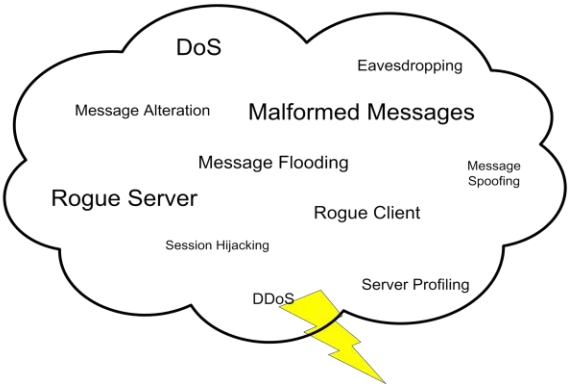
OPC UA vs OPC Classic

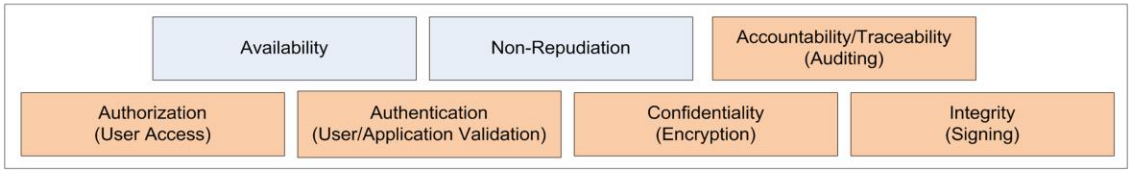
By Paul Hunkar

Security and Communication comparison

In the world of automation security has become a major source of discussion and an important part of most systems. The OPC Foundation has for a long time support secure interoperability between systems made by multiple vendors, initially with its OPC DA, OPC A&E, OPC HAD and OPC Security interfaces (collectively now termed OPC Classic) and most recently with OPC Unified Architecture (UA). These two standards provide common functionality, but OPC UA unifies and enhances the functionality provide by the OPC Foundation Classic interfaces. These two standards provide, when consider from a security point of view, a somewhat different view of security and a very different manner of implementing the view. In this paper I'll discuss the various aspect of security and how they can be met by both standards and attempting to highlight the differences.

This paper will concentrate on security as it would relate to the communication between the systems provided by vendors. Security has many additional aspects and these should also be considered when deploying a system, but this paper only focuses on the communication related aspects.





This paper will try to address some of the most common aspect of Security. Security aspects include encryption and or signing of data that is being transmitted between two systems, the identification of applications (server and or client), the authentication and authorization of the user of the client application, the transmission of data through firewalls and auditing. When considering security it is import to also consider the environment in which the applications are executing.

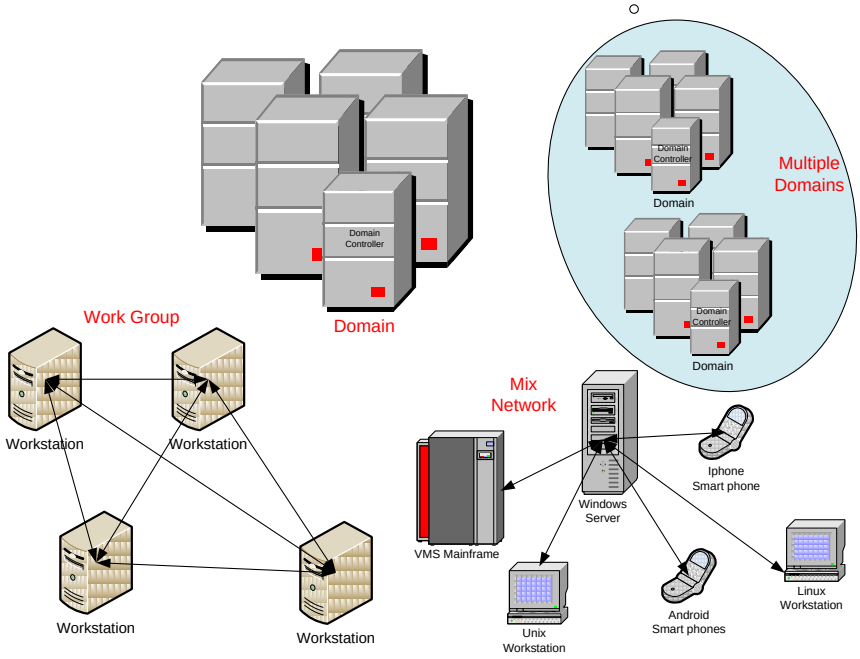
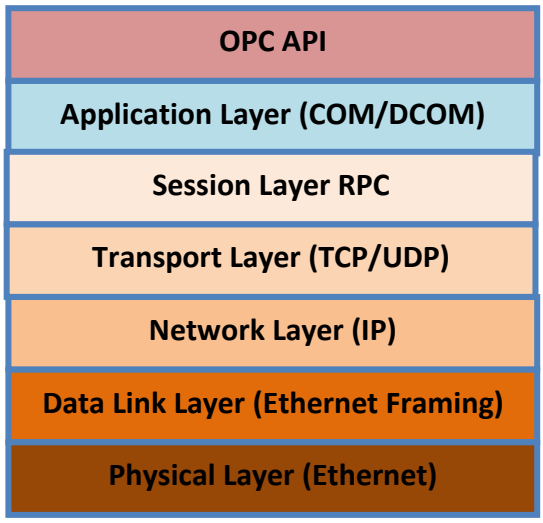


Figure 1 - Network Types

Applications could run in a Windows domain, a windows user group, on a standalone machine or in a mix operating system environment (i.e. not just windows). I'll try to talk a little about each of these and how they relate to OPC Classic vs OPC UA.

General Overview of OPC Classic and OPC UA Security

Let's discuss security in general for OPC Classic. OPC Classic as a set of specifications does not define security as part of any of the interface specifications. OPC Classic is built on top of DCOM/COM transport which is a Microsoft based communication protocol and has security



features included (see Figure 2 - OPC Classic CommunicationFigure 2). The OPC Foundation defines recommendations for how to configure the COM/DCOM layer for security via the OPC Security Specification. Documents have been generated that describe the Windows/DCOM provided security and how to apply or configure it in the world of OPC Classic (see the end of the paper for a list of related documentation). Some import configuration issues are to restrict DCOM to TCP and to restrict access. The key aspect of DCOM / Windows security is that it is based on users and the rights granted to the users (the user under which the application is executing). In short OPC Classic relies on the security provided by the underlining communication protocol and the family of Windows Operating systems. For one of the use cases, in which multiple operating systems are present OPC Classic is very difficult to implement since it requires COM/DCOM be available on the alternate operating system. There are some third party versions of DCOM available for some non-windows platforms, but in general OPC Classic is very rarely used on a non-windows platform.

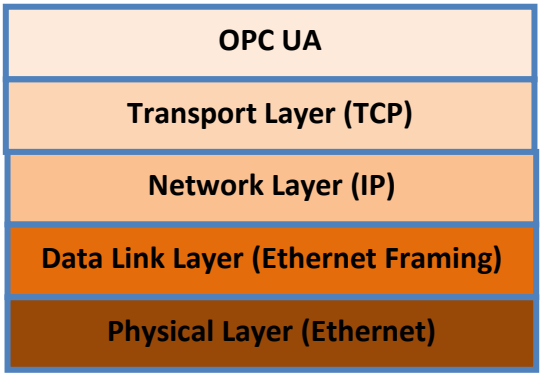


Figure 3 - OPC UA Communications

Let's discuss security in general for OPC UA. OPC UA has a specification (Part 2) that describes security threats and attacks and how the OPC UA standard is designed to mitigate these threats

and attacks. OPC UA has security related functionality included throughout the 13 part specification. OPC UA as a standard is not locked to a single communication transport or operating system, thus it defines security at a layer above the transport. This ensures that as new transports are added security will be maintained. Figure 3 illustrates a TCP transport; other transports may add an additional layer between TCP and OPC UA. The security features of OPC UA were designed to be easily enhanced as security standards improve without having to make changes to applications. It was also modelled and made use of current best practices in security (think banking).

In general both OPC Classic and OPC UA have security capabilities.

Encryption and or Signing of the data flow between systems

The signing of the data flow (message) ensures that no one can change what is sent and received. It requires the generation of a cryptographic signature that can be easily regenerated

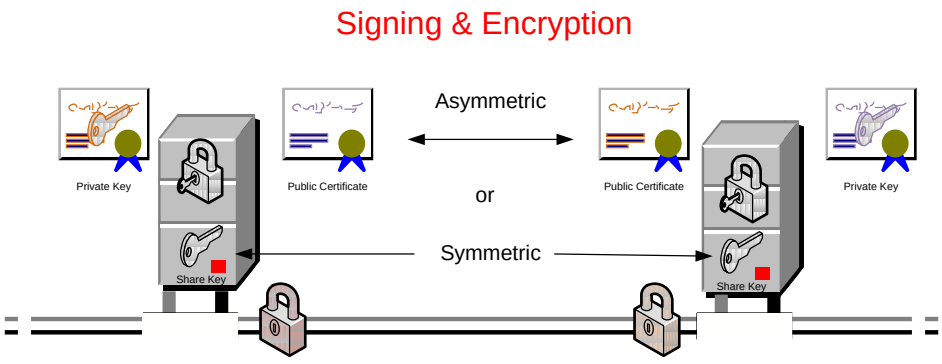


Figure 4 - Sign and Encrypt

by the receiver of the message. If anything has changed the receiver will not get the same signature and can tell the message has been altered. Encrypting takes signing to the next level in that no one but the recipient can even read what is in the message. It uses a cryptographic translation of what is in the message, so that only the receiver has the required information to decrypt the message.

- DCOM can be configured to provide both signing of data and encryption. In the DCOM world this is often discussed as ensuring the data is verifiable and it is private. The setup and configuration of it varies by version of windows and the environment in which the machine are deployed. In a Windows domain, it requires fairly simple configuration (Identifying the DCOM client(s) and server(s) and checking a box that Private communication is required). The key point is that by default DCOM does not have this enabled, but to secure communication it should be configured. In a non-domain

environment, it requires setup steps on each client and server machine, but this can be accomplished as part of the required configuration for DCOM.

- For OPC UA, The default transport options have Signing and Encryption enabled. The Server configuration can choose which communication options are available to client to use for connection (i.e. the end user can select None to disable encryption and signing or can select to only allow Sign or only Sign&Encrypt). All Certified Servers and Clients are required to support this aspect of security and it is provided by stacks and toolkits that are used to build the application. The configuration only has to be performed on servers, client just have to choose which of the available communication option to make use of. Since OPC UA was architected for multiple platforms, this functionality does not change for domain based, work groups or multiplatform based applications.

Identification and securing applications

This functionality ensures that an application is communicating only with appropriate other applications, that there is no man in the middle or rogue server or client. That the given client is only communicating with an authorized server and the server is responding only to an authorized client. This goes beyond restricting what a user does, since it restricts which application instance are allow to communicate with which other application.

Figure 5 – Identifying and Securing Application

- In DCOM this can be configured, but requires a fair amount of work since, since it requires configuration of items other than DCOM, such as firewalls and RPC. Not all OPC Classic server and clients can support this type of restriction, so testing is required. Typically this type of restriction are not configured, instead just user access is configured (see user access).
- In OPC UA, this is required and provided by default. All applications are required to support this to be certified. The same application installed on two different machines can be configured with different access rights. The access rights apply to clients as well as servers, so the same client installed on two different machines can have different servers that it is allowed to access. For example a standard client that is installed on two operator stations (for two different areas in a plant) would have different lists of

servers that they are allowed to connect to. This is accomplished in OPC UA by the use of Certificates. All Applications have a unique certificate assigned to them and a trust list that indicates which other certificates (applications) are to be trusted (allowed). The OPC Foundation provides tools (as well as vendors) for configuring certificates and certificate authorities (CA) allowing relatively easy configuration, especially for someone with IT experience. OPC UA has announced Global Discover Service functionality (part 12) that makes deploying certificate much easier for the non-IT person. This functionality, when it becomes available will allow management of all certificates from a single administrative application (such as a domain) by simple menu selections.

User access rights.

This is the restriction of what items in a server can be access and in what manner they can be accessed (read/write/browse) by a given user. For example: Is the user allowed to read values, write values, browse the address space etc. This authorization of access implies that the user has been identified and authenticated. The client must provide credentials to the server identifying the user that is executing the application. User Access restrictions can be very wide as in apply to the entire server or can be specified down to individual item in a server. User access can also be ignored, in that it could be configured for anonymous access.

Figure 6 - User Security

- DCOM typically bases its overall security on this type of server wide access. In a domain environment it can be easily configured, but by default it is not restricted. The domain would need to identify all OPC Classic servers and assign appropriate access rights to the User groups (roles) associated with the clients that will be connecting. This configuration can be pushed from the domain to each Server. Since all accounts in a domain are managed from a central controller this is easy to do, but must be done to ensure connectivity. In a non-domain environment it can be configured, but requires additional work or tricks to identify and ensure users (common accounts on all machines, common user group (roles) etc.…). For the specific case of two or more

untrusted domains, user security cannot be configured, luckily this type of configured rarely occurs. It usually only results from when two companies must share data. OPC Classic applications may provide individual access restriction for a given item vs the general DCOM read or write access, but few if any applications actually include this type of restriction. Typically it is a wide set of access rights, in that a given user is allowed to access the server for read or for read/write. Other complications can a raise in that occasionally an application is operating under a service account (think console) and the actual operator is required to login. Typically the console application is equipped to handle the impersonation required for these checks, but not all application will provide this type of functionality.

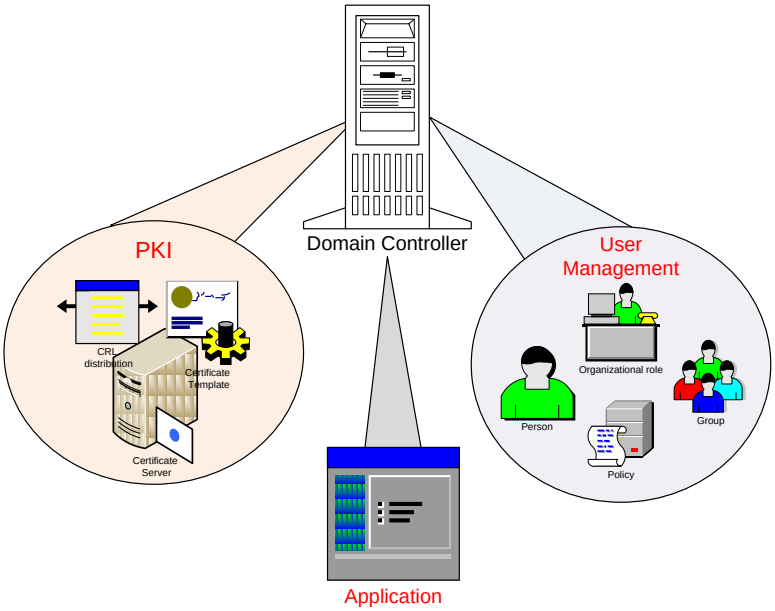


Figure 7 - Domain Controller

- In OPC UA, there are three options for identifying a user: A user account and password, Kerberos tokens or certificates. All applications are required to support Username/password, which is the simplest to implement, but can be more work to configure, since it must be configured on each application. Kerberos is a standard manner of exchanging user identities without exchanging passwords. It is supported by windows domains and easily implemented in an environment that has a Kerberos token server. Certificates are an easy extension of OPC UA, since the certificate handling is already required for identification of applications (see 2). The selection of which manner of identifying user is application specific. Once the user is identified and authenticated than the application can restrict the access rights for a given user with respect to read/write browse etc. This is just like in DCOM where it depends on the application and most applications do not actually provide individual item security, they

more typically group users into groups (engineers, operator etc.) and generate restriction by these groups server wide. OPC UA user identification works in the domain, cross domain, multiple platform and workgroup environments in similar manner.

Firewalls

Firewalls are used in systems to protect machines and access points from intrusion. They function by restricting the types of connection that are allowed, the ports to which connection are allowed and the communication protocols allowed. They are becoming mandatory in most systems to ensure security.

Figure 8 - Firewalls

- OPC Classic makes use of DCOM, which intern makes use of RPC. RPC by default makes use of dynamic port allocation. This makes it very difficult to configure a firewall, since a large range of ports would need to be opened. In addition in OPC Classic, the server needs to be able to initiate communication with a client for callbacks requiring access from a server to a client. This functionality results in requiring all client machines to also be configured as if they were server and all server machines to also be configured as if they were a client, i.e. Opening the firewall in the opposite direction. RPC can be configured to either restrict the range of ports that are used or can statically assign a fixed port to a given OPC Classic server. The fixed port assignment may not work for all OPC Classic products so a given vendors offering should be tested to ensure it will allow the static port. DCOM is a standard protocol
- OPC UA defines a fixed port or ports for a communication channel. The actual port is determined by the endpoint(s) and protocol(s) exposed by the server. The HTTP protocol can run over the default port 80, which is rarely blocked. Clients initiate all communication so no out bound communication. The fixed ports and client initiated communication result in very easy firewall configuration. The OPC Foundation configuration tool will (depending on the firewall being used) perform this configuration at the touch of a button.

Auditing

Auditing is the generation of messages that track all security related actions in a system. It includes all failed attempts to connect to the system, who successfully connects to a system, any actions they perform that alter the system, any errors or security violation that occur. The purpose of audit records is to allow an analysis of the actions in a system after some problem is detected. It can also be analyzed to detect problems.

- DCOM/COM does not by default generate any level of logs, but users can enable logging for DCOM Connection requests, System Object Access and SMB logins. By enabling these, the operating system can generate events that can be used to track activity. This still does not provide detailed information as to what the activity was, only the "who connected". What was written to would require additional auditing functionality is built into the classic application. This type of auditing is not defined as part of any OPC Classic specification.
- OPC UA defines audit messages and when they are required. An OPC UA server that supports the Auditing profile can generate audit events for all connection and security related actions. In addition it generates audit events for all user actions that make changes to the address space on the server. The auditing provided by OPC UA also supports linking of audit events from the client to the server allowing easier review of audit logs from multiple devices.

Conclusions

OPC Classic can be configured to provide a fair amount of security, but all of the security Is actually provided by the functionality in Windows and DCOM/COM. It does require a fair amount of configuration and knowledge to configure. Some environments do not support security configuration and some application cannot be configured to support all security configuration.

OPC UA was designed with security in mind and thus has security integrated throughout the set of specifications. To become certified an application must support all basic security functionality. OPC UA is designed to work in multiple environments and on multiple platforms, but as a result it is not tied to one platform and the security built into the platform.

Author

PAUL HUNKAR is president of DS Interoperability, independent consulting firm, specializing in design and development of software systems. He has been running DS Interoperability for 4 years, his client list includes Yokogawa, ABB, Unified Automation, RoviSys, the OPC Foundation. Paul is member of the Technical Advisory Council of the OPC Foundation. He is editor for the Profiles, Security, and Alarm & Condition parts of the OPC UA Specification. He was the Director of Certification for the OPC Foundation and the chair of the OPC Foundation Compliance Working Group. He is Chair of the ISA-95 OPC UA working group. He had previously worked for major Process Automation vendors for more than 25 years in a variety of engineering roles including designing, constructing and managing development of advanced control applications, operator interface systems, historical systems and investigating new technologies. Paul has a Bachelors Degree in Computer Engineering from the University of Michigan and a Masters Degree in Computer Engineering from Case Western Reserve University.

DSInteroperability,
Hudson, Ohio,
Tel. +1 440 337 4161,
E-Mail: **Paul.Hunkar@DSInteroperability.com**

References

The following documents can be used to further investigate OPC security.

OPC Security 1.00 Specification

Unified Architecture security

Securing your OPC classic control system

Using OPC via DCOM with windows XP service pack 2

OPC, DCOM and Security

The OPC UA Security Model for Administrator

OPC Security White Paper #1

OPC Security White Paper #2

OPC Security White Paper #3

OPC UA 1.02 Part 2: Security Model

OPC UA 1.02 Part 4: Services

OPC UA 1.02 Part 6: Mappings

OPC UA 1.02 Part 12: Discover