

Creating Secure OPC Architectures

Adriel Michaud

(Author’s note: This is a technical guide on how to secure typical OPC implementations. In the interest of creating a concise, yet complete guide; we will not be going over any OPC, DCOM, or Windows security basics. For DCOM and OPC basics, see <http://MatrikonOPC.com/training/index.aspx>.)

OPC represents an easy to use, ubiquitous, reliable method of communication. As a result of it being based on DCOM; it is well known by security professionals, but unfortunately is also known by virus writers and hackers. Layers of security mean that should one part of the system be compromised, the rest will remain secure. Typical layers used include: physical business layer security, physical process layer security, exposure to the internet, business to process layer traversal, OPC architecture security, DCOM configuration, login security, firewalls, etc.

The amount of security required will be different for different implementations. A system at the bottom of a secure mine that will never be connected to the rest of the world will not have to be secured with remote attackers in mind. Physical plant security is a big part of system security, but we can’t assume that it’s enough. Relying on security through a single point, or relying on security through obscurity, is not enough.

For our purposes, we’ll be concentrating on OPC DA, HDA, and A&E. Applications using other specifications are few and far between. OPC

systems include OPC Servers and OPC Clients. OPC Servers represent the largest risk out of all the components because they typically have direct access to devices and data. While it is fairly trivial to introduce an OPC Client to an unsecured point in a system, it is extremely difficult to install and configure an OPC Server to interface with plant resources. This is because OPC Servers typically have privileged or exclusive connections to devices and applications. Therefore, we will concentrate mostly on securing OPC Servers from unauthorized OPC Client access and restricting authorized OPC Clients to only the functions necessary.

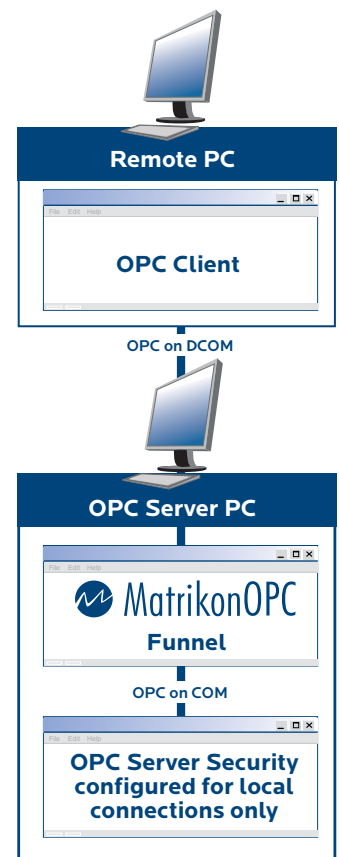


Fig1. Shielding sensitive OPC Servers using MatrikonOPC Funnel

SECURITY TECHNIQUES USING OFF-THE-SHELF PRODUCTS

Before talking about COM/DCOM, it is important to show how a few key products can greatly improve the security of your system, and your peace of mind.



Fig 2. "Push" Architecture using MatrikonOPC Data Manager and OPC Server for Caching

SHIELDING SENSITIVE OPC SERVERS

When possible and practical, sensitive OPC Servers should be shielded from direct OPC Client access. This is because most OPC Servers in use have no limiting mechanisms. They just give full access to everything and anyone. One mechanism is limiting traffic to one way only. It can be done directly on hardware, such as on some PLCs, and can also be



Fig 3. Tunnelled OPC using MatrikonOPC Tunneller

done on some OPC Servers and products. Other desirable parameters would be the ability to limit OPC Clients to certain tags only, and be able to set which are read/write, and which are read-only. OPC Funnel is one such product. It can be used to shield an OPC Server from direct OPC Client access. The target OPC Server has its security locked to COM connections only. Funnel makes the connection to the OPC Server, and is exposed as necessary to the outside. Any OPC Client can be added and view the data, but they can only view/change the data we have allowed.

PUSH ARCHITECTURES

Another way of securing OPC architectures is through a "push architecture." Recall that we are concerned with OPC Clients being added to our system and accessing data. This is because OPC Clients generally control what the OPC Server does, and sometimes that's more power than we're comfortable giving them. In these instances, we can "push" the data to a cache local to the OPC Client and completely lock the OPC Server from outside DCOM connections. This is possible because client and server security settings are separate. With this type of architecture, we take power away from the outside OPC Clients and put it on the locked-down OPC Server PC. In the diagram shown, we use MatrikonOPC's Data Manager to "push" the data to MatrikonOPC's Server for Caching. The Server for Caching is configured to only allow DCOM connections from Data Manager and local COM connections. This architecture prevents arbitrarily added OPC Clients from accessing our OPC Server and can have the tags locked down just like the shielding technique shown above has.

TUNNELLING OPC

The above architectures solve the issue of OPC security, but make significant compromises in other areas. One of those areas is in firewalls, which don't work well with DCOM. OPC Tunnelling enables us to lock down all PCs by using software firewalls and removing DCOM from the system. To completely lock the data stream between OPC Client and OPC Server, encrypt the tunneled data and configure the server to

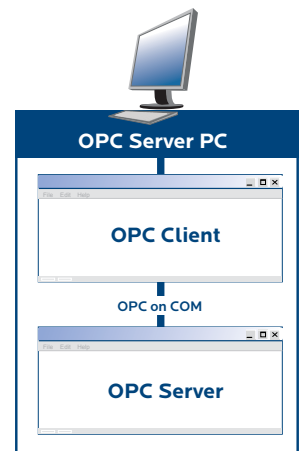


Fig 4. COM Connection

only allow tunneled connections from the OPC Client's IP address. Matrikon's OPC Tunneller enables data stream encryption and only allows access from OPC Client's you specify. In my opinion this architecture represents the best in layered security for OPC systems because it removes DCOM from the equation and allows use of firewalls on all PC's. Tunneled OPC makes OPC over the internet possible without VPN connections and their associated security configuration. Unlike XML-DA, OPC tunneled with Matrikon's OPC Tunneller does not take a significant performance hit.

COM/DCOM SECURITY TECHNIQUES

Specific COM/DCOM security is based on ACL's (Access Control Lists) and can be difficult even for IT staff. Specific settings will be omitted for length considerations. Contact your vendor for recommendations on your implementation.

LOCKING DOWN FOR A LOCAL CLIENT

To lock down a PC for a local connection we will close down DCOM, restrict the OPC Server to only allow OPC Clients running as the right user, and possibly ensure that the wrong person can never run the OPC Client with the right privileges. To close down COM/DCOM, a good technique is to create a user that only exists on the local machine, and then give launch and access permissions on your OPC Server to just that user. If it's possible to launch your OPC Client as that user without being logged in as that user, you can further lock down the station by removing the ability to interactively log in with that account.

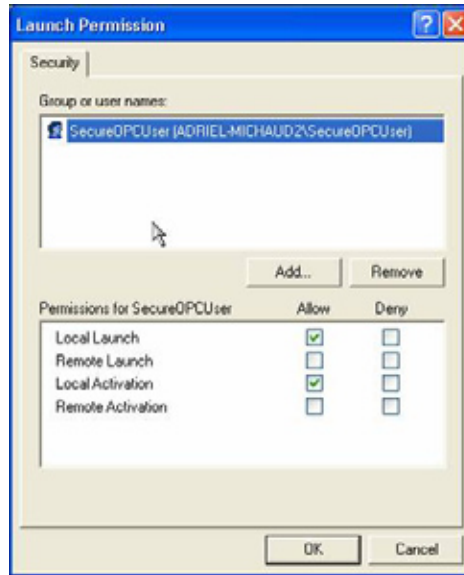


Fig 5. COM/DCOM Launch Permissions

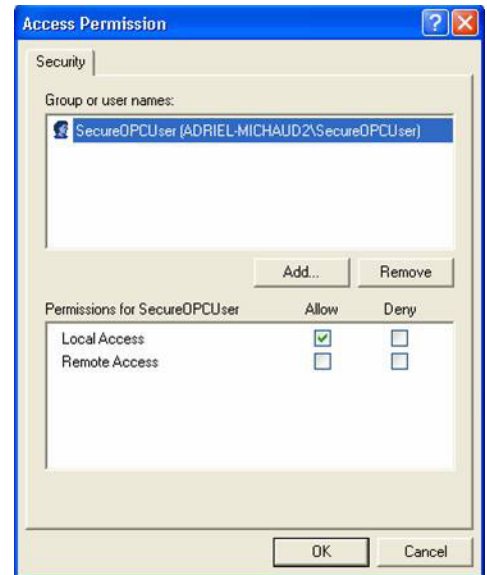


Fig 6. COM/DCOM Access Permissions

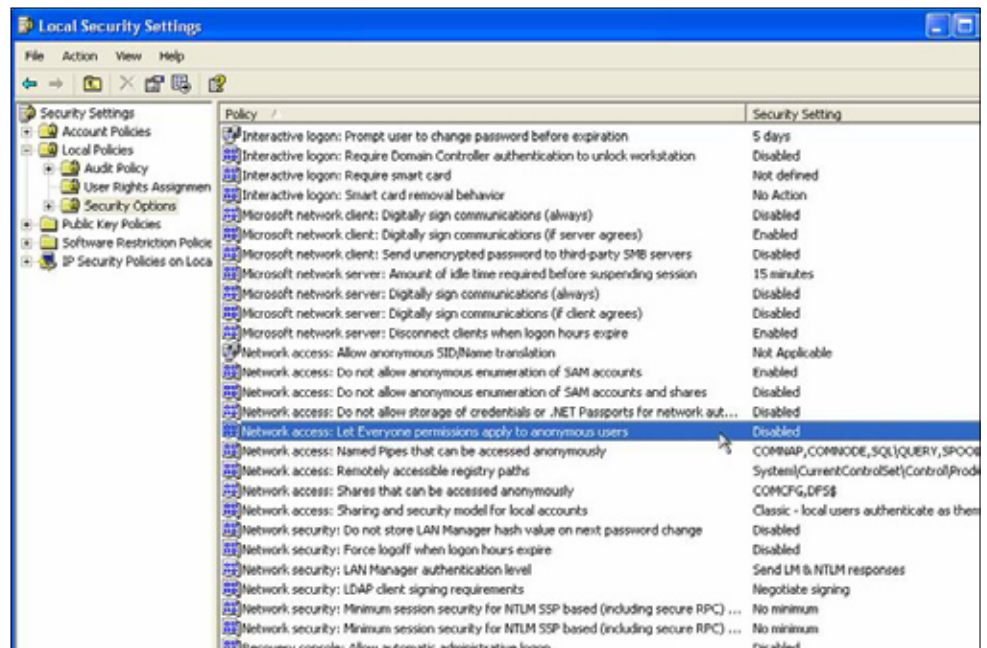


Fig 7. Local Security Policy

LOCKING DOWN FOR A REMOTE CLIENT USING DOMAINS

Quite a bit more difficult, even to make the initial connection, is the DCOM security around remote connections. The best technique is again to use a specific user. Have IT create a non-interactive user to use for DCOM communications, and set your OPC Client and Server to use this user. Restrict your OPC Server to only allow access by this user. Locking down OPCEnum, the application that lists OPC Servers, will block outside OPC Clients from

even knowing that you have any OPC Servers available. This is done through dcomcnfg.exe just like all the other DCOM settings. Depending on how you start your OPC Server, you may also deny the "Everyone" group launch permissions to prevent the possibility of launching the OPC Server from an OPC Client. If you have difficulty making a connection, open up your DCOM security completely, then start restricting it step by step.

LOCKING DOWN FOR A REMOTE CLIENT USING WORKGROUPS

Potentially the worst from a security standpoint, remote connections through workgroups are also the most difficult to even configure initial connections for. OPC Clients and Servers must usually be run as the same user with the same password to get any connection at all. Often, local security policy must be edited to enable the "Everyone permissions apply to anonymous users" option, just to be able to OPC browse the PC. This option is potentially dangerous, so we typically avoid this type of configuration when security is paramount, or we use OPC tunnelling.

SECURITY IN THE FUTURE

Currently, a number of plants do not employ any DCOM or OPC architectural security. In some of these plants, it is because the risk has been considerably reduced through other means such as complete disconnection from the outside world and impressive on-site security. For plants where security is a concern, vendors should be consulted. MatrikonOPC has experts with both OPC and network security backgrounds who can give recommendations or implement the desired level of security in your system. We are committed to supplying products and services that create secure, reliable OPC architectures that end users require in today's operating environments. To talk with

one of our OPC Solutions Architects or to arrange for an in-depth security assessment, see <http://matrikonopc.com/main/contact.aspx> or contact us at OPCExperts@MatrikonOPC.com.

Adriel Michaud, CNT, is the Lead Solution Architect for MatrikonOPC. Drawing from on-site experience with OPC, along with a previous technical support role, Adriel is the main technical consultant responsible for creating robust systems for industrial connectivity with a team of Solutions Architects.

Currently Adriel is also called upon to lend his expertise toward training new technical support specialists, and gives regular lectures to the MatrikonOPC Sales force. Adriel regularly presents workshops and webinars around the globe to promote continue to promote the OPC open connectivity standards and educate others in how OPC can aid them in constructing their systems.

Copyright © 2007 Matrikon, Inc.