Autonomic computing architecture for SCADA cyber security

Nazir, Sajid; Patel, Shushma; Patel, Dilip

Published in:
International Journal of Cognitive Informatics and Natural Intelligence

DOI:
10.4018/IJCINI.2017100104

Publication date:
2017

Document Version
Publisher's PDF, also known as Version of record

Autonomic Computing Architecture for SCADA Cyber Security

1. INTRODUCTION

Cognitive computing relates to intelligent computing platforms that are based on the disciplines of artificial intelligence, machine learning, and other innovative technologies. These technologies can be used to design systems that mimic the human brain to learn about their environment and can autonomously predict an impending anomalous situation. IBM first used the term 'Autonomic Computing' in 2001 to combat the looming complexity crisis (Ganek and Corbi, 2003). The concept has been inspired by the human biological autonomic system. An autonomic system is self-healing, self-regulating, self-optimising and self-protecting (Ganek and Corbi, 2003). Therefore, the system should be able to protect itself against both malicious attacks and unintended mistakes by the operator.

Supervisory Control and Data Acquisition (SCADA) systems are used to monitor and control complex infrastructures of national importance such as transportation networks, power generation and manufacturing plants. SCADA systems can be visualised as a layered architecture, as shown in Figure 1. The field devices (sensors, etc.) at the lowest layer interact with the physical processes. At layer 2, the Programmable Logic Controllers (PLC), and Remote Terminal Units (RTUs) aggregate data values from the lower layer and communicate the commands and their responses through the communications network to the SCADA server and Human Machine Interface (HMI). The generation of commands at the top layer and collection of responses from the lowest layer results in the monitoring and control of the process. The applicability of SCADA systems has become widespread due to industrial automation, cost reduction and growth in global economies (Nazir *et al.*, 2017).

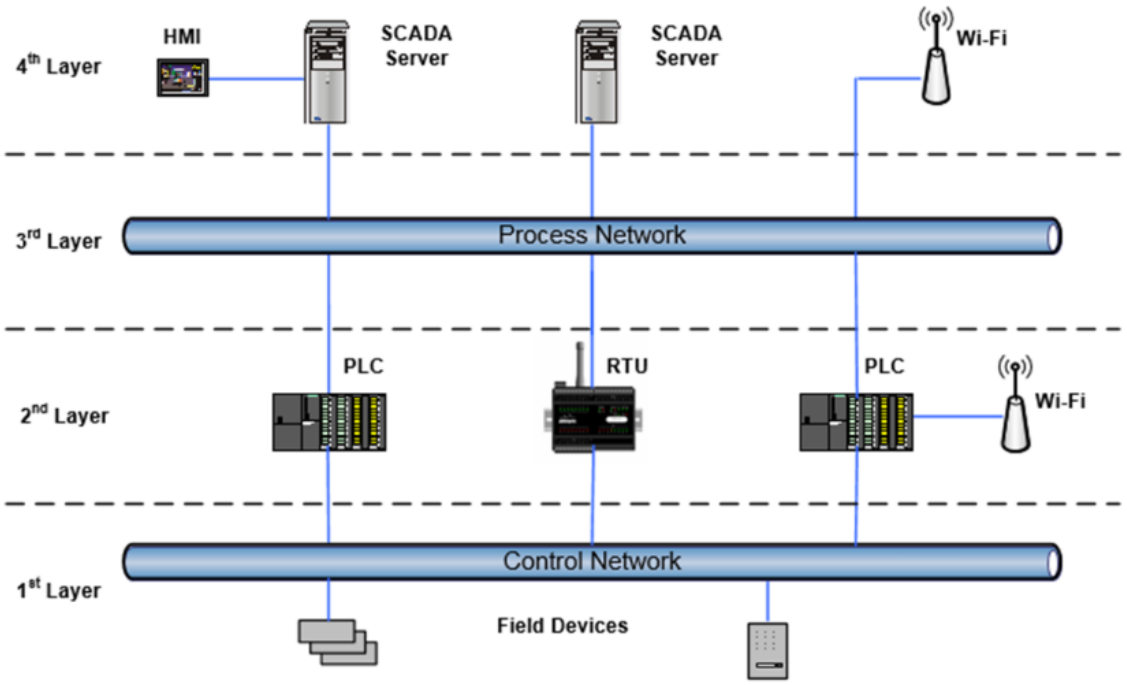

Fig. 1. Layered Architecture of a SCADA system.

Traditionally, SCADA systems were developed as closed systems with security being the overriding factor, and no Internet connectivity. However, to leverage efficiency and gain a competitive advantage,

the systems are increasingly becoming connected to the Internet and cloud technologies. SCADA system security vulnerabilities were first highlighted by the Stuxnet attack (Karnouskos, 2011). Subsequently, there has been an increase in the frequency and sophistication, of the attacks as evidenced by Constantin (2014).

Isolation and obscurity as a mechanism for protection is no longer an option for critical infrastructures (Mahoney and Gandhi, 2011). At the same time systems are getting so complex that it is difficult to develop effective defence strategies, as there is a lack of understanding of the complex interactions between the many system entities (Khadraoui and Feltus, 2015). Digital forensics becomes difficult due to the increased numbers and complexity of the cases (Taveras, 2013). The systems complexity and interactions go beyond the capability of system developers and integrators as a result of interconnectivity (Kephart and Chess, 2003). Thus, increasingly there is a lack of understanding of the holistic system, which makes it very difficult to tune a system and to make decisions in case of changed requirements. This has led to a realization that conventional and inflexible techniques will not help. What is needed is a new way of looking at the problem of cyber security that is robust, manageable and self-realising with a minimum requirement to monitor systems to make decisions. What is proposed is an entirely new way of thinking about the problem where the system itself is intelligent and helps to maintain and extend its behaviour, with the use of autonomic computing (Kephart and Chess, 2003).

The basic principles of autonomic computing are highly relevant for the protection of the increasingly complex SCADA system because: (i) the boundaries between physical and virtual systems have been blurred through virtualisation. It is possible to host a cluster of machines in a virtual environment; (ii) even with hardware there are sufficient advances in other domains with self-healing materials; (iii) advances in machine learning, artificial intelligence and the knowledge base need to be capitalised for protection; (iv) the systems are highly interconnected and the distributed nature of the systems pose an exponential complexity.

There has been some research on autonomic computing applications to complex SCADA systems. The application of autonomic computing for smart grids has been discussed (Greer and Rodriguez-Martinez, 2012) as a solution to manage system complexities. Key components of a self-protecting SCADA system have been proposed and a survey of techniques provided for the realisation of such systems (Chen Abdelwahed, 2014). Also, there are few dedicated research groups (Autonomic Computing Lab; Cloud and Autonomic Computing Centre; Fortes *et al*., 2014) focusing research on the applicability of autonomic computing to cyber security. JADE (JADE, 2009) provides a framework for building autonomic management systems. A test bed was developed for modelling critical infrastructures for testing autonomic technologies (Autonomic Computing Lab; Cox, 2011).

However, there is a lack of progress in developing architectures to support applications before the full potential of autonomic computing for SCADA security can be realised. We propose to use the autonomic computing paradigm features to SCADA system security, in particular focussing on self-protecting SCADA systems. This paper incorporates autonomic computing paradigm elements to extend the SCADA architecture to safeguard against the emerging cyber security challenges and threats facing SCADA industrial applications.

In section 2 the relevant features of SCADA systems are described. Cognitive computing is discussed in section 3. Section 4 covers the autonomic computing paradigm. Section 5 proposes the architectural framework for SCADA cyber security and finally section 6 concludes the paper.

2. SCADA SYSTEMS VULNERABILITIES AND THREATS LANDSCAPE

SCADA systems were developed to be used as stand-alone systems which by their very nature made it difficult for an outside attacker to exploit the system. However, the many benefits associated with interconnecting the system to the Internet have transformed the SCADA systems into a highly interconnected system (Taveras, 2013; Nazir *et al*., 2017) accessible over the Internet (Fig 2). Therefore the protection offered by an unconnected SCADA system is not available anymore. The benefits are too

lucrative to be ignored by vendors and industry. Unfortunately it comes with an increased exposure to threats. The system interactions are complex, opening new threat entry points as there are many third party libraries and hardware assembled with components from around the world, with exploitable threats such as backdoors, often unknown to the SCADA system vendor.

The systems developers design customized solutions to address a particular problem. The systems are fairly long term deployments as the controlled processes have large financial and industrial outlays. The criticality of maintaining the process means that the systems remain in continuous operation and have a range of redundancies incorporated to protect stalling the system for foreseeable problems.

SCADA communications protocols such as Modbus, Distributed Network Protocol (DNP), IEC 870-5 and T103 are described by GE Communications Protocol. Most SCADA communications protocols have no encryption as they were designed when the SCADA systems existed only as stand-alone systems, rendering protocol authentication unnecessary. The Modbus protocol is one of the most common protocols for SCADA systems that operate on simple request-response messaging (Al Baalbaki *et al*., 2013). The diversity of the protocols and their inoperability also creates obstacles to design secure communications (Sheldon *et al*., 2004). There are many publicly available tools that can capture network traffic wirelessly. Also the wireless devices that feed data to the SCADA system provide easy entry points for the intruder into the system because the end devices do not have adequate protection, due to very low power requirements.

SCADA application vendors design their software to be hosted on generic operating systems such as Windows and Linux variants for widespread deployments; however, this makes SCADA applications exposed to the same vulnerabilities as that of the operating system. The long operational lifetime of SCADA software means that the host operating system may be beyond technical support. The features being added to the SCADA systems add further complexity and the systems become difficult to develop and maintain. Thus it becomes difficult to understand and restore systems to their operational state from a compromised state resulting from a cyber attack.

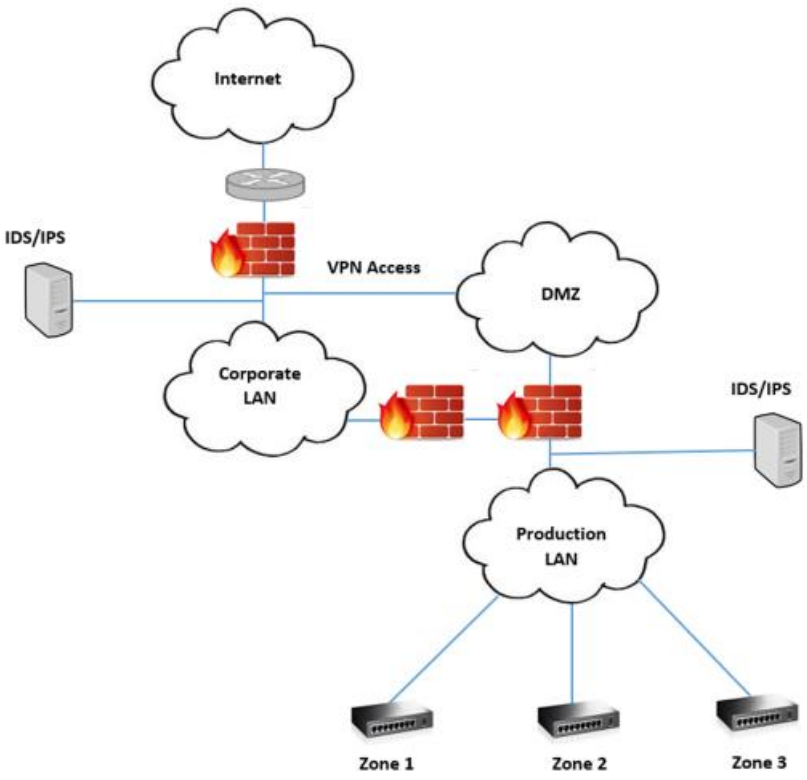

Fig. 2. Multiple pathways and Internat Connectivity to a Production System.

The cyber attack paradigms have progressed much beyond the simple attack methodologies such as man-in-the-middle (MITM) and Denial of Service (DOS) attacks (Chen and Abdelwahed, 2014), and are waged with increasing sophistication to hide detection. The traditional defence approaches are unable to cope with the latest attack methodologies where for example, the system parameters are altered, and are individually legitimate, but on the whole result in system collapse. Correct operation of the system needs not only the correct commands but commands that are consistent with the prevailing state of the system. It is possible for an attacker to inject a valid sequence of commands that gradually take the system to an unstable condition. The systems also operate under very tight timing constraints and can have undesired consequences in case of timing violation. Even the smallest intrusions on the critical infrastructure controls, can result in malfunctions which have devastating ripple effects on the system as a whole. The system is susceptible to attacks with minor effects, which can alter the system behaviour in a negative manner, leading to a ripple effect that compromises the whole system. The SCADA system entities are generally spread over a large geographical area, thus necessitating synchronisation of information at each location.

The threat landscape is rapidly evolving (Khadraoui and Feltus, 2015) and has gained momentum because the SCADA systems are now accessible over the Internet, and are no longer protected by obscurity as the communications protocols and characteristics are available to interested parties. Currently, both the state and non-state agents are trying to exploit the system's vulnerabilities. Cox (2011) discusses in detail threat ontologies.

In contrast to the attacks launched from outside, threats can also emanate from an innocent or deliberate mistake from an insider. Such attacks could cause more harm as they could be launched with some understanding about the system operation.

PROTECTION SCHEMES

Some recent technology adoptions and improvements in the SCADA systems are promising to aid developing systems that can result in an autonomic SCADA system. System protection can be ensured through many techniques. The majority depend on the judgement of a human to provide safeguards for the system.

The latest trends and innovations, such as virtualisation, analytics and databases, and wireless communications, which must work together in close collaboration to achieve the system mission, have been applied to SCADA systems. The integrated framework can rightly be called systems of systems as the complexity has increased beyond simple control and monitoring tasks, the fundamental basis of SCADA. This complexity implies that developing and maintaining such systems are reaching the limits of human cognition (Kephart and Chess, 2003; Huebscher and McCann, 2008).

System vendors have been cognisant of the prevailing cyber security environment and have added a number of features to the product offerings. These features include, for example multiplexing proxy, encryption and role based access to make the intruder's task difficult. Most SCADA vendors allow integration with relational databases in addition to the built-in historical databases that have some advantages (SQL). Relational databases such as Oracle have their own integrated analytics and data mining services that can make it easier to uncover any anomalous activity.

The machine learning and data analytics techniques have revolutionised many application domains and have recently been introduced in SCADA applications software. Such native integration makes it easier for the SCADA developers to analyse the systems operations and identify impending attacks (Kirsch *et al*., 2014; Carcano *et al.*, 2011). Machine learning and other such techniques can effectively analyse a system to detect anomalous activities. Such unsupervised anomaly detection schemes are more appropriate and efficient compared to human analysts (Jiang and Yasakethu, 2013) and other signature based approaches (Chen Abdelwahed, 2014). The system can thus learn new approaches and provide defence against as yet unseen scenarios, as in the case of supervised learning approaches. The other techniques of interest could be based on agent based, artificial intelligence, and adaptive systems (Greer and Rodriguez-Martinez, 2012). The future of cyber security lies with exploiting such techniques

that can not only autonomously assess the threats to the system security, but also contain and mitigate the threat from spreading, resulting in more damage. The operator alert can notify the human operator to initiate disaster recovery operations.

Virtualisation techniques provide many benefits that can advantageously be applied to support the autonomic computing paradigm. Virtualisation enables easy containment of an attack, restoring and disaster recovery, change and optimisation of system resources, etc., in a truly elastic manner.

A recent breakthrough in this direction is that of the Autonomic Computing paradigm. With Autonomic Computing, the ultimate control still rests with a human but the drudgery of data manipulation and threat assessment can be taken out of the loop.

3. COGNITIVE INFORMATICS AND COMPUTING

Cognitive Informatics is a broad and multidisciplinary field of cognition and information sciences that investigates the human information processing and its applicability for computing applications. A comprehensive review of the cognitive informatics framework is provided by Wang (2007a) and it also describes the applications from the fields of computing and software engineering. It uses Concept Algebra (CA), Real-Time System Algebra (RTPA) and System Algebra (SA) to formulate and represent knowledge using a formal notation. It can have diverse goals based on the application field but the overriding aim is to improve the human-machine interaction through better decision making. The hard problems in various engineering and scientific fields can be solved much easily if we knew the cognitive processes of the human brain (Wang, 2007a). For example, object recognition and classification problem in computer vision is hard for computers but comes naturally to humans, where a lot of progress has been made by mimicking the cognitive processes of the brain through Artificial Neural Networks (ANN). Similarly, the application of machine learning and agent based processing can help overcome the cyber threats facing the SCADA systems.

The theoretical framework for cognitive informatics and cognitive computing is presented by Wang *et al*. (2015) using a reductive model of the brain. It has been argued that the brain and natural intelligence can be explained through the reductive hierarchy at different levels.

The cognitive processes of formal inferences are described by Wang (2011b) cover both the applied and theoretical research processes using Real-Time Process Algebra (RTPA). It theorizes and demonstrates how the formal inferences in the human brain can be described using the cognitive processes of deduction, induction, abduction, and analogy. It provides a set of mathematical models and cognitive process for formal inference. This formalization of models is also helpful to design the intelligent computers based on Cognitive Computing (CC).

Cognitive computing comprises of intelligent computing methodologies to build autonomous systems that mimic the inference mechanisms of the human brain (Wang, 2009). Thus a system can detect anomalies, events and entities in a system through pattern recognition and data mining. These pro-active and self-learning systems can provide an effective defence against cyber threats, as signature based approaches can only work against known threats, It is also very important for critical infrastructure cyber security systems that the threat is anticipated and predicted before it strikes, otherwise it could be difficult to contain the resulting damage.

The future developments in the field of cognitive informatics have been described by Wang *et al*. (2011a; 2011c). The advances in the field of cognitive informatics have led to the development of cognitive computing. Computing can be classified at four levels in computation intelligence: data, information, knowledge, and intelligence (Wang *et al*., 2011c; 2015). Data and information processing have been well studied but the same has not been the case for the higher levels of computational intelligence are yet to be studied. This will foster an era of an intelligent revolution that will meet the human needs of wisdom and intelligence. Highly intelligent systems will be accessible to ordinary people to solve everyday problems (Wang *et al*., 2015). The recent trend of "Cognitive processes of the brain, particularly the perceptive cognitive processes, are the fundamental means for describing autonomic

computing systems, such as robots, software agent systems, and distributed intelligent networks." (Wang, 2007b).

4. AUTONOMIC COMPUTING PARADIGM

The roots of autonomic computing can be traced to the work by Norbett Wiener, John von Neumann, Alan Turing, and Claude E. Shannon on automata (Wang, 2007b). Autonomic computing leads to intelligent behaviours such as those driven through goals and inferences (Wang, 2007b). The theoretical and engineering foundations for autonomic computing together with a comprehensive set of theoretical foundations that is, cognitive informatics, behaviours, and intelligent science have been identified and the theorems for imperative and autonomic computing provide a solid foundation for the application of the field of autonomic computing to engineering applications (Wang, 2007b).

Autonomic Computing is one of the trans-disciplinary applications of Cognitive Informatics and an autonomic computing system using its intelligence can autonomously carry out its actions based on the set of events and goals (Wang, 2007a; 2007b). This contrasts with an imperative system whose behaviour is controlled by a stored program and is thus deterministic. The motivation for autonomic systems is to deal with the system complexity, which has reached an overwhelming proportion and is inspired by the human nervous system (Poslad, 2011).

The increase in system complexity and applications heterogeneity has made it difficult to process the information. This has necessitated the use of paradigms inspired by biological systems such as autonomic computing (Parashar and Hariri, 2005) that have a goal to realise systems and applications which operate autonomously based on high level rules to meet the system mission. It differs from Artificial Intelligence (AI) in that unlike those systems the ultimate decision may be taken by the human operator

The basic idea of the Autonomic Computing paradigm is that the system should be intelligent to enable it to develop and maintain itself in an optimised state. The human body's feedback and control mechanisms (Kephart and Chess, 2003; Parashar and Hariri, 2005) have formed the basis of general systems theory and holism for the development and management of computer based systems. The autonomic computing paradigm mimics the autonomic human nervous system. The ability to self-manage SCADA system security threats by developing learning systems that recognise vulnerabilities will be hugely advantageous. The agents and software services will form a part of the systems, gathering data and monitoring systems continuously (Yang *et al*., 2005).

Autonomic computing can result from the use of different technologies, however an autonomic system must demonstrate the following four main features: self-configuring; self-healing; self-optimising; and self-protecting (Ganek and Corbi, 2003):

1) Self-configuring: The system must be able to reconfigure its behaviour based on the changing system requirements. For example, to acquire more system resources, such as memory, in case the system is overburdened.

2) *Self-healing:* In response to detecting a compromised element in its configurtion, or lack of resources, an autonomic system can respond by repairing itself to a good state. Based on this assessment the system should be able to, for example, isolate the system components that have been compromised and continue operation with the remaining elements and at the same time attemping to restore the compromised system elements.

3) Self-optimising: The system must be able to assess the current state of the system variables and be abe to tune them to result in an optimised tuned behaviour. This is crucial as in the case of complex systems there are thousands of system parameters that can affect the system performance. Knowing or applying them all for best results is beyond the grasp of the human mind, in a resonable amount of time.

4) Self-protecting: The system should be aware of the normal system operation and be able to continuously monitor the current system state to determine when deviations occur. It can then take measures to contain the threat and to handle it

Autonomic computing facilitates identifying factors that relate to a specific state – homeostasis. The development of a knowledge network will help to identify what 'homeostasis' is and when there is an imbalance, to understand the structure of the network, the defences, the threats and the attacks. The threats can be classified into two categories: 1) process-related: when valid credentials are used to make legitimate changes that can impact on industrial processes. These can also be due to an error in the input of incorrect values or an actual attack (Crawford, 2006) by, for example, disgruntled employees; and 2) system-related: which are exploited via software or configuration vulnerabilities. For example, flaws in communication protocols, which are low level (layers 1 and 2) attacks on the SCADA architecture (Pidikiti *et al*., 2013). Developing a mechanism to mine logged data on process-related incidents is a potential solution to developing an autonomic computing approach for SCADA security. Identifying user activities and classifying the actions into signed-on or known user actions allows the analysis of threats as legitimate system commands by legitimate users, or by illegitimate users, to distinguish the threats into attacks or errors by developing a knowledge base (Hadžiosmanović *et al*., 2012).

The autonomic computing system incorporated to monitor a SCADA system may generate false alarms and therefore it may be necessary, based on the application domain, for a human operator to make a final decision based on the evidence.

5. ARCHITECTURAL FRAMEWORK FOR SCADA SECURITY

In this section we provide a brief overview of the architectures proposed in the research literature and propose a framework that can be used to design SCADA systems that have built-in layered protection against both known and unknown threats.

An autonomic system enables a SCADA system to optimise, configure and protect itself in case of changing the system state to a compromised one. The work to date for securing SCADA security focuses on discrete approaches. However, we propose an integrated approach that combines, the discrete knowledge based approaches with cognitive approaches. The memory layer of the Layered Reference Model of the Brain (LRMB) (layer 2), reflects the knowledge base that captures the short term, long term and transient memories. This can be utilised to capture process- and systems-related threats. Memory can be defined as a set of subconscious cognitive processes that retain the external or internal information about various SCADA security events. The subconscious knowledge base is inherited from the range of events and threats identified, and the conscious subsystem, however, is acquired and flexible, based on the autonomic computing paradigm (Wang *et al*., 2006a; Wang and Wang, 2006b).

Some autonomic architectures have been proposed in the research literature. The IBM autonomic computing system comprises, monitoring, analysing, planning, executing and a knowledge base component (Ebbers *et al*., 2006) and was proposed for large-scale commercial systems. The architecture utilises Touchpoint Autonomic Managers that are self-configuring, self-healing, self-optimizing and self-protecting.

An introduction to autonomic computing together with the challenges and opportunities are presented in Parashar and Hariri (2005). An Ultrastable system is discussed with reference to living organisms and human nervous system. The authors highlight the challenges in designing the general purpose systems that can address the emerging needs and complexity of services and applications. They propose architecture for an autonomic element as a smallest functional unit and propose a manager for each autonomic element.

Chen and Abdelwahed (2014) highlight the need for better security for the SCADA system and present an autonomic security model comprising of risk assessment, early warning and prevention, intrusion detection, and intrusion response. The signature based detection techniques can only be useful against known attacks whereas the anomaly based detection techniques have a high false alarm rate. Similarly demilitarised zones, access controls and firewalls do not provide adequate protection as with

time the attackers learn the vulnerabilities of the communication protocols and those of the operating system.

A detailed survey of autonomic computing models and applications is provided by Huebscher and McCann(2008). An Autonomic Critical Infrastructure Protection (ACIP) system using anomaly detection and autonomic computing is proposed by Al-Baalbaaki and Al-Nashif (2013). The modular system has online monitoring, feature selection and correlation, multi-level behaviour analysis, visualisation, and adaptive learning. The evaluation of ACIP is described using Modbus traffic generator for the Modbus traces between a server and five different PLCs. The proposed system could detect and stop a variety of attacks on the Modbus protocol (Al-Baalbaaki and Al-Nashif, 2013).

It was shown that by incorporating knowledge of a physical model of the system it was possible to identify the attacks through changes in system behaviour (Cardenas *et al*., 2011). The detection of attacks was formulated as anomaly-based intrusion detection. The results show that the response algorithm keeps the system in a safe state during an attack. Automatic response mechanisms were proposed on system state estimation. However, they caution that an automatic detection and response methodology might not be applicable for all processes in control systems.

A methodology for designing a smart critical architecture that protects communications, controls and computations using moving target defence and autonomic computing is proposed by Hariri *et al*. and also develop a Resilient Smart Critical Infrastructure Testbed (RSCIT). A general autonomic computing environment (Autonomia) was developed for control and management of smart critical infrastructures.

A survivable cyber-secure infrastructures (SCI) architecture is proposed by Sheldon *et al*. (2004) for a power grid and proposes a cognitive agent architecture combining agent-based and autonomic computing. Cognitive components are described as comprising of processes that are reactive, deliberate, or reflective.

In contrast to the architectures above, our proposed architecture combines three features to provide a threat-resilient SCADA framework: (i) virtualisation of computing and networking resources (ii) hierarchy of autonomic managers (AMs) to identify threats at different scales (iii) protection against false alarms.

Virtualisation refers to creating a virtual rather than physical version of computer hardware, storage and networks. The advantages are that the computing resources can be elastically assigned as required and it is much easier to monitor the virtual machines. In case of a cyber attack, a clean instance can be easily launched and the compromised machine can be isolated for forensics. Also, Disaster recovery and rollback can be performed easily. We propose hosting the SCADA system on a virtual platform. The advantages are that it can provide high availability through protection against hardware and software failures. Thus creating a broad generalised structure based on virtualisation wherein appropriate technologies can be selected to best suit an application within the given framework.

We propose the concept of hierarchical autonomic managers that can scale protection from a small to a wide area. A domain autonomic manager, AM_d performs real-time analysis of their limited domain (database, communications, etc.,) at a small scale. These domain-based analyses are then aggregated at the local system level, AM_l for identification of anomalies to counter the threats locally. This relieves the central autonomic manager, AM_c to take more holistic actions. Thus, a central autonomic manager can perform an analysis of system wide aggregated analysis to counter system wide variations to identify possible threats.

Thus, the inference of AM is based on the intelligent aggregation of the inferences of its lower level AM.

$$Inferences\ AM_c = \sum_{i=1}^{N} InferencesAM_l$$

We argue that despite the current state-of-the-art in autonomic computing applications, such as, machine learning and neural networks applied to SCADA systems, the ultimate decision should lie with the human operator. This is due to the criticality of the SCADA applications that might jeopardise the

safety and health of people, or compromise national security and infrastructures in case of false alarms. This of course, will vary from one application to another and a human decision-maker could be in the loop at some or all layers of AMs. The hierarchy of autonomic managers abstracts the information as it proceeds from low to high levels (domain to global) and can recommend actions to make it easier for a human operator to make a decision.

The structure and execution cycle of an AM is shown in Fig 3. It plans based on the given goals and rules, executes its plan which could be monitoring, comparison, infers the result of its execution to be an anomaly or a progression towards one, reports the inference to its higher AM. The knowledge base is analogous to the human nervous system storing structured and unstructured information used by the autonomic manager during its operation.

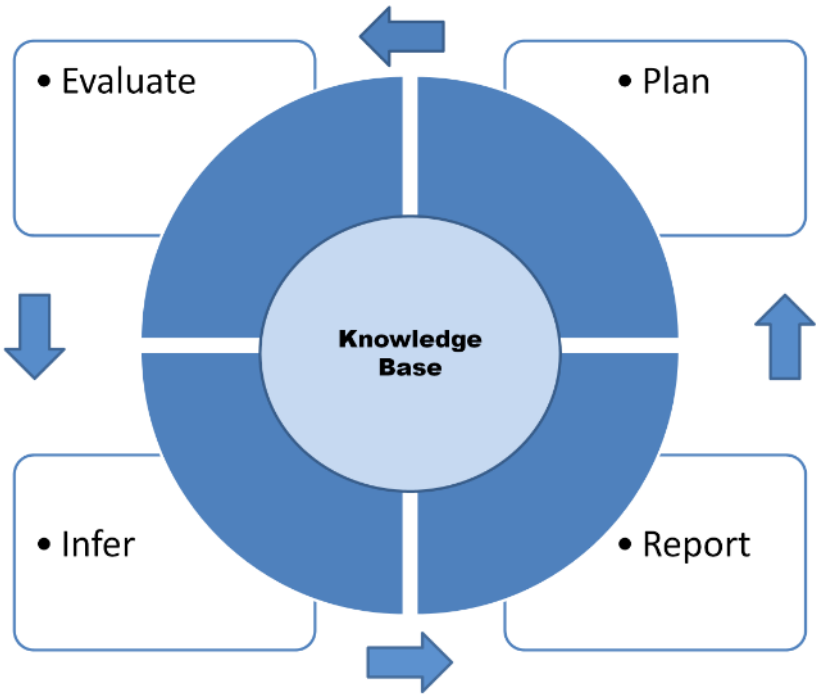

Fig 3. Structure and execution cycle of an autonomic manager.

The autonomic manager, as shown in Fig 3, can be used at various security layers of the system. The hierarchy helps to place the inferences at appropriate levels and the intelligence can travel up to the highest layer, that is, the central AM.

A SCADA system can have a large geographical spread, exposing it to exploitation at many locations, therefore necessitating an autonomic manager at each location that can monitor the security in the local areas and coordinate the efforts through the central manager. A simplified SCADA system architecture is shown in Figure 4. At the heart of the system is a central autonomic manager, that can enforce the broad threat mitigation and containment policies in the managed system as defined by the system administrator. The knowledge base provides the various historical system models that are continuously modified to the current state and are analysed to check conformance. The local autonomic managers continually observe the system state and act promptly in case of identified security threats to the local system.

Our proposed architecture provides a broad generalised structure based on virtualisation wherein appropriate technologies can be selected to best suit an application within the given framework. The identification of anomalies at an area level helps to counter the threats locally, relieving the central autonomic manager to take more holistic actions to counter system wide threats.

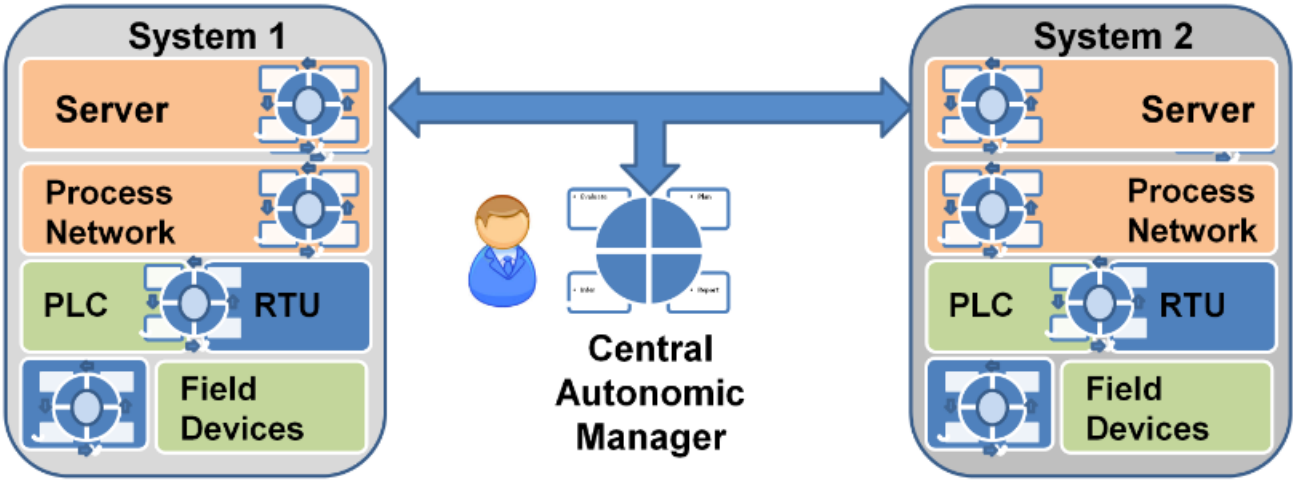

Fig 4. Proposed Architecture for an autonomic SCADA system.

It is also pertinent to point out here that the autonomic manager itself can be the target of a cyber attack. Such exploitation can be avoided through redundant deployments of managers and an integrated approach as proposed.

6. CONCLUSION

The evolving cyber threat landscape dictates changes to cyber defence approaches for the protection of SCADA systems. Unlike the traditional defence approaches where the response is governed by tailoring and monitoring according to threats, the concept of autonomic computing provides an advantage, as the systems are self-protecting. Thus, the cognitive and autonomic computing paradigms are very promising to develop SCADA system cyber security architectures that facilitate proactive threat mitigation methodologies. The autonomous nature enables flexible and scalable solutions across a wide range of SCADA system architectures and applications.

This paper provides an overview of the autonomic computing based architectures for SCADA security. We propose the concept of hierarchical autonomic managers that helps to extract, aggregate and refine intelligent inferences for ultimate decision making by a human operator. The proposed framework is generic and can be suitably applied across a range of real-world SCADA applications.

ACKNOWLEDGMENT

The research is sponsored by London South Bank University and Firstco Ltd., London, UK, through Innovate UK funding.

REFERENCES

Autonomic Computing Lab. http://acl.ece.arizona.edu/research.html [Accessed 20 August 2017].

Al Baalbaki, B., Al-Nashif, Y., Hariri, S. and Kelly, D., 2013, May. Autonomic critical infrastructure protection (acip) system. In *Computer Systems and Applications (AICCSA), 2013 ACS International Conference on* (pp. 1-4). IEEE.

Carcano, A., Coletta, A., Guglielmi, M., Masera, M., Fovino, I.N. and Trombetta, A., 2011. A multidimensional critical state analysis for detecting intrusions in SCADA systems. *IEEE Transactions on Industrial Informatics*, *7*(2), pp.179-186.

Cárdenas, A.A., Amin, S., Lin, Z.S., Huang, Y.L., Huang, C.Y. and Sastry, S., 2011, March. Attacks against process control systems: risk assessment, detection, and response. In *Proceedings of the 6th ACM symposium on information, computer and communications security* (pp. 355-366). ACM.

Chen, Q. and Abdelwahed, S., 2014, April. Towards realizing self-protecting SCADA systems. In *Proceedings of the 9th Annual Cyber and Information Security Research Conference* (pp. 105-108). ACM.

Cloud and Autonomic Computing Centre: https://sites.google.com/nsfcac.org/home [Accessed 20 August 2017].

Constantin L., 2014. "New Havex malware variants target industrial control system and SCADA users," *PC World*, Jun 2014.

Cox, D.P., 2011. The application of autonomic computing for the protection of industrial control systems. The University of Arizona.

Crawford, M., 2006. Utility hack led to security overhaul. Computerworld, 2006, pp.1-2.

Ebbers M., Byrne F., Adrados P. G., Martin R., and Veilleux J., 2006. Autonomic Computing (Chapter 8) IBM Introduction to the New Mainframe: Large-Scale Commercial Computing. ftp://public.dhe.ibm.com/systems/z/z_coursematerials/lscc/Large_Scale_Commercial_Computing_Student.pdf [Accessed 20 August 2017].

Fortes J., Parashar M., Hariri S., Banicescu I., 2014. Center for Cloud and Autonomic Computing (CAC). Compendium of Industry-Nominated NSF I/UCRC Technological Breakthroughs.

Ganek, A.G. and Corbi, T.A., 2003. The dawning of the autonomic computing era. IBM systems Journal, 42(1), pp.5-18.

GE, Communications Protocol, https://www.gegridsolutions.com/app/DownloadFile.aspx?prod=gesapm&type=8&file=7 [Accessed 20 August 2017].

Greer, M. and Rodriguez-Martinez, M., 2012. Autonomic computing drives innovation of energy smart grids. *Procedia Computer Science*, *12*, pp.314-319.

Hadžiosmanović, D., Bolzoni, D. and Hartel, P.H., 2012. A log mining approach for process monitoring in SCADA. *International Journal of Information Security*, pp.1-21.

Hariri S., Pacheco J., Tunc C., and Al-Nashif Y. ( ). A Methodolgy for Designing Resilient and Smart Critical Infrastructures. http://ecedha.org/docs/default-source/source/designing-resilient-and-smart-critical-infrastructures.pdf?sfvrsn=0 [Accessed 20 August 2017].

Huebscher, M.C. and McCann, J.A., 2008. A survey of autonomic computing—degrees, models, and applications. *ACM Computing Surveys (CSUR)*, *40*(3), p.7.

JADE - A framework for developing autonomic administration software. from http://raweb.inria.fr/rapportsactivite/RA2009/sardes/uid40.html [Accessed 20 August 2017].

Jiang, J. and Yasakethu, L., 2013, October. Anomaly detection via one class svm for protection of scada systems. In *Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), 2013 International Conference on* (pp. 82-88). IEEE.

Karnouskos, S., 2011, November. Stuxnet worm impact on industrial cyber-physical system security. In *IECON 2011-37th Annual Conference on IEEE Industrial Electronics Society* (pp. 4490-4494). IEEE.

Kephart, J.O. and Chess, D.M., 2003. The vision of autonomic computing. *Computer*, *36*(1), pp.41-50.

Kirsch, J., Goose, S., Amir, Y., Wei, D. and Skare, P., 2014. Survivable SCADA via intrusion-tolerant replication. *IEEE Transactions on Smart Grid*, *5*(1), pp.60-70.

Khadraoui, D. and Feltus, C., 2015. Designing Security Policies for Complex SCADA Systems Protection. *INFOCOMP 2015*, p.66.

Mahoney, W. and Gandhi, R.A., 2011. An integrated framework for control system simulation and regulatory compliance monitoring. *International Journal of Critical Infrastructure Protection*, *4*(1), pp.41-53.

Nazir, S., Patel, S. and Patel, D., 2017. Assessing and augmenting SCADA cyber security: A survey of techniques. *Computers & Security*, *70*, pp.436-454.

Parashar M. and Hariri S., 2005. Autonomic Computing: An Overview," in Unconventional Programming Paradigms. Lecture Notes in Computer Science, vol 3566. Springer, Berlin, Heidelberg.

Pidikiti, D.S., Kalluri, R., Kumar, R.S. and Bindhumadhava, B.S., 2013. SCADA communication protocols: vulnerabilities, attacks and possible mitigations. *CSI transactions on ICT*, *1*(2), pp.135-141.

Poslad, S., 2011. *Ubiquitous computing: smart devices, environments and interactions*. John Wiley & Sons.

Sheldon, F., Potok, T., Langston, M., Krings, A. and Oman, P., 2004, July. Autonomic approach to survivable cyber-secure infrastructures. In *IEEE Int. Conf. on Web Services (ICWS 2004), California, USA.*

SQL The Next Big Thing in SCADA, White Paper, Inductive Automation, 2012. https://www.automation.com/pdf_articles/inductive_automation/WhitePaper_SQL_The_Next_Big_Thing_in_SCADA.pdf [Accessed 20 August 2017].

Taveras, P., 2013. SCADA live forensics: real time data acquisition process to detect, prevent or evaluate critical situations. *European Scientific Journal, ESJ*, *9*(21).

Wang Y., Wang Y., Patel S.,and Patel D., 2006a. A layered reference model of the brain (LRMB),". *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 36(2), 124-133.

Wang Y. and Wang Y., 2006b. Cognitive informatics models of the brain" *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 36(2),* 203-207.

Wang Y., 2007a. The Theoretical Framework of Cognitive Informatics. *Int'l J. of Cognitive Informatics and Natural Intelligence,* 1(1), 1-27.

Wang Y., 2007b. Towards Theoretical Foundations of Autonomic Computing. *Int'l J. of Cognitive Informatics and Natural Intelligence,* 1(3), 1-16.

Wang Y., 2009. Cognitive Computing and machinable thought. In *2009 8th IEEE International Conference on Cognitive Informatics*, Kowloon, Hong Kong, pp. 6-8.

Wang Y., Widrow B., Zhang B., Kinser W., Sugawara K., Sun, F., Lu J., Lu J., Weise T., and Zhang D., 2011a. Perspectives on the Field of Cognitive Informatics and its Future Development. *Int'l J. of Cognitive Informatics and Natural Intelligence,* 5(1), 1-17.

Wang Y., 2011b. The Cognitive Processes of Formal Inferences. *Int'l J. of Cognitive Informatics and Natural Intelligence,* 1(4), 75-86.

Wang Y., Berwick R. C., Haykin S., Pedrycz W., Kinser W., Baciu G., Zhang D., Bhavsar V. C., and Gavrilova M., 2011c. Cognitive Informatics and Cognitive Computing in Year 10 and Beyond. *Int'l J. of Cognitive Informatics and Natural Intelligence,* 5(4), 1-21.

Wang Y., Rolls E. T., Howard N., Raskin V., Kinsner W., Murtagh F., Bhavsar V. C., Patel S., Patel D., and Shell D. F., 2015. Cognitive Informatics and Computational Intelligence: From Information Revolution to Intelligence Revolution, *International Journal of Software Science and Computational Intelligence (IJSSCI)*, 7(2), 50-69.

Yang L., Cao X., Gen X., and Zhang J., 2012. A Knowledge expression method of SCADA network attack and defence based on factor state space. *Journal of Theoretical and Applied Information Technology*, 46(2).