The CHARIOT project

Konstantinos Loupos, Bora Caglayan, Alexandros Papageorgiou, Basile Starynkevitch, Franck Vedrine, Christos Skoufis, Stelios Christofi, Bill Karakostas, Antonis Mygiakis, George Theofilis, et al.

Cognition Enabled IoT Platform for Industrial IoT Safety, Security and Privacy – The CHARIOT Project

Konstantinos Loupos
INLECOM Innovation
Athens, Greece
konstantinos.loupos@inlecomsystems.com

Bora Caglayan
IBM Ireland Ltd
Ballsbridge, Ireland
Bora.Caglayan@ibm.com

Alexandros Papageorgiou
INLECOM Systems
London, UK
Alexandros.papageorgiou@inlecomsystems.com

Basile Starynkevitch, Franck Vedrine, CEA, LIST, Gif-sur-Yvette, France
Basile.Starynkevitch@cea.fr

Christos Skoufis, Stelios Christofi
EBOS Technologies Ltd
Nicosia, Cyprus
christoss@ebos.com.cy

Bill Karakostas
VLTN GCV
Antwerpen, Belgium
bill.karakostas@vltn.be

Antonis Mygiakis,
George Theofilis
CLMS Hellas
Athens, Greece
a.migiakis@clmsuk.com

Andrea Chiappetta
ASPISEC Srl
Rome, Italy
a.chiappetta@aspisec.com

Harris Avgoustidis, George Boulougouris
TELCOSERV
Metamorfosi, Greece
h.avg@telcoserv.gr

***Abstract* — The CHARIOT project is developing an innovative design method and cognitive computing platform that supports a unified and integrated approach towards Data Privacy, Security and Safety of Industrial IoT Systems including several technologies such as: an advanced Privacy and security protection method building on state of the art Public Key Infrastructure (PKI) technologies, a blockchain based ledger that provides categorization and affirmation of IoT physical, operational and functional changes through a combination of a cognitive engine and private keys, a FOG-based decentralized infrastructure for Firmware Security integrity checking leveraging Blockchain ledgers to enhance physical, operational and functional security of IoT systems, an IoT Safety Supervision Engine towards securing IoT data, devices and functionality in new and existing industry-specific safety critical systems, a cognitive System and Method with accompanying supervision, analytics and prediction models enabling high security and integrity of Industrials IoT as well as new methods and tools for static code analysis of IoT devices, resulting in more efficient secure and safer IoT software development. This publication provides a technical overview of the above technologies and methodologies as currently being integrated in the CHARIOT project.**

***Keywords*—Industrial IoT, Data Privacy, Security, Safety, Cognition.**

I. Introduction and Industrial Challenges

Cloud Computing and Internet of Things (IoT) technologies have been quickly advancing recently under concept of future internet, Industry 4.0 and related frameworks. IoT devices are designed following industrial requirements sometimes not considering recent risks that are associated to openness, scalability, interoperability and application independence, which is often the reason for new risks relating to information security and privacy, data protection and safety. Securing data, objects, networks and infrastructures inside any IoT system is expected to pose criticality on standardization activities of the next years. CHARIOT EC, research project, recognizes and replies to this challenge, identifying needs and risks and implementing a next generation cognitive IoT platform that can enable the creation of intelligent IoT applications with intelligent shielding and supervision of privacy, cyber-security and safety threats, as well as complement existing IoT systems in non-intrusive ways and yet help guarantee robust security by placing devices and hardware as the root of trust [1]. CHARIOT (Cognitive Heterogeneous Architecture for Industrial IoT) is an EC, research project under the IoT-03-2017 - R&I on IoT integration and platforms (www.chariot-project.eu).

II. CHARIOT Technological Overview

CHARIOT develops an integrated design method and cognitive computing platform supporting a unified approach towards Privacy, Security and Safety (PSS) of IoT Systems, that places devices and hardware at the root of trust, in turn contributing to high security and integrity of industrial IoT. CHARIOT specifies a Methodological Framework for the Design and Operation of Secure and Safe IoT Applications addressing System Safety as a cross cutting concern. CHARIOT design method bridges the systems engineering gaps that currently exists between the formal safety engineering techniques applied in the development and testing of safety critical systems and the rapidly evolving and ad-hoc manner in IoT devices are developed and deployed. CHARIOT develops an Open Cognitive IoT Architecture and Platform, exhibiting intelligent safety behavior in the diverse and complex ways in which the safety critical system and the IoT system will interact securely. CHARIOT also develops a runtime IoT Privacy, Security and Intelligent Safety Supervision Engine (IPSE) which will act continuously to understand and monitor the cyber-physical ecosystem made up of the IoT devices, safety critical systems and a PSS policy knowledge-base in real-time [2].

III. Cognitive IoT Platform and Dashboard Design

In parallel with the above technologies, CHARIOT develops the Cognitive System and related methodology as the accompanying supervision, analytics and prediction models enabling high security and integrity of Industrial IoT devices and systems. This includes the development of the intelligent behavior of the CHARIOT platform to enable IoT critical systems' secure interaction. In this direction a web-of-things environment is being developed interacting with heterogeneous IoT devices and systems. This includes the development of the suitable interfaces for the topological and functional behavior models for the IoT system devices (components) as well as additional safety and privacy structures that communicate through open APIs and include security services employing Blockchain, IoT security profiles and fog services.

This module will be mainly used to predict and avoid potentially endangering behavior in the IoT system that is now handled in an optimized way in cooperation with safety critical systems and run-time environments of the IoT system itself. This is using the existing IoT platform (IBM's Watson IoT) to demonstrate the CHARIOT innovative concept and capability and support integration with other safety, privacy and machine-learning cloud services via relevant open APIs, thus supporting third party integration and innovation.

At the same time, a security, privacy and safety awareness configurable dashboard is being developed in order to remotely monitor in near real-time the various IoT gateways and sensors in the CHARIOT IoT network. This will be the actual end-user interface to the CHARIOT solution and platform. The dashboard has been designed as part of the entire IPSE implementation (see next chapter) and will also be used for both understanding of the IoT ecosystem topology as well as for the post data analytical purposes to assist in the refinement and improvements of PSS policies.

This advanced intelligence dashboard utilizes the latest state-of-the-art web technologies in order to deliver rich content information to the LL users and achieve cross-browser and multi-device compatibility. Further to that, the dashboard follows rules regarding user friendliness, responsiveness and configurability as web solution, based on the LLs needs. The Dashboard supports various components that are expected to support and enhance the user experience as well as the actual system operation and configuration such as: interactive widgets, customizable entries, multiple components, data filtering etc.

IV. IoT Privacy, Security and Safety Supervision Engine (IPSE)

This component, as a major component of CHARIOT, is being developed as a set of novel runtime components which act in concert to understand and monitor the cyber-physical ecosystem made up of the IoT gateway and devices, the safety critical systems and safety/security policy knowledge-base. The components were designed by taking into consideration the operation and scalability requirements of the three living labs. IPSE can be scaled out by distributing the runtime across multiple nodes if needed.

The CHARIOT Privacy Engine employs current security protocols and recent technologies (e.g. Blockchain) in order to provide a strong foundation for the trusted interchange of information about and between the participants in the system-of-systems. The CHARIOT systems Safety Engine also analyses the IoT topology and signal metadata relative to the relevant safety profiles and applies closed-loop machine-learning techniques to detect safety violations and alert conditions. The primal objective of this engine is to develop a cognitive engine that will leverage the Cyber-Physical topological representation of the system-of-systems combined with the security and safety polices to provide a real-time risk mapping that will enable both static analysis and continuous monitoring to assess safety impact and appropriate response actions that will be in-turn communicated to the system operators and security teams for appropriate reactions.

The safety supervision engine is responsible for interacting with the CHARIOT IoT platform, providing a centralized intelligence and control functionality for applying the necessary privacy, security, and safety policies to all the components in the IoT system of systems, monitor IoT devices and systems to detect abnormalities in their behavior and analyze their causes, maintain an internal topological representation of the constantly evolving IoT system of systems and collect and represent privacy, security and safety (PSS) policies combining the threat intelligence in the topology in order to provide a real-time risk map, impact assessment and triggering of appropriate response actions. The safety supervision system can plug to multiple data sources near-real time and enforce machine learning based policy checkers such as anomaly detectors using anomaly detection models based on deep learning and expert rules.

The engine also maintains safety, security and privacy even when unknown (or not approved) devices and sensors are connected to the network, ensuring that they do not interfere to the normal operation of existing IoT components. This tool also assesses the topology to detect whether the IoT ecosystem has entered or is predicted to be advancing towards an abnormal (unsafe/insecure) state, and automatically activate a safety remediation in response to this unsafe state, to reduce the impacts on users and other IoT components and restrict abnormal operations and allow operations of safe functions to maintain at reduced level the operation of the controlled system.

The combination of physical and cyber components has put cyber-security in the limelight in the power industry. Communications, sensing as well as technologies of intelligent control are being applied in the field devices, bringing a change to the conventional structure of systems for power and changing the infrastructure of power into a more interactive, controllable and dynamic system. Because of that, the created smart grid environment enhances the probability of being attacked maliciously. Control and monitoring decision apparatus like protection relays based on micro-processors provide the easiest weak point for attacking to hackers [3].

The Security Engine service is a part of the IPSE functionality in CHARIOT platform responsible to check the integrity of the firmware of IoT gateways and devices. The Security Engine is liable to raise the guard level when any security related issue arises. Therefore, this service is able to detect tampered firmware which could contain potential threats by checking consolidated firmware and subsequent firmware updates to guarantee no security breaches.

V. Blockchain and PKI Technologies

Leveraging existing blockchain technologies along with traditional PKI schematics enables CHARIOT to revolutionize the field of identity management and operational security. Blockchain acts as the backbone of the system by establishing a trustless network of parties between the CHARIOT services, the IoT Gateways and the IoT Sensors. The implementation will be based on a permissioned blockchain that will become the mediator of any communications occurring within the network, acting as an overlay service that can be integrated by existing and future developed services via a straightforward and simplistic RESTful API. The smart contracts that will materialize within the blockchain network will facilitate the provision, amendment and revocation of identities within the trustless network formed by the blockchain. Each smart contract will be capable of employing a multi-signature protocol instead of a single-authority protocol. This will be enhancing the security of the system by requiring multiple administrator entities to conform to a blockchain modification.

The basic CHARIOT deployment will have a set of generic smart contracts that aim to cover the most common IoT network layouts. CHARIOT provides a basic smart contract solution that will be able to automatically and autonomously generate a fully functional smart contract for the CHARIOT blockchain. This will enable the CHARIOT blockchain security overlay to cover any and all types of network layouts. The identities stored on the blockchain network are cryptographically agnostic, meaning that any type of public cryptographic key can be assigned to an identity as long as it can be represented in a binary format. Those two characteristics of the CHARIOT blockchain ensure that it will be relevant for years to come and be able to adapt to new technologies due to its fluid structure and future-proof design.

VI. Static Code Analysis and Firmware Security Tool

As another significant constituent of the CHARIOT solution, CEA is developing an open-source software cross-compilation tool that can be used by IoT engineers designing IoT systems and developing device firmware in C or C++ source code running on them. This toolchain cross-compiles but also analyzes the source code with the same compiler and the same compilation options. For that, the analyzer is developed as a plugins/extension module for the GCC based compilers that the software industry is currently using and https://github.com/bstarynk/bismon, a new persistent system, to progressively record and assemble the static analysis results coming from the GCC plugins. bismon is an add-on upon standard compilation infrastructure to generate analysis results that are consistent with the compilation chain.

Compilation and analysis aim to be used during and will be executed at compilation/linking stage and will use meta-programming techniques to foster “declarative” high-level programming styles. This will enable the developers (as the IoT device firmware developers) to identify most safety critical functions executed at the IoT device or gateway level. Also, firmware compiled with that toolset will carry some cryptographic signature to enable filtering of firmware updates in the gateway. The bismon software is expected to be running, as a server process, on a Linux local-area-network server, in a trusted environment. It will provide a Web interface to drive the analyses on the source code. Thanks to its orthogonal persistence [4], bismon is restarted each morning and loads a textual representation of its persistent state, and is stopped every evening, dumping before an updated persistent state in the same textual format, which might be version-controlled by git. In that aspect bismon is behaving somehow like a NoSQL database, but keeping all the data in memory. Since all the data related to some IoT firmware source code of less than a million lines fits easily into the RAM (e.g. 128Gbytes) of a powerful Linux workstation.

To manage the consistency between the analyses and the need to compile with different GCC versions, the bismon system will generate GCC plugins in C++ form. These plugins are practically tied to a particular GCC version. A GCC plugin for gcc-7 usually won't work on gcc-8 without some porting efforts, so it is easier to generate, using metaprogramming techniques, the C++ code of the plugin from some higher-level representation, and rerun the plugin generation when upgrading the GCC compiler. In that sense bismon is a follow-up to the older GCC MELT project [5], which translated a Lisp-like domain specific language to GCC plugins in C++. To ease that generation, bismon has to process the plugin-related header files of GCC, in the spirit of CLASP[6].

Bismon aims to check that a given firmware source code does not overflow its call stack, since in firmware code the call stack is statically allocated and depends upon the particular RAM size and layout, and avoiding stack overflow which is a significant issue in IoT software. In practice, the GCC compiler does know the call frame size of every individual function. Of course, these frame sizes depend upon the particular optimization options (e.g. -O2 or -Os) passed to gcc or g++.

Improved compilation toolchain will generate some CHARIOT metadata embedded in the firmware binary and register them in the CHARIOT blockchain. That metadata includes cryptographic quality hashcodes of the source code, of the compiled binary, of any additional binary required by the firmware - such as some screen image, or a catalog of error messages, or a read-only table of coefficients related to the physical function of the IoT device.

VII. IoT Sensors and Connectivity

The IoT relies on a vast and heterogeneous set of sensors, each one providing specific functions accessible through its own dialect. There is thus the need for an abstraction layer capable of harmonizing the access to the different devices with a common language and procedure. Accordingly, there is the need for the sensor to introduce a wrapping layer, consisting of two main sub-layers: the interface and the communication sub-layers. The first one should provide a web-based interface, exposing the methods available through a standard interface and is responsible for the management of all the configuration aspects involved in the communication of the sensor with the external world. The second sub-layer should implement the logic behind the web service methods and translates these methods into a set of device-specific commands to communicate with real-world.

In CHARIOT, the reference Gateway used is the PANTHORA Gateway [7]. In CHARIOT, the connection to wireless sensors in order to be secure and safe, lead to the need of redesign the Gateway BLE and Wi Fi module in order to implement a protocol with private public key encryption scheme. The MRF24WN0MB module is designed to be used with Microchip's MPLAB® Harmony Integrated Software Framework. The Gateway uplink Northbound communication interface supports the MQTT protocol and can be integrated to open management systems. The supported communication technologies include IP-based wired Ethernet and wireless 2G/3G/4G and Wi-Fi to access the CHARIOT backbone.

VIII. System Deployment at Industrial Sites

The CHARIOT project provides a concrete deployment of an IIoT system including both hardware and software in a large range of industrial sectors, such as airports, rail infrastructure and smart buildings. Leveraging the existing infrastructure of sensors on each industrial sector and expanding it with “smart” IoT devices and technologies, incorporating a FOG-based decentralized enhancement, the CHARIOT project aims to provide a more efficient, secure and safer IoT operating environment. The system deployment performed in a real time operational environment in Trenitalia’s infrastructure, a Terminal building in the Athens International Airport and in aBuilding of IBM Campus in Ireland. In a rail environment, monitoring the data traffic between the train on-board IoT sensors, installed in the mechanic and electronic equipment of the train, and TRENTIALIA’s back bone system, enables the early detection of unauthorized IoT devices and anomalous data communication and in addition determines when a train maintenance is required. By monitoring the IoT devices and blockchain encrypted communication in building environments (IBM Business campus), CHARIOT project enables the IoT evolution to a cognitive IoT environment and provides a safer and efficiently managed working environment. Monitoring the IoT infrastructure and communications in an airport, CHARIOT project enhances the facilities protection on physical and cyber threats by an early detection and prediction of hazardous situations and reduction of the false positive alarms.

Conclusion

The CHARIOT project, at the time of preparation of this publication has completed its first 18 months of execution and is now into the second year, where system integration, validation and deployment to the actual living labs are expected to take place following the project plan. In an iterative approach, this validation phase will drive further technological developments and research efforts following the project agile approach to provide feedback from the end-users to the development efforts. CHARIOT is currently in an integration phase where the FOG servers’ connectivity is being defined to ensure the components interfacing and communication through the CHARIOT platform in each pilot.

Acknowledgments

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 780075”. The authors acknowledge the research outcomes of this publication belonging to the CHARIOT consortium.

References

[1] Adel S. Elmaghraby, Michael M. Losavio, “Cyber security challenges in Smart Cities: Safety, security and privacy”, Journal of Advanced Research Volume 5, Issue 4, Pages 491–497, July 2014.

[2] CHARIOT Grant Agreement number 780075, Annex 1.

[3] Konstantinou, C., & Maniatakos, M. (2015, November). Impact of firmware modification attacks on power systems field devices. In Smart Grid Communications (SmartGridComm), 2015 IEEE International Conference on (pp. 283-288), IEEE.

[4] Alan Dearle, Graham N. C. Kirby, Ronald Morrison, “Orthogonal Persistence Revisited”, ICOODB 2009, LNCS 5936.

[5] Basile Starynkevitch, “MELT - a Translated Domain Specific Language Embedded in the GCC Compiler”, DSL 2011 IFIP conf., Bordeaux, France.

[6] Christian E. Schafmeister, CLASP - A Common Lisp that Interoperates with C++ and Uses the LLVM Backend, Proceedings of the 8th European Lisp Symposium on European Lisp Symposium, ELS 2015

[7] PANTHORA - http://telcoserv.gr/portfolio-item/panthora-rms