

# Secure Arrowhead Framework in the Local Cloud

Lidiya Raju<sup>1</sup> Jeena P Abraham<sup>2</sup>

<sup>1,2</sup>Assistant Professor

<sup>1,2</sup>Department of Computer Science and Engineering

<sup>1,2</sup>Mount Zion College of Engineering, Kadammanitta, Kerala.

**Abstract**— This paper presents a construct that supports security in an arrowhead framework. By providing the means of service oriented architecture, the arrowhead framework supports local cloud functionalities for automation applications. In order to provide secure communications in a local cloud, arrowhead themselves request the service as framework. This framework can have an impact on scalability also.

**Key words:** Cloud Computing, Local Cloud, and Arrow head

## I. INTRODUCTION

The Arrowhead Framework can be applied to multiple fields, like smart cities, industrial automation, smart grids, etc. The application of arrowhead architecture to such scenarios allows for higher flexibility, scalability [1]. The architecture of an Arrowhead-enabled system of systems is based on the thought of Local Cloud, which is a enclosed set of computational resources.

The local cloud also makes it achievable to make things easier monitoring and to reduce the managing complexity. It provided by the Arrowhead Framework and support Arrowhead applications by satisfying the non-functional requirements of the system.

Arrowhead frame work aims at support at various levels, using IoT technologies and a System of Systems approach. A particularly important non-functional requirement for automation systems is guarantee the Quality of Service required by the application services, which varies with the application [2]. Cloud based automation systems, choice of centralized or distributed control and data to information computations.

The paper focus on, new frame work arrowhead architecture provide scalability and security in the local cloud. An arrow head frame work mandatory core services are required. Capability to register a service to the local cloud and to discover which services are registered with the local cloud. Enabling loosely coupled data exchange between producer and consumer systems, service exchanges .Authenticate systems and Authorization service exchange.

## II. THE ARROWHEAD FRAMEWORK

The arrowhead frame work provide the features, are following:

**Performance:** The technologies are not developed for resource constrained devices and simplified to work properly on those devices.

**Scalability:** The addition of multiple things to the networks poses problems on the overall processing chain: sensor nodes, communication bandwidth and data processing.

**Security:** automation systems are usually composed of many resource constrained devices, which are not be capable of using complex algorithms for implementing security functions.

The Arrowhead Framework is dedicated to supporting local cloud automation functionalities by present a number of services. In Arrowhead frame work, which allow one element of the architecture to request information and actions from other elements of the architecture. Each element providing or consuming a service is called a system. Each physical or virtual platform providing computational resources in a local cloud is called a device.

The services offered in a local cloud include discovery of services, loosely coupled data exchange between producer and consumer services, security-related services. The Arrowhead Framework offers functionalities through the definition of Core Services, among which three services are mandatory and present in each Arrowhead local cloud: Service Discovery (SD), Authentication (AA) and Orchestration (O).

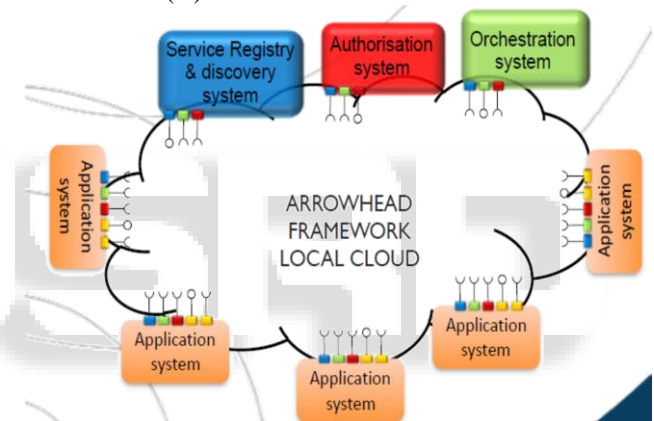


Fig. 1: Arrowhead Frame work Architecture

The ServiceDiscovery, which is offered by DeviceRegistry, SystemRegistry and ServiceRegistry systems, allows devices, systems and services to be registered in the local cloud. The Authentication is used to authenticate and provide authorization for connections between services. The Orchestration service is used to create the identical between service producers and service consumers, to allow service.

The process to allow service execution starts with the proper registration of devices, systems, and services. Each device declares which systems it is hosting, and to which other devices it is connected. Each system declares which services it produces and consumes.

The service orchestration responds to the declarative standard. The orchestration process is driven by the description of devices, systems and services in the Registry, and the Orchestration system is an engine that gets through the data, distils them into rules, and performs the matching between potential service producers and consumers.

## III. CONCLUSION

Thus the arrowhead framework architecture is implemented to improve security and scalability. This paper presented the

Arrowhead general approach, and went on describing how it can include services to support security requests for the interaction of consumers and producers. The Orchestration System of Arrowhead takes care of bargaining with the security manager system. Thus provide proper security and scalability in the local cloud by using arrowhead framework architecture.

#### REFERENCE

- [1] [Luis Lino Ferreira, Michele Albano "QoS-as-a-Service in the Local Cloud" , IEEE 2016
- [2] Ferreira, Luis Lino, et al. "Arrowhead compliant virtual market of energy", Emerging Technology and Factory Automation (ETFA), 2014 IEEE. IEEE, 2014.
- [3] Estevez-Ayres, P. Basanta-Val, M. Garcia-Valls, J. Fisteus, and L. Almeida, "QoS-aware real-time composition algorithms for service-based applications," IEEE Trans. Ind. Informatics, vol. 5, no. 3, pp. 278–288, Aug. 2009.
- [4] [4] Varga, Pál, and Csaba Hegedus. "Service Interaction through Gateways for Inter-Cloud Collaboration within the Arrowhead Framework." 5th IEEE WirelessVitaE, Hyderabad, India (2015).
- [5] Nichols, K., Blake, S., Baker, F. and D. Black, "Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers", RFC 2474, December 1998.
- [6] Object Management Group, Inc. (OMG), "Data Distribution Service for Real-Time Systems Specification", Version 1.1, December 2005
- [7] AMQP Advanced Message Queuing Protocol, Protocol Specification, Version 0-9-1, 13 November 2008,
- [8] P. Saint-Andre, K. Smith, and R. Troncon, "XMPP: The Definitive Guide", O'Reilly, 2009 .