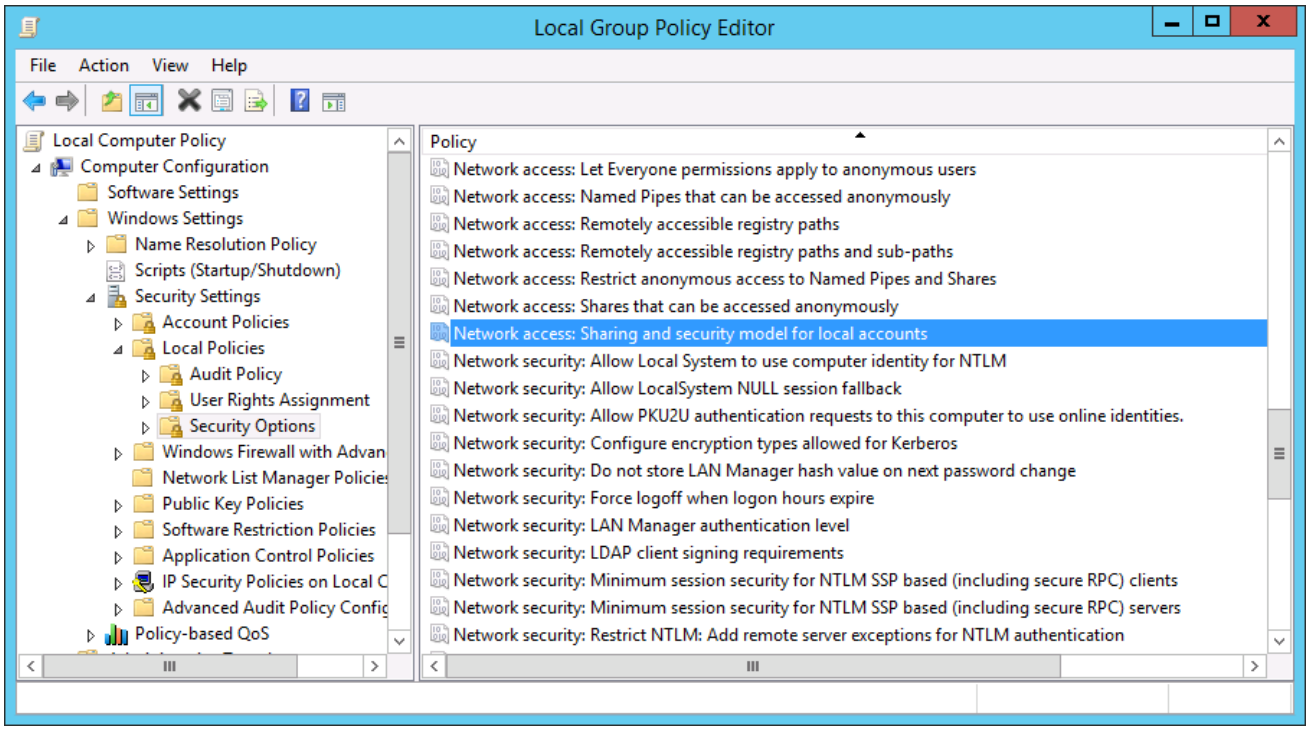

Figure 26 Local Network Access Security Policy 2

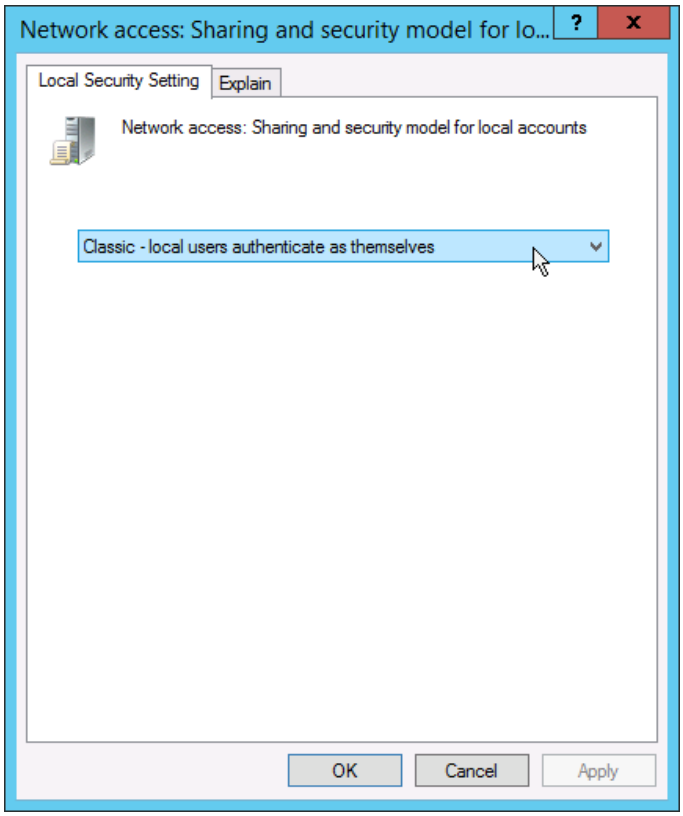

Figure 27 Local Network Access Security Policy Settings 2

9. From the User Rights Assignment policies, select Access this computer from the network. Double-click to open the Properties dialogue. Ensure that the Everyone and Users groups have been added to this policy.

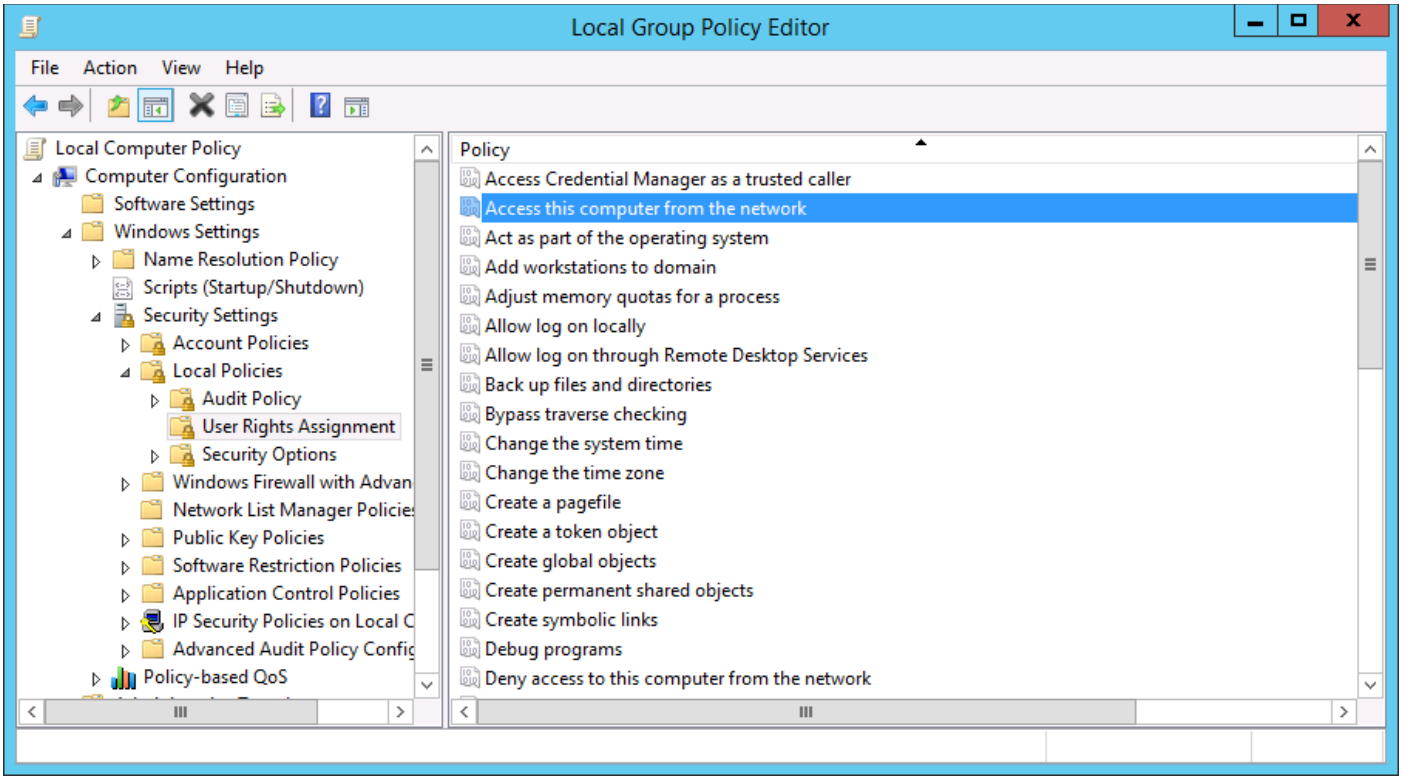

Figure 28 Local Security Policy - User Rights Assignment

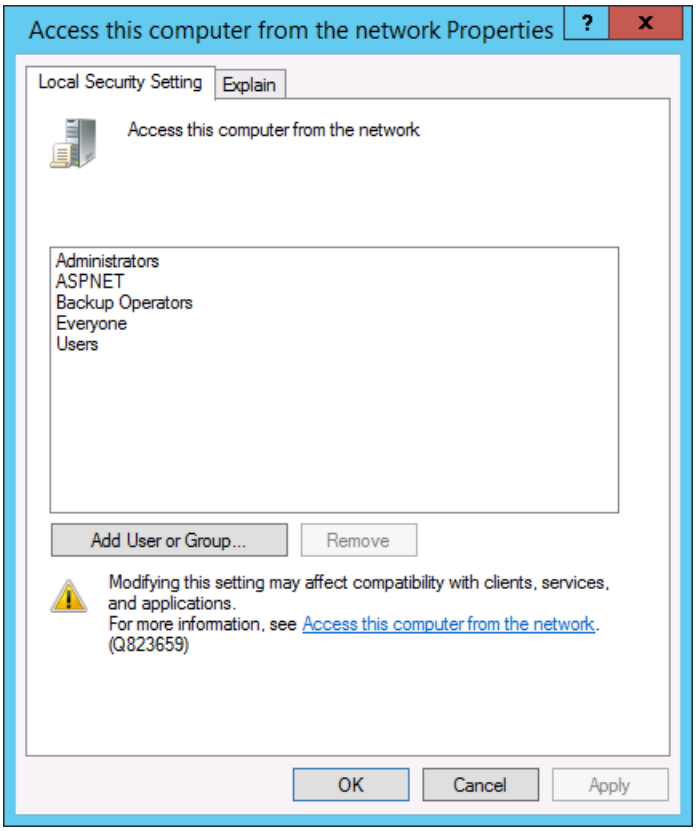

Figure 29 User Rights Assignment

10. As with the settings in previous sections of this document, the security on your system has been set to minimum levels. Successful connection indicates that one or more of these policies (or the default settings if undefined) were responsible for the connection issues you were experiencing. Should this be the case, consult with your IT Department and vendor support team to implement the proper security configuration. If you are still unable to connect client to server, additional troubleshooting is required.

Windows Firewall

Since Windows XP SP2 / Server 2003 SP1, the Windows Firewall is turned on by default. This software firewall will prevent DCOM communication by blocking the remote calls that DCOM requires for such functions as DNS name resolution, function calls and callbacks, to name a few. Exceptions can be made in the firewall, either by application or by port number. This process is described elsewhere, for example in the Windows Help files. The issue is that DCOM requires such a wide range of ports be opened that there are serious gaps left in the security of any system thus configured. The Windows Firewall with Advanced Security that is available in newer versions of Windows allows for more detailed configuration of the firewall including rules for both inbound and outbound traffic. However, this can entail some very complex configuration. It is also quite common to find that access to this configuration tool is controlled by system administrators through a GPO. For troubleshooting purposes, it is more effective to simply turn the firewall off. It may be necessary to contact your IT Department / System Administrator to do this.

To turn off the Windows Firewall, perform the following steps:

1. Open the **Control Panel** and double-click the **Windows Firewall** icon.

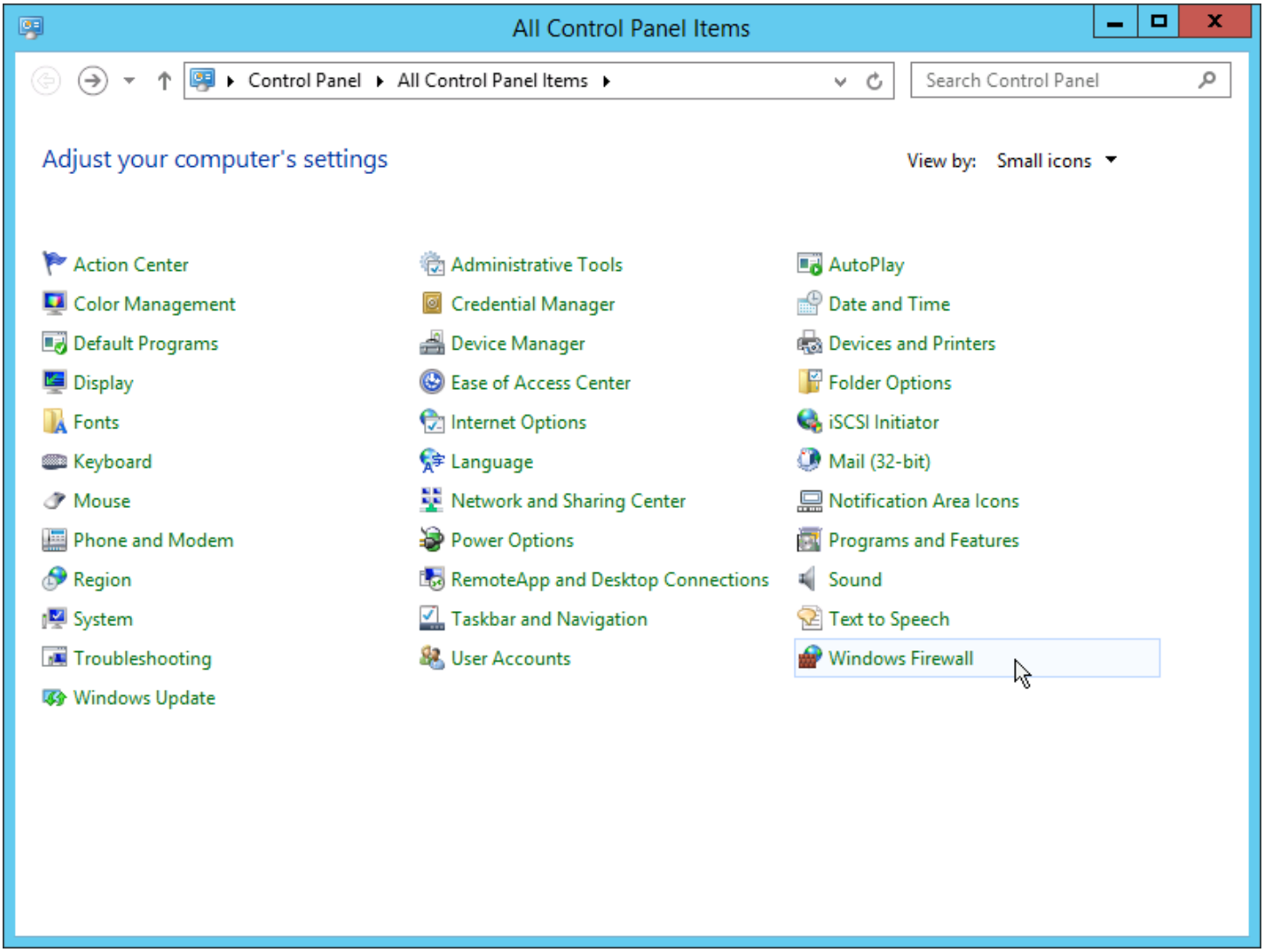

Figure 30 Windows Control Panel

2. Select Turn Windows Firewall on or off on the left side of the window.

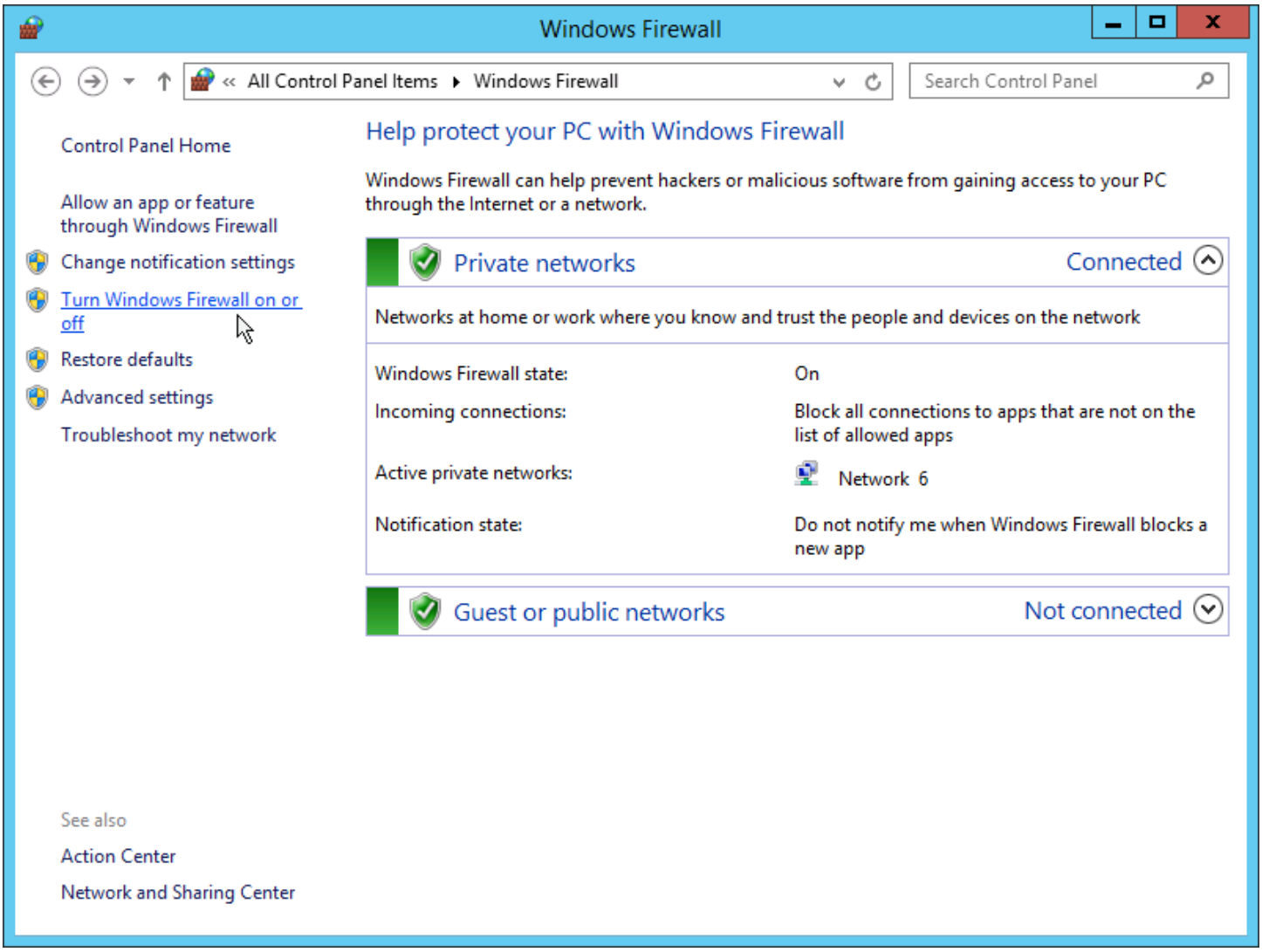

Figure 31 Windows Firewall Properties

3. For the appropriate network type (Domain network settings not pictured here), select the **Turn off Windows Firewall** radio button.

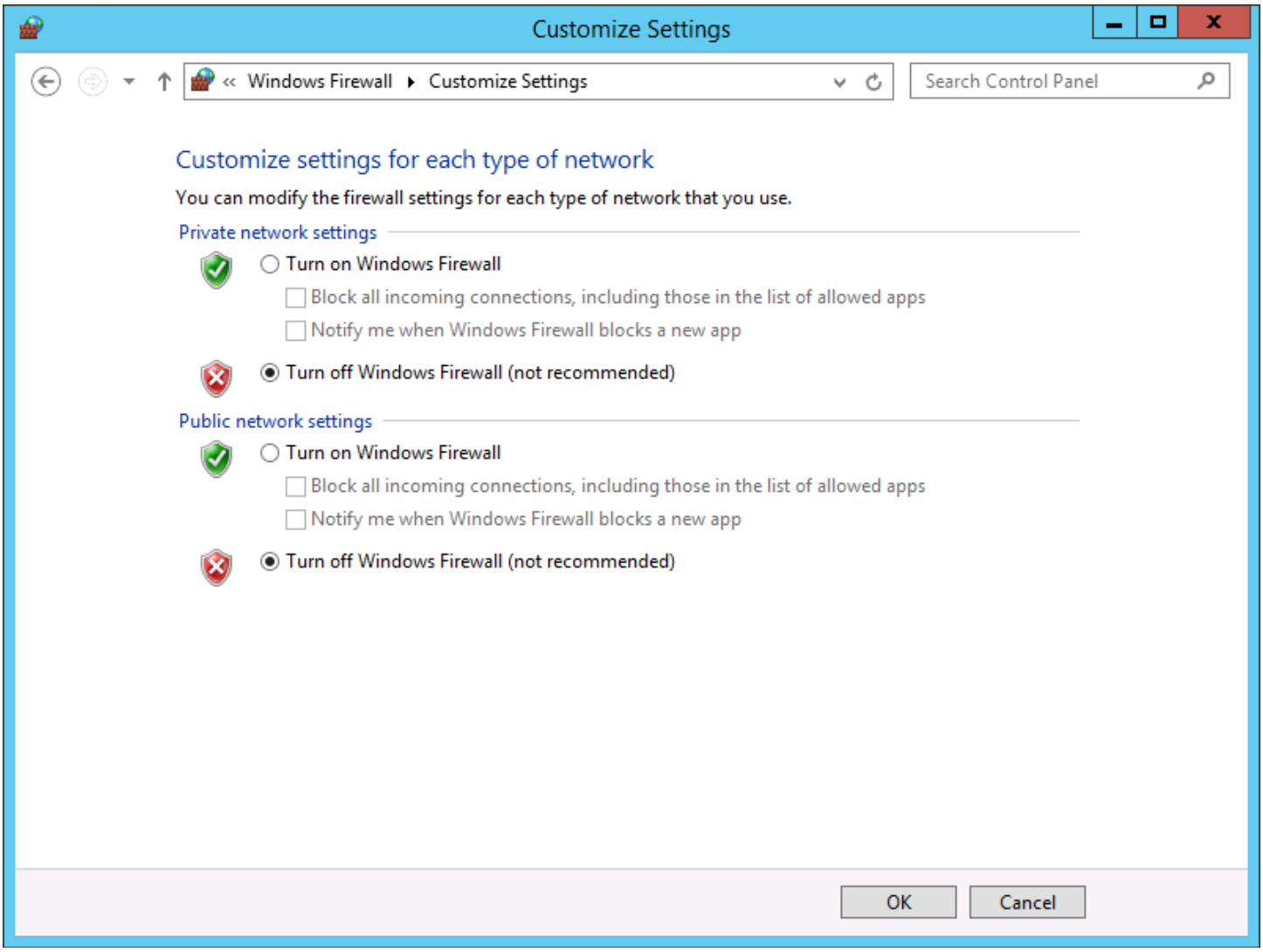

Figure 32 Windows Firewall Settings

4. If this restores/permits connectivity, you will need to configure the Firewall to allow the OPC components access or make arrangements as necessary to turn the Firewall off on a permanent basis. Alternatively, Matrikon's **OPC UA Tunneller** can be used to simplify configuration (refer to the **Network Components** section of this document).

Security Configuration and Network Components

Security configuration on the OPC host platforms is the most common cause of connectivity issues. Approximately 90 – 95% of all connectivity issues are caused by one or more misconfigured security settings. When troubleshooting these issues, it is important to remember that anything that affects system security can potentially have an effect on OPC communication. Application of security or product updates for the operating system or any application can be considered suspect if COM applications fail following system maintenance.

Network components and architecture can likewise erect barriers between OPC applications. Network segmentation, multiple domains, hardware firewalls, NAT, and port-forwarding configurations can all prevent OPC clients from connecting and communicating with OPC servers.

Command line utilities such as Ping, TraceRT, Telnet, Netstat, and Netsh, as well as applications such as Wireshark can be used to investigate whether the network is at issue when experiencing OPC communication failure. When using these utilities / applications, coordinate with your IT Department / System Administrator to ensure such tools fall within the acceptable use guidelines for you system.

Limitations

DCOM was developed to function in a specific environment where the following conditions applied:

1. All machines and users belonged to the same domain.
2. There were no firewalls enabled on any machines or network devices.
3. All communication media were highly reliable.
4. There were no bandwidth restrictions.

All of these were typical of a LAN setup in an average office environment; however, this bears little resemblance to the process control networks of today. Multiple domains, security-oriented IT policies, geographically dispersed data sources, and a multitude of other factors all make OPC communication based on DCOM extremely complicated to configure while maintaining security.

Matrikon's OPC UA Tunneller provides successful OPC communications across firewalls or domain/workgroup barriers. Using a single TCP port to the remote computer, it is much easier to configure routers and firewalls without sacrificing security. It is one of our most popular products because of its ease of use, automatic reconnection system, and time savings in implementation that it offers.

Contact your Account Manager or visit our website at www.matrikonopc.com for more information on this and other MatrikonOPC solutions.