Binary Protocal of OPC-UA: Combining Protocal for Induatrial Internet

Lingqiao Zhao

School of electronics and information engineering, Tongji University, Shanghai 201804, China.

1830794@tongji.edu.cn

Abstract

OPC UA (unified architecture) is the next generation OPC standard, which provides a complete, safe and reliable cross platform architecture to obtain real-time and historical data and time. Nowadays, the core of OPC communication standard is interoperability and standardization. The traditional OPC technology solves the problem of interoperability between hardware devices at the control level, and the communication standardization at the enterprise level is also needed. This paper summarizes the existing OPC architecture, and proposes a new OPC architecture based on binary, which solves the inherent drawbacks of the existing OPC UA protocol. The experiments show that the improved opc-ua has better compatibility as the inevitable choice of factory interconnection.

Keywords

OPC UA, Communication, Standardization, Binary Protocol.

1. Introduction

In order to cope with the trend of standardization and cross platform, in order to promote OPC better, OPC foundation has launched a new OPC standard OPC UA based on the successful application of OPC in recent years. The OPC UA interface protocol includes the previous A & E, Da, OPC XML, DA or HDA. All the previous objects can be accessed with only one address space, and it is not restricted by the windows platform, because it is defined from the transport layer above the packet. This will be mentioned later, which leads to higher flexibility and security than the previous OPC.

OPC UA is based on the new generation technology provided by OPC foundation, which provides safe, reliable and independent of manufacturers, and realizes the transmission of raw data and preprocessed information from manufacturing level to production planning or ERP level. Through OPC UA, all the required information is available to each authorized application and every authorized person at any time and anywhere. This function is independent of the manufacturer's original application, programming language and operating system. At present, OPC's high reliability and extensibility of the OPC platform have been provided by OPC as an independent platform. OPC UA is no longer dependent on DCOM, but based on Service Oriented Architecture (SOA). Nowadays, OPC UA has become a bridge between enterprise layer and embedded automation system independent of Microsoft, UNIX or other operating systems.

Traditional OPC com features distribute different functions to multiple com servers and represent functions with different characteristics through interface connection. OPC COM server provides alarm, but does not provide continuous access to the data that triggers the alarm. For example, the OPC COM server that provides historical data storage does not allow the current data to be read and updated. This feature creates integration problems because information from a single system cannot be accessed in a consistent manner. OPC UA solves the integration problem of common address which contains many kinds of available information through a single service.

The core of OPC communication standard is interoperability and standardization. The traditional OPC technology solves the problem of interoperability between hardware devices at the control level, and the communication standardization at the enterprise level is also needed. The access specification before OPC UA is based on COM / DCOM technology of Microsoft, which will bring irreparable

weakness to the communication of new layer. In addition, the traditional OPC technology is not flexible enough and the platform is limited. The OPC foundation has released the latest unified data communication method, OPC unified architecture (OPC UA), which covers OPC real-time data access specification (OPC DA), OPC historical data access specification (OPC HDA), OPC alarm event access specification (OPC A & E) and OPC security protocol (OPC security) But on the basis of it, the function is extended.

2. Problems of OPC-UA

2.1 Unified access

Traditional OPC com features distribute different functions in multiple com servers, and represent different functions through interface connection. OPC COM server provides access to alarm data that triggers alarms but does not continue to provide continuous. For example, OPC com servers that provide storage of historical data do not allow current data to be read and updated. This feature creates integration problems because information from a single system cannot be accessed in a consistent way. OPC UA solves the integration problem of common address access through a single service, which contains a variety of available information.

2.2 Authentication Interoperability

OPC UA features the same server and client testing tools as the successful OPC com authentication program. These testing tools enable suppliers to verify whether their products meet the requirements of characteristics and improve the quality of products. After the OPC UA and OPC com features are certified, the corresponding certification marks can be obtained. By using OPC certified products, the system integration cost of end users can be reduced. OPC UA is designed for usability and redundancy architecture. Complete configurable timeout, error detection, and recovery features enable OPC UA products to seamlessly handle errors or failures (such as loss of network communication). The standard OPC UA module supporting redundant functions makes it possible to deploy applications from different manufacturers.

2.3 Cross domain firewall and Internet

OPC UA features the same server and client testing tools as the successful OPC com authentication program. These testing tools enable suppliers to verify whether their products meet the requirements of characteristics and improve the quality of products. After the OPC UA and OPC com features are certified, the corresponding certification marks can be obtained. By using OPC certified products, the system integration cost of end users can be reduced. OPC UA is designed for usability and redundancy architecture. Complete configurable timeout, error detection, and recovery features enable OPC UA products to seamlessly handle errors or failures (such as loss of network communication). The standard OPC UA module supporting redundant functions makes it possible to deploy applications from different manufacturers.

2.4 Standard security model

In the past, security issues were considered last, and many vendors did not test their product security licenses. This means that it is difficult or impossible for end users to configure security. OPC UA architecture solves this problem through the standard security model that UA application must implement. This enhances interoperability and reduces configuration and maintenance costs. Opcua is also conducive to the development of standard tools for security settings management of any opcua product on any platform. Lightweight OPC UA can be used as an effective binary communication protocol. For example, OPC UA has been transplanted to many embedded systems, including VxWorks, Linux and proprietary RTOSs (real time operating systems). Top level OPC UA applications support enterprise standard XML page service protocol. The cost of system integration can be reduced through a common architecture.

3. Benefits of binary protocal

OPC UA PubSub is an extension of the popular OPC UA protocol, which allows applications to publish messages to multiple subscribers through intermediate agents such as mqtt agents. OPC uapubsub messages can be formatted in XML, JSON, or efficient OPC UA binary formats. When using the latter approach, the publisher can encrypt and digitally sign the message before it is sent to the agent to ensure that no one other than the target recipient can read or modify the message. This will protect the publisher's data, even if the agent is stored on disk while the message is waiting to be delivered.

Publishers and subscribers need to share a key to communicate securely. This is achieved by using a special OPC UA server called "secure key service" (sks). Applications that require a key provide credentials to sks securely using the OPC UA client server protocol. Sks determines whether they have access to the requested key and returns one or more keys. This process requires multiple keys because they change frequently to ensure that the application can be removed from the system in a reasonable amount of time if necessary.

The rich OPC UA information model allows applications to represent their systems using terms and constructs that match the application domain. This information model is also an important part of OPC UA PubSub because it provides an overall framework for describing the content of messages to be sent to broker. The OPC UA PubSub specification defines a mechanism for notifying subscribers that the message structure has changed, and allows publishers to send new metadata to in band or out of band. Other solutions expect subscribers to process known message content or rely on specific rules to parse JSON or XML. our example below.

4. Conclusion

OPC UA PubSub is an extension of the popular OPC UA protocol, which allows applications to publish messages to multiple subscribers through intermediate agents such as mqtt agents. Publishers and subscribers need to share a key to communicate securely. This is achieved by using a special OPC UA server called "secure key service" (sks).

Our updated opc-ua protocol based on binary system performs well in standard security model, cross domain firewall and Internet, authentication interoperability and authentication interoperability. Any factory operator who wants to use the Internet of things will benefit from the security toolbox provided by OPC UA, which helps to ensure the security and integrity of privacy when data flows are thrown out of a system managed by a third party. When combined with OPC UA information modeling framework and widely used in many existing OPC products, the toolbox will become the leading tool. This paper summarizes the existing OPC architecture, and proposes a new OPC architecture based on binary, which solves the inherent drawbacks of the existing OPC UA protocol. The experiments show that the improved opc-ua has better compatibility as the inevitable choice of factory interconnection.

References

[1] G. Martinov, A. Issa and L. Martinova, "Controlling CAN Servo Step Drives and Their Remote Monitoring by Using Protocol OPC UA," 2019 International Multi-Conference on Industrial Engineering and Modern Technologies (FarEastCon), Vladivostok, Russia, 2019, pp. 1-5, doi: 10.1109/ FarEastCon.2019.8934338.

[2] C. Petre and A. Korodi, "Honeypot Inside an OPC UA Wrapper for Water Pumping Stations," 2019 22nd International Conference on Control Systems and Computer Science (CSCS), Bucharest, Romania, 2019, pp. 72-77, doi: 10.1109/CSCS.2019.00020.

[3] C. Eymüller, J. Hanke, A. Hoffmann, M. Kugelmann and W. Reif, "Real-time capable OPC-UA Programs over TSN for distributed industrial control," 2020 25th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), Vienna, Austria, 2020, pp. 278-285, doi: 10.1109/ETFA46521.2020.9212171.

[4] Y. H. Lai, Y. -H. Huang, C. F. Lai, S. Y. Chen and Y. -C. Chang, "Dynamic Adjustment Mechanism based on OPC-UA Architecture for IIoT Applications," 2020 Indo – Taiwan 2nd International Conference on Computing, Analytics and Networks (Indo-Taiwan ICAN), Rajpura, Punjab, India, 2020, pp. 335-338, doi: 10.1109/Indo-TaiwanICAN48429.2020.9181337.

[5] C. Eymüller, J. Hanke, A. Hoffmann, M. Kugelmann and W. Reif, "Real-time capable OPC-UA Programs over TSN for distributed industrial control," 2020 25th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), Vienna, Austria, 2020, pp. 278-285, doi: 10.1109/ETFA46521.2020.9212171.

[6] H. Cho and J. Jeong, "Implementation and Performance Analysis of Power and Cost-Reduced OPC UA Gateway for Industrial IoT Platforms," 2018 28th International Telecommunication Networks and Applications Conference (ITNAC), Sydney, NSW, 2018, pp. 1-3, doi: 10.1109/ ATNAC. 2018.8615377.