2017

An investigation into some security issues in the DDS messaging protocol

Thomas White
Edith Cowan University

Michael N. Johnstone
Edith Cowan University

Matthew Peacock
Edith Cowan University

DOI: 10.4225/75/5a84fcff95b52
Originally published as: White, T., Johnstone, M.N. Peacock, M. (2017). An investigation into some security issues in the DDS messaging protocol. In Valli, C. (Ed.). (2017). The Proceedings of 15th Australian Information Security Management Conference, 5-6 December, 2017, Edith Cowan University, Perth, Western Australia. (pp.132-139).

AN INVESTIGATION INTO SOME SECURITY ISSUES IN THE DDS MESSAGING PROTOCOL

Thomas White[2], Michael N. Johnstone[1, 2], Matthew Peacock[1, 2]
[1]Security Research Institute, [2]School of Science, Edith Cowan University, Perth, Western Australia
{thomas.white, m.johnstone, m.peacock}@ecu.edu.au

Abstract

The convergence of Operational Technology and Information Technology is driving integration of the Internet of Things and Industrial Control Systems to form the Industrial Internet of Things. Due to the influence of Information Technology, security has become a high priority particularly when implementations expand into critical infrastructure. At present there appears to be minimal research addressing security considerations for industrial systems which implement application layer IoT messaging protocols such as Data Distribution Services (DDS). Simulated IoT devices in a virtual environment using the DDSI-RTPS protocol were used to demonstrate that enumeration of devices is possible by a non-authenticated client in both active and passive mode. Further, modified sequence numbers were found to be a potential denial of service attack, and malicious heartbeat messages were fashioned to be effective at denying receipt of legitimate messages.

Keywords*:* Data Distribution Services, DDS, Critical Infrastructure, Cyber-physical systems, Internet of Things, Network Security

INTRODUCTION

The Internet of Things (IoT) refers to the multitude of interconnected computers, sensors, controllers, and other devices which interact with the physical world. Ubiquitous computing devices are the driving force behind technologies such as smart electrical grids, autonomous cars, wearable health devices and home automation. Evans (2011) made the frequently cited prediction that the number of connected devices would surpass 50 billion by the year 2020, however, revised predictions have forecast the number of devices to be significantly fewer, with Gartner, Inc. (2017) forecasting 20.8 million devices by 2020. Even with a revised prediction this is still a significant number of devices reinforcing the need for robust security considerations.

Potential vulnerabilities in IoT messaging protocols could have serious repercussions if exploited. Whilst in theory industrial networks should be robust, this is not always the case, and the impact of unauthorised access or data modification within these networks could be quite severe. Interruptions or compromise of a power grid by exploiting OPC UA information transfer, intercepting personal health data through a poorly-secured CoAP-based health tracker, or attacking a DDS-based tactical control system in a military vessel are examples of potential attacks which, if successful, could have serious consequences for critical infrastructure.

Physical damage is major concern for industrial cyber-physical systems, but cyber-attacks in general are also creating a significant cost for organisations. IBM and the Ponemon Institute (2016) have stated in their Cost of Data Breach study that the average data breach in Australia comes at a cost to the breached organisation of $2.64 million, at an average cost of $142 per stolen record. Analysing and understanding vulnerabilities in IoT protocols can assist organisations in evaluating how their risk appetite may influence protocol choice when making architectural design decisions.

This research aims to test if identified vulnerabilities that appear to be present in parts of the DDS protocol are realisable. The remainder of the paper describes the security landscape for Industrial IoT systems, defines the experimental methodology used and discusses the findings of the research.

SECURITY ISSUES IN IOT SYSTEMS

Historically, protocol security has been an avenue for exploitation. For example, DNS, FTP, ICMP and EAP are protocols which have had vulnerabilities in their design, rather than programming errors in implementations of the protocols. Even recently ratified protocols such as HTTP/2 have been found to contain vulnerabilities (Imperva, 2016). In addition to common protocols in use on the Internet, continued research has revealed vulnerabilities in control systems protocols, for example BACnet (Peacock & Johnstone, 2014) and DNP3

(Crain & Bratus, 2015) demonstrating that continuing analysis of these protocols can reveal further weaknesses and reinforcing that control systems are a continued focus for security vulnerability analysis.

The security of Industrial Control Systems (ICS) has been viewed as a cause for concern in recent times (Harp & Gregory-Brown, 2016). Many legacy control systems run on standards, protocols and software designed and implemented at a time when the threat landscape was primarily physical based, due to less interconnection between devices. However, in an interconnected world, ICS are gaining attention from cyber adversaries. For example, in 2015 Ukraine's power grid was attacked (Lee, Assante & Conway, 2016) and availability severely compromised after attackers gained access to SCADA systems and shut down parts of the grid. This was one of the first known successful cyber-attacks on power infrastructure, highlighting the growing threat of sophisticated attack operations against cyber-physical infrastructure.

Data Distribution Services or DDS (Object Management Group, 2015) is an open standard primarily intended for peer-to-peer inter-device communications. This protocol defines a data-centric publish/subscribe model and is focussed on low latency communications between devices, rather than between a device and a server or between two servers. The specification defines DDS as:

"... a Data-Centric Publish-Subscribe (DCPS) model for distributed application communication and integration. This specification defines both the Application Interfaces (APIs) and the Communication Semantics (behaviour and quality of service) that enable the efficient delivery of information from information producers to matching consumers." (Object Management Group, 2015, p. 1)

DDS has found uses in many critical environments, such as amongst the energy and aerospace industries, as well as the military. Wang et al. (2008) explored the use of DDS in network-centric operations and warfare systems, demonstrating the increased use of these protocols in environments where security is essential. This is unsurprising as the DDS protocol has broad usage in military applications, having originally been developed by Thales (2015) for use in their TACTICOS Combat Management System. This usage has been one of the primary drivers for the high performance and resilient design requirements of DDS. DDS defers to TLS to provide the bulk of security rather than providing security at the application layer. However, reliance on TLS is clearly not sufficient, given the creation of a standardised post-protocol ratification security specification (aptly named DDS Security). This additional specification provides "authentication, authorization, non-repudiation, confidentiality and integrity" (Object Management Group, 2016) to DDS implementations. He & Liang (2015) have analysed the DDS specification for security issues and put forward a scenario where unauthorised publishers or subscribers may be able to inject data into the DDS network or receive data not intended for the legitimate recipient. They present a high-level overview of theoretical attacks on DDS and it is these types of attacks that DDS Security has been designed to mitigate. Unfortunately, at this point there appears to be limited research on the effectiveness of the DDS Security specification in mitigating the defined theoretical attacks.

Given the range of vulnerable network protocols in use in the IoT, and the associated cost of data breaches; further research is necessary to reduce the attack surface of critical infrastructure installations. The following section describes a series of laboratory experiments undertaken which aims to test a subset of vulnerabilities specific to the DDS protocol.

RESEARCH METHOD

The research was designed as a number of laboratory experiments. A combination of appropriate hardware and software resources were used to attempt to detect, capture, and then analyse specific communication used by a selection of devices using an implementation of the DDS protocol (DDSI-RTPS). The specific research questions were:

1. What risks do vulnerabilities in IoT messaging protocols introduce to IIoT networks and critical infrastructure?
    a. Are there theoretical vulnerabilities present in the Real-Time Publish Subscribe DDS Interoperability Standard protocol specification?
    b. If so, can these vulnerabilities be tested with simulated IoT devices in an isolated environment?

The hypotheses supporting the research questions and experiments designed to test the hypotheses are listed in ***Table 1*** and ***Table 2*** respectively.

Table 1: Hypotheses derived from research questions

Hypotheses
H_1: Enumeration of devices is possible by a non-authenticated client.
H_2: Sequence number and heartbeat messages can be formulated to deny receipt of messages in a DataReader.

Table 2: Experiments designed to test hypotheses

Experiment	**Description**	**Hypothesis**
E_1: Participant Enumeration (Passive**)**	To identify and enumerate RTPS participants on a network	H_1
E_2: Participant Enumeration (Active)	To identify and enumerate RTPS participants on a network	H_1
E_3: Heartbeat Spoofing	To deny receipt of messages to RTPS participants on a network	H_2

Materials:

The virtual lab consisted of four virtual machines, representing devices in the scenario connected by a virtual switch representing a DDS bus. The network topology is shown as ***Figure 1***. All simulation and data collection occurred within an isolated, controlled laboratory environment, therefore the risk of unauthorised access to systems when testing for vulnerabilities was minimised.

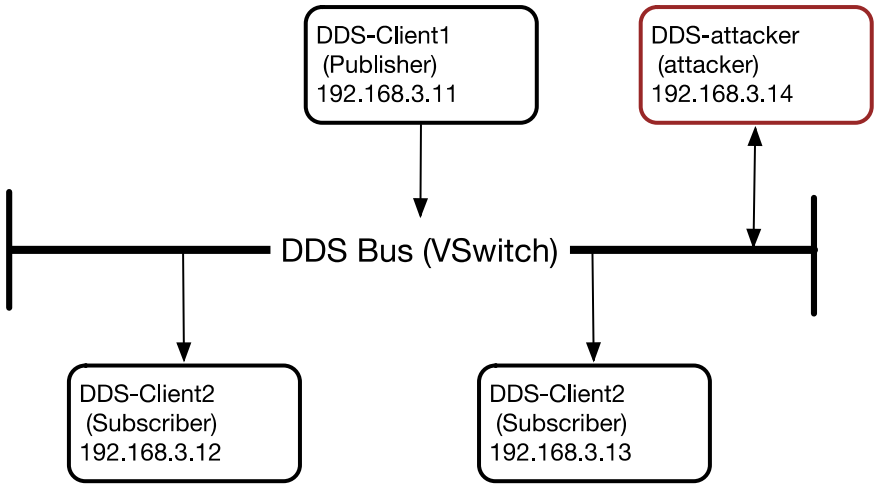

Figure 1: Scenario Network Topology

The virtual machines are listed in ***Table 3***. Note that RTPS participants may generally contain both *DataWriters* and *DataReaders*, which is the case for the attacker virtual machine in research.

Table 3: Virtual machines used in experiments

Hostname	**IPAddress**	**Operating System**	**Purpose**
DDS-client1	192.168.3.11	Ubuntu 17.04	RTPS Participant (Example Publisher)
DDS-client2	192.168.3.12	Ubuntu 17.04	RTPS Participant (Example Subscriber)
DDS-client3	192.168.3.13	Ubuntu 17.04	RTPS Participant (Example Subscriber)
DDS-attacker	192.168.3.14	Kali 2017.1	Testing remote experiments (Attacker)

ANALYSIS AND DISCUSSION

Enumeration

Information gathering is crucial for any attacker when attempting to penetrate a network, and no less so in industrial systems. DDSI-RTPS is reasonably verbose by default, providing reliably identifiable traffic. ***Figure 2*** shows the output of a Python script executed from the attacker, which successfully detects multicast RTPS SPDP packets transmitted on the local network segment as part of E_1. The information that can be obtained from a single SPDP message include: *Host IP address*, *RTPS GUID Prefix*, *RTPS Version*, *vendor ID*, *Time synchronisation information* and the Contents of *Submessages*.

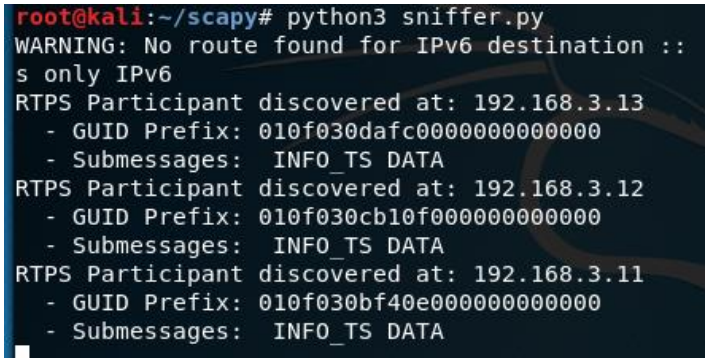

Figure 2: Passive Network Scan and Enumeration Output

In ***Figure 2*** the *Source Address*, *GUID prefix* and overall *Submessages* are displayed. The result of E_1 provides support for H_1 (Enumeration of devices is possible by a non-authenticated client).

The packet capture reconstruction in ***4*** demonstrates the DDSI-RTPS Discovery announcement from the attacker (192.168.3.14) to each scanned address and the associated response. For clarity, only a small range of the network address space was scanned in this simulation (192.168.3.10 - 192.168.3.15).

The packet capture has been colour coded as:

- Yellow indicates legitimate communication between the 3 RTPS participants;
- Red indicates traffic from the attacker; and
- Blue indicates a response to the attacker's discovery message.

Table 4: Active Network Scan and Response Capture

No.	Time	Source	Destination	Protocol	Length	Info
46	12.11818	192.168.3.13	192.168.3.11	RTPS	106	INFO_DST, ACKNACK
47	12.11818	192.168.3.12	192.168.3.11	RTPS	106	INFO_DST, ACKNACK
49	14.00157	192.168.3.11	192.168.3.12	RTPS	154	INFO_TS, DATA, HEARTBEAT
50	14.00157	192.168.3.11	192.168.3.13	RTPS	154	INFO_TS, DATA, HEARTBEAT
51	14.11858	192.168.3.13	192.168.3.11	RTPS	106	INFO_DST, ACKNACK
52	14.11859	192.168.3.12	192.168.3.11	RTPS	106	INFO_DST, ACKNACK
53	14.16125	192.168.3.14	192.168.3.10	RTPS	206	DATA(p)
54	14.1615	192.168.3.12	192.168.3.14	RTPS	270	INFO_TS, DATA(p)
55	14.1615	192.168.3.12	192.168.3.14	RTPS	106	INFO_DST, ACKNACK

Table 4 shows the result of an active scanning and response experiment (E_2) which provides further support for H_1 (Enumeration of devices is possible by a non-authenticated client). Thus H_1 is accepted, given that DDSI-RTPS can provide reliable communications over an unreliable communication medium or best-effort protocols.

Denial of Service

DDSI-RTPS uses *HEARTBEAT* messages sent from a *DataWriter* to a *DataReader* to indicate available sequence numbers on the writer so that the reader can synchronise and determine if any messages are missing. The reader may respond with an *ACKNACK* to indicate to the writer any messages which may be missing, or if the writer has specifically requested a mandatory *ACKNACK* from the reader by setting the *FINAL* flag in the *HEARTBEAT* message.

It was theorised that advancing the sequence number state on the reader may cause the reader to miss legitimate messages if the reader transitioned to a state where it is expecting a higher sequence number than the writer is currently using.
Initial experimentation was conducted through extracting the appropriate DDSI-RTPS *HEARTBEAT* message from a packet capture and modifying the *GUID Prefix*, *entity ID* and *sequence number* fields. With the altered *GUID Prefix* reference implementation, test programs stopped processing once the 'malicious' *HEARTBEAT* messages were sent. The experiment was repeated with varying sequence numbers. Once the legitimate *DataWriter* reached the sequence number provided by the *attacker*, the subscriber would recommence processing messages from the attacker provided sequence number, messages between the last real and attacker provided sequence number are not transmitted.

The specification defines certain conditions in which a *DataReader* must treat a sequence number as invalid and thus the entire *HEARTBEAT submessage* as invalid. These conditions include:

- Negative sequence numbers (The SequenceNumber data structure is signed, however negative sequence numbers are invalid); and
- Last sequence number < first sequence number.

In the conducted experiments, sending a negative sequence number, or sending a sequence number which is lower than the sequence number most recently allocated by the legitimate *DataWriters* had no effect on the processing of messages by the *DataReaders.*

*Table **5*** shows an extract of the packet capture taken during E_3. Once the attacker (192.168.3.14) sends a malicious *HEARTBEAT Submessage* (packet 504), the *DDS-client2* acknowledges the new sequence number (packet 505), then stops responding to the *HEARTBEAT Submessages* from the legitimate *DataWriter (192.168.3.11)*. This result supports H_2 (Sequence number and heartbeat messages can be formulated to deny receipt of messages in a *DataReader*).

Table 5: Network Packet Capture of HEARTBEAT Experiment

No.	Time	Source	Destination	Protocol	Length	Info
490	59.89478	192.168.3.11	192.168.3.13	RTPS	154	INFO_TS, DATA, HEARTBEAT
491	59.89478	192.168.3.11	192.168.3.12	RTPS	154	INFO_TS, DATA, HEARTBEAT
493	59.94067	192.168.3.11	192.168.3.13	RTPS	94	HEARTBEAT
494	59.94067	192.168.3.11	192.168.3.12	RTPS	94	HEARTBEAT
495	60.01186	192.168.3.13	192.168.3.11	RTPS	106	INFO_DST, ACKNACK
496	60.01186	192.168.3.12	192.168.3.11	RTPS	106	INFO_DST, ACKNACK
504	60.82424	192.168.3.14	192.168.3.12	RTPS	94	HEARTBEAT
505	60.94139	192.168.3.12	192.168.3.11	RTPS	110	INFO_DST, ACKNACK
508	61.89489	192.168.3.11	192.168.3.13	RTPS	154	INFO_TS, DATA, HEARTBEAT
509	61.89489	192.168.3.11	192.168.3.12	RTPS	154	INFO_TS, DATA, HEARTBEAT
510	61.98716	192.168.3.11	192.168.3.13	RTPS	94	HEARTBEAT
511	61.98717	192.168.3.11	192.168.3.12	RTPS	94	HEARTBEAT
512	62.01184	192.168.3.13	192.168.3.11	RTPS	106	INFO_DST, ACKNACK

Research question one posited, "What risks do vulnerabilities in IoT messaging protocols introduce to IIoT networks and critical infrastructure?" In relation to DDSI-RTPS, the vulnerabilities introduced could cause significant risk in an industrial control network. Reconnaissance is often the first task undertaken by a cyber adversary, results from E_1 and E_2 show that an attack can passively and actively survey the DDSI-RTPS network to discover all devices running on the bus. Modification of the sequence numbers can result in loss of message transmission between devices on the DDSI-RTPS network. Given the ability to forge malicious *HEARTBEAT* messages, H_2 can be accepted, as a device which has received the malicious packet is prevented from processing further messages. Given that industrial control systems often do not directly employ network-monitoring software, but rather gain system insight via system specific data collection such as Trending or Polling, this type of attack may go unnoticed, or not identified as a cyber-attack for a duration longer than is typical of IoT based networks. With the acceptance of both H_1 and H_2, this paper argues that the introduction of vulnerable IoT messaging protocols into IIoT networks increases the ability of cyber adversaries to undertake reconnaissance of industrial control system networks, and impede the availability of critical systems operating in the network.

CONCLUSION

This research set out to examine security flaws in the DDS protocol (specifically, the Real-Time Publish Subscribe extension). There was theoretical evidence that the protocol could be suborned. The experiments undertaken suggest that the identified theoretical vulnerabilities are present in the Real-Time Publish Subscribe DDS Interoperability Standard protocol specification, answering Research Question 1a. The vulnerabilities were tested with simulated IoT devices in an isolated environment, with acceptance of both H_1 and H_2, answering Research Question 1b affirmatively. The experiments undertaken suggest that enumeration of IIoT devices communicating with DDSI-RTPS is possible by a non-authenticated client in both passive and active mode, respectively. Additionally, modified sequence numbers were found to be largely ineffective at preventing messages from reaching *DataReaders.* However, if a large enough sequence number is provided, in relation to the current sequence number, a denial of service attack is effectively achieved. Additionally, malicious heartbeat messages sent from an attacker device can be crafted to deny receipt of messages between a *DataWriter* and *DataReader*. Given these results, incorporating vulnerable IoT protocols such as DDSI-RTPS into IIoT, which manage critical infrastructure without mitigating the vulnerable protocol increases the risk of cyber adversaries conducting reconnaissance and impeding the availability of critical device-to-device network communication.

REFERENCES

Crain, J. A., & Bratus, S. (2015). Bolt-On Security Extensions for Industrial Control System Protocols: A Case Study of DNP3 SAv5. IEEE Security Privacy, 13(3), 74–79. doi:10.1109/MSP.2015.47

Evans, D. (2011). The Internet of Things: How the Next Evolution of the Internet Is Changing Everything. Retrieved from http://www.cisco.com/c/dam/en_us/about/ac79/docs/innov/IoT_IBSG_0411FINAL.pdf

Gartner, Inc. (2017). Gartner Says 8.4 Billion Connected. Retrieved from http://www.gartner.com/newsroom/id/3598917Dineen, M., & Cahill, V. (2001). Towards an open architecture for Real-time Traffic Information Management. *Proceedings of the 8th World Congress on Intelligent Transport Systems*. Sydney, Australia.

Harp, D., & Gregory-Brown, B. (2016). SANS 2016 State of ICS Security Survey. SANS. Retrieved from https://www.sans.org/reading-room/whitepapers/analyst/2016-state-ics-security-survey-37067

He, Z. Y., & Liang, Y. (2015). Study on the DDS Network Information Security Technology. Applied Mechanics and Materials; Zurich, 738–739, 1213–1216. doi:10.4028/www.scientific.net/AMM.738-739.1213

IBM, & Ponemon Institute. (2016). 2016 Cost of Data Breach Study: Australia. Retrieved from https://www-03.ibm.com/security/au/data-breach/index.html

Imperva. (2016). HTTP/2: In-depth analysis of the top four flaws of the next generation web protocol. Retrieved from https://www.imperva.com/docs/Imperva_HII_HTTP2.pdf

Lee, R. M., Assante, M. J., & Conway, T. (2016). Analysis of the cyber attack on the Ukrainian power grid. SANS Industrial Control Systems.

Object Management Group. (2015). DDS. Retrieved from http://www.omg.org/spec/DDS/1.4/PDF/

Object Management Group. (2016). DDS Security. Retrieved from http://www.omg.org/spec/DDS-SECURITY/

Peacock, M., & Johnstone, M. (2014). An analysis of security issues in building automation systems. Australian Information Security Management Conference. doi:10.4225/75/57b691dfd9386

Thales. (2015). TACTICOS. Retrieved from https://www.thalesgroup.com/sites/default/files/asset/document/thales_tacticos.pdf

Wang, N., Schmidt, D. C., Hag, H. van't, & Corsaro, A. (2008). Toward an adaptive data distribution service for dynamic large-scale network-centric operation and warfare (NCOW) systems. In MILCOM 2008 - 2008 IEEE Military Communications Conference (pp. 1–7). doi:10.1109/MILCOM.2008.4753364