A Cyber-Physical Approach to Resilience and Robustness by Design

Giovanni Di Orio[1], Guilherme Brito[2], Pedro Maló[3]
NOVA School of Science and Technology – FCT NOVA
UNINOVA - Instituto de Desenvolvimento
de Novas Tecnologias

Abhinav Sadu[4], Nikolaus Wirtz[5], Antonello Monti[6]
RWTH Aachen University
Institute for Automation of Complex Power Systems

***Abstract*—Modern critical infrastructures (e.g. Critical Energy Infrastructures) are increasingly evolving into complex and distributed networks of Cyber-Physical Systems. Although the cyber systems provide great flexibility in the operation of critical infrastructure, it also introduces additional security threats that need to be properly addressed during the design and development phase. In this landscape, resilience and robustness by design are becoming fundamental requirements. In order to achieve that, new approaches and technological solutions have to be developed that guarantee i) the fast incident/attack detection; and ii) the adoption of proper mitigation strategies that ensure the continuity of service from the infrastructure. The "Double Virtualization" emerged recently as a potential strategy/approach to ensure the robust and resilient design and management of critical energy infrastructures based on Cyber-Physical Systems. The presented approach exploits the separation of the virtual capabilities/functionalities of a device from the physical system and/or platform used to run/execute them while allowing to dynamically (re-) configure the system in the presence of predicted and unpredicted incidents/accidents. Internet-based technologies are used for developing and deploying the envisioned approach.**

***Keywords*—Double virtualization; critical energy infrastructures; cyber-physical systems; resilience**

I. Introduction

The evolution of critical infrastructures into complex distributed networks of Cyber-Physical Systems (CPSs) has posed several challenges on how to monitor and control these systems [1]. The physical dimension of hardware components, and the cyber dimension of computations and communications are both susceptible to attacks that could potentially bring down the entire system [2]. This is particularly true in the Critical Energy Infrastructure (CEI) domain, characterized by vast, dispersed and heterogenous infrastructure of assets forming a multifaceted operational environment.

To address these challenges, i.e. to facilitate the monitoring and control of this kind of infrastructures, smart grid concept has evolved. The smart grid deeply relies on the usage of communication and information technologies to enhance the control and monitoring of the grid, while providing a better "awareness" about the state of the grid [3], [4]. As stated in[3], smart grid incorporates several technical initiatives such as Advanced Metering Infrastructure (AMI), Wide-Area Monitoring, Protection and Control (WAMPAC) systems based on Phasor Measurement Units (PMUs) that are aiming to provide the guidelines and guidance on how to collect, transport, use and present data generated by the grid assets. Since these initiatives heavily rely on Information & Communication Technology (ICT) systems and, they are exposing the smart grid to a wide range of security threats and more in general to vulnerabilities that need to be managed to keep the system secure [5]. As a matter of fact, the proliferation of smart devices (and exploitation of cyber advances) can practically enable anyone to gain access and interact with the smart grid supporting infrastructure. As stated in [3], cyber-attacks can take many forms, depending on their objective and goal, while being distributed in location. All these aspects together make it nearly impossible to design and develop a "one-size fits all" approach that guarantees the security for every asset within the infrastructure.

With this in mind, the main purpose of this paper is to present a specific strategy, approach and technological development – the so called "Double Virtualization" (DV) – to enable resilience and robustness of WAMPAC against cyber and physical attacks. Typically, system level control and monitoring functions of a smart grid are deployed in dedicated computational units. Any cyber-physical attacks on these dedicated computational units would compromise the complete operation of the smart grids. To deal with this situation and to minimize the effects of cyber-physical attacks a strategy is proposed that is built on top of cloud computing paradigm and – thus – based on the principle that monitoring and control capabilities/functionalities are logically separated from the hosting computational hardware and/or platform. In such a scenario, it will be possible to dynamically allocate/relocate virtual functionalities/capabilities to other similar computational hardware and/or platform under cyber-physical attacks. The proposed strategy does not contemplate the avoidance of cyber and physical attacks, on the contrary, it focus on their early detection and on defining the mechanisms that ensure continuous operation of a CEI (like the smart grid), by increasing the availability of the control and monitoring functions of the CEI that are re allocated in different hardware under any cyber-physical attacks.

II. Related Works and Supporting Concepts

A. Cyber-Physical Systems and Smart Grids

Nowadays, the conventional systems and processes – in the most disparate context of application e.g. manufacturing, healthcare, automotive, smart grids, logistics etc. and different nature e.g. mechanical, electrical, and chemical – are evolving

into CPS. As stated in [6], the term "Cyber-Physical Systems" has been coined in 2006. Today, several definitions of CPS can be found in the literature. According to [7], CPS can be defined as transformative technologies that allow the management of physical assets and computational capabilities of interconnected systems. The definitions in [8], [9], highlight the concept of collaboration and service provisioning. As a matter of fact, CPS are defined as systems of collaborating computational entities that are strictly connected to the surrounding physical assets providing and using services to/from the internet. A working definition for CPS has been offered in [10], where a CPS is defined as a system consisting of computational, communication and control components combined with physical processes. Nowadays, CPSs are the foundation and the key element for smart grids. As a matter of fact, the two major elements of a smart grid are: the supporting infrastructure and the power application. In particular, the former is the one that delivers "smartness" to the grid and concerns with the integration of new technologies (cyber advances) and approaches for enhancing the monitoring and control activities of the operations within the grid. Therefore, Smart grids are opted as the application domain of the presented research, where securing are provisioning them with innovative mechanisms for responding to cyber physical attacks are actually the main objectives.

B. Cyber-Physical Systems and Industry

The research stream on CPS is extremely active and vibrant in the manufacturing domain as confirmed by the number of research activities on the topic. As a matter of fact, there is an extensive literature dealing with the materialization of the CPS vision and related challenges – technical, societal and educational – as confirmed in [11]. Modern production systems and their related control and monitoring solutions can be easily modelled as a network of interconnected and collaborative CPSs where communication takes place constantly both horizontally and vertically. However, the classical heterogeneity in equipment, encompassing distinct functions, form factors, network interfaces and I/O features supported by dissimilar software and hardware platforms is pushing for a new and well-defined strategy to increase the devices interoperability and agility performance [12]. It is necessary to comprehend that today's problem is no longer networking (protocols, connectivity, etc.) nor it is hardware (CPU/memory power is already there, at low-cost and low-power consumption) but rather it is on how to link disparate heterogeneous data sources to the specific needs and interaction forms of applications and platforms. In this scenario, the abstraction and/or virtualization of physical entities in terms of their functionalities – provided as services available over the network – is a necessary condition to ensure the creation of a highly dynamic and evolvable environment while detaching functionalities from the specific runtime, protocols and communication needs as confirmed in [13]–[16]. Furthermore, industrial initiatives such as the Reference Architectural Model for Industry 4.0 (RAMI 4.0) confirms the trend. In particular the Asset Administration Shell concept establishes the guidelines and methodology for industry digitization, i.e. for integrating industrial assets into I4.0 communication backbone [17], [18]. Finally, the research performed by the authors in the manufacturing domain provided the foundation for the design and development of the proposed DV strategy.

C. NIST Framework for Critical Infrastructure Cybersecurity

The NIST Framework for Critical Infrastructure Cybersecurity [19] has been developed to deliver a systematic approach for managing cybersecurity-related risk that is aligned with the typical requirements of critical infrastructure providers [20]. The framework is built around five core concurrent and continuous functions, namely: i) Identify, ii) Protect, iii) Detect, iv) Respond, and v) Recover. These functions together allow organizations to express, in a high-level strategic view, its management of the cybersecurity risk [19]. The NIST framework core elements provided the foundation for framing the application context, developing and implementing the necessary protection and detection mechanisms, as well as, the mitigation actions and recover strategies for robustness and resilience.

D. Wide-Area Monitoring, Protection and Control

The WAMPAC leverages the PMUs and Phasor Data Concentrators (PDCs) to gain real-time awareness of the current state of the smart grid supporting infrastructure and related operations [21]. The WAMPAC can be further divided into three main components, namely: Wide-Area Monitoring Systems (WAMSs), Wide-Area Protection Systems (WAPs), and Wide-Area Control (WAC) [3]. The WAMPAC system provided the environment for the deployment of "Double Virtualization" services. In particular, some services have been developed [22] and later clustered under the name "Double Virtualization" and deployed in already existent WAMPAC to enhance the smart grid supporting infrastructure availability.

E. The Observe-Orient-Decide and Act Pattern

The Observe-Orient-Decide-Act (OODA) pattern (see Fig. 1), introduced by John Boyd[23], is a multi-staged approach to facilitate and speed-up the decision-making process. According to this pattern, the decision-making process occurs in a recurring cycle of four core stages, namely, the observe, orient, decide and act. The main objective is to deliver highly reactive and responsive systems that are capable to continuously adapt and evolve to changing and/or unpredictable circumstances. The OODA cycle has provided the foundation for the design of the overall DV strategy and related processes. In particular, the "Double Virtualization" strategy has been redefined and distilled around this cycle to encompass the OODA four interacting processes:

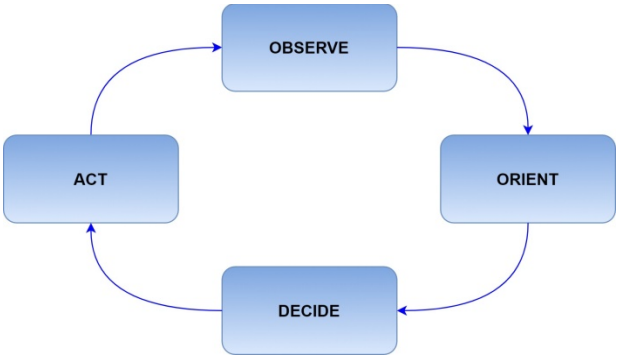

Fig. 1. OODA Simple Loop.

- *Observe:* the stage where related sensorial data is gathered. Data is received from a variety of sources at each moment, which in DV translates to data incoming from all the connected devices.
- *Orient:* DV handles the data to enrich it with meaning that is needed for further inspection: match the data to the respective device, comparison to previous stages, apply inspection algorithms and so on, in order to find meaningful flaws or deviations in the most efficient way. The observations made in this step shape the decisions and actions of the next iterations of the loop.
- *Decide:* the input from the current orientation supplies the model with the possible paths and concludes for the most suitable to deliver (similar to a hypothesis, as referred by Boyd), therefore forming the plan to take into the next step.
- *Act:* The DV system then executes the formerly defined plan, while maintaining track of the advances made and sending information back to observation, thus restarting the loop.

F. Previous Research

Traditionally the power grids have been operated using a centralized automation architecture where in the primary monitoring, control and protection algorithms run at a central server. The SCADA and WAMPAC systems provide the algorithms with the real time measurements and facilitate with the automation systems for real time control. Since the major intelligence for operation of the grid is deployed in a single server, it is always important to ensure continuous availability of the intelligence (algorithms) even under cyber-physical attacks on the device hosting them. Therefore in SUCCESS[1] [24] the concept of Double Virtualization has been introduced. Here DV is designed for virtualizing the monitoring, control and protection algorithms. Furthermore, these virtualized algorithms were moved from one device to another when a specific device hosting the algorithms failed under cyber-physical attacks. A proof of concept implementation was then done based on CALVIN[2] Internet-of-Things (IoT) platform and presented in [22]. The study showed that CALVIN was a suitable platform for DV and that with DV the availability of the key automation functions was improved. However, usage of CALVIN confined the implementation of such automation functions to its own language, thus demanding considerable time to learn how to integrate them in its framework. Moreover, CALVIN does not provide an easy way to execute and control external computational processes, such as scripts or even the simple execution of terminal emulators, and it was found that CALVIN lacks on offering flexibility, security and scalability to larger systems and heavier functions.

III. System Model and Problem Formulation

A. WAMPAC Architecture

Fig. 2 shows a typical hierarchical WAMPAC architecture and communication layout together with the distinct computer layers ("cloud", "fog", "edge" and "device"). The state of the system is measured by PMUs. PMU measurements are collected and communicated to devices within the upper hierarchical levels of the architecture – called Phasor Data Concentrators (PDCs) –through high speed communication links (based on the IEEE C37.118.2 [25] standard for synchrophasor data transfer for power systems) to produce a real-time, time-aligned output data stream. PDCs can exchange phasor data with others PDCs. Finally, collected data are communicated to the WAMPAC control center where several potential applications are executed such as state estimation, model validation, early warning systems, etc.

The networked and wide distribution nature of the WAMPAC architecture opens the doors to several vulnerabilities and/or possibility for cyber-attacks that can potentially affect the normal functioning of the system. It is necessary to design, develop and implement appropriate countermeasures to ensure the early detection and localization of those attacks while minimizing their impact on the system.

B. Cyber Attack Classification, Vulnerabilities and Entry Points in WAMPAC

The cyber-physical nature of a WAMPAC-based system implies that cyber-attacks can be easily directed to both power/physical and communication resources. Taking into account the WAMPAC environment, there are several relevant threats where the "Double Virtualization" approach/strategy can be applied as countermeasure, and that can be clustered according to [3] into three main groups, namely: i) Time-based attacks; ii) Integrity attacks; and iii) Reply attacks. However, in the context of the present research only time-based attacks have been considered, i.e. detectors have been implemented and "Double Virtualization" has been applied as a special measure to enhance availability and resilience of grid monitoring and control applications.

In a time-based attack, the attacker tries to compromise the normal conditions and/or operations of the system by making a device, resource or service unavailable over the network. An example of this is the Denial of Service (DoS) attack.

The DoS exploits communication vulnerabilities of CPS to inject false data with the objective of over-flooding the network. Furthermore, devices themselves are vulnerable to physical damage by malicious and/or accidental causes that can lead to the loss of all the critical data provided by that channel.

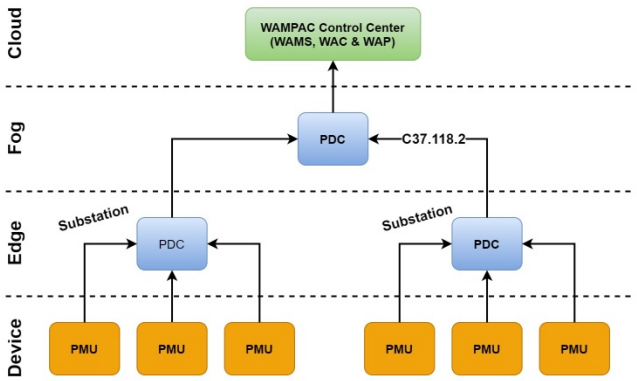

Fig. 2. Typical WAMPAC Architecture.

[1] https://success-energy.eu/
[2] https://www.github.com/EricssonResearch/calvin-base

As stated in [26], for each identified threat it is possible to model and identify an attack surface (see Fig. 3) with related multiple entry points. The attack surface allows to highlight the penetrable boundaries – i.e. the boundaries that an attacker can use to connect with – of the system under study as well as the internal path to critical resources. In the case of CPS entry points translate to: i) physical connections like cables, power sockets, etc., and ii) cyber connections like global accessible APIs, communication sockets, open ports, etc. These are also the boundaries that have been considered in the present work (see the yellow boxes in Fig. 3) as possible vulnerabilities and entry points that need to be properly handled by providing a strategy that allows monitoring and control functionalities to remain available even if devices are compromised in both physically and cyber dimensions.

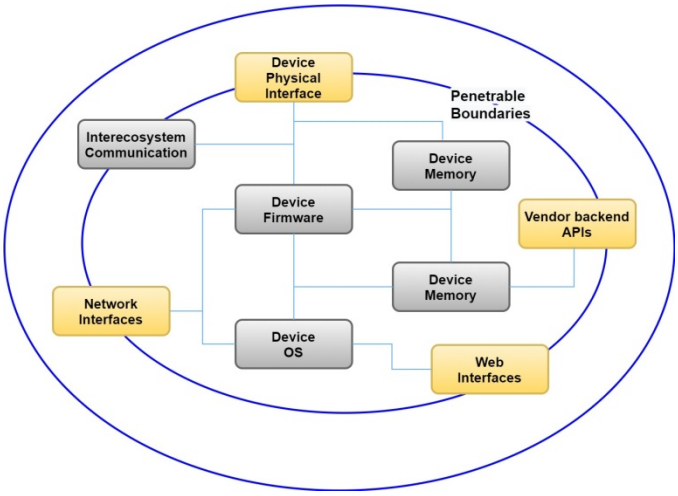

Fig. 3. Attack Surface, and Considered Entry Points, Adapted from [26].

IV. Implementation of Double Virtualization

A. Necessary Technologies for Double Virtualization

1) Node-RED: The implementation of DV presented in this publication is supported by Node-RED [3] framework. Node-RED is a flow-based programming tool oriented to development of IoT applications, which provides an easy-to-use browser-based graphical editor. Node-RED supplies a wide set of core Nodes, and possesses a big and highly active community that contributes with free-of-charge custom nodes.

Furthermore, the panoply of exisitng features and nodes on Node-RED enable it to easily access terminal emulators, and therefore execute and manage running processes of the host machine. On the other hand, it also provides the development environment to establish connections to the running applications and processes, while also enabling their functionality extension as part of the embedded logic. This is one of the main reasons why this technology has been opted as the development and deployment framework for DV. Altogether, Node-RED has the potential to offer connectivity of several devices, by creating APIs and services supported on crossover environenments, using varied protocols and which can be easily deployed in its runtime.

2) PM2: PM2[4] is an advanced process manager oriented to Node.js applications, however, it has also the capability of managing other types of processes such as Bash or Python scripts and can be easily invoked by either the command line prompt or by embedding in any Node.js application such as Node-RED.

B. Double Virtualization Architecture

To integrate DV functionality and logic within an ordinary WAMPAC architecture and configuration, several components and related functionalities and interfaces have been designed and implemented. These functionalities have been deployed using the existing WAMPAC infrastructure, i.e. WAMPAC device platform, to equip current devices with the necessary logic for DV and to create two type of devices, namely:

1) DV asset device: devices where the application layer, i.e. all the application running on the them, is/are virtualized in order to be easily managed by the DV components and logic (see Fig. 4); and.

2) DV administration and management (DVA and M) device (see Fig. 5): devices that are logically settled at a higher level than the DV Asset devices and that are intended to run, manage and initiate the whole DV process by extracting, collecting and processing data provided by the DV Asset devices, in real time, as well as to execute the necessary actions according to the result of the data processing task.

Both *DV Asset* and *DVA&M* devices are necessary for *DV*, i.e. the *DV* process and functionalities emerge as the result of the communication and interaction between these two types of devices.

The research presented in this paper is intended to spotlight the *DV* concept and functionality by providing an example of application where *DV Asset* and *DVA&M* devices are used to deliver to WAMPAC system the capability of:

1) detecting failures that compromise the availability of *DV Asset* devices; and

2) mitigating detected failures to minimize their impact on the system.

According to these objectives the architecture and system configuration presented in Fig. 6 have been implemented where a single *DVA&M* device and several *DV Asset* devices are employed to provide the necessary results.

C. DV Asset device

The *DV Asset* devices provide the following functionalities and/or services:

- *Virtualization service*: it includes all the necessary mechanisms to enable the virtualization of device applications. Thus, it is responsible to create an abstraction layer that allows to separate device hosted applications from the specific hardware architecture while allowing these applications to be moved easily moved and interpreted by other *DV Asset* devices.

[3] https://nodered.org/

[4] https://pm2.keymetrics.io/

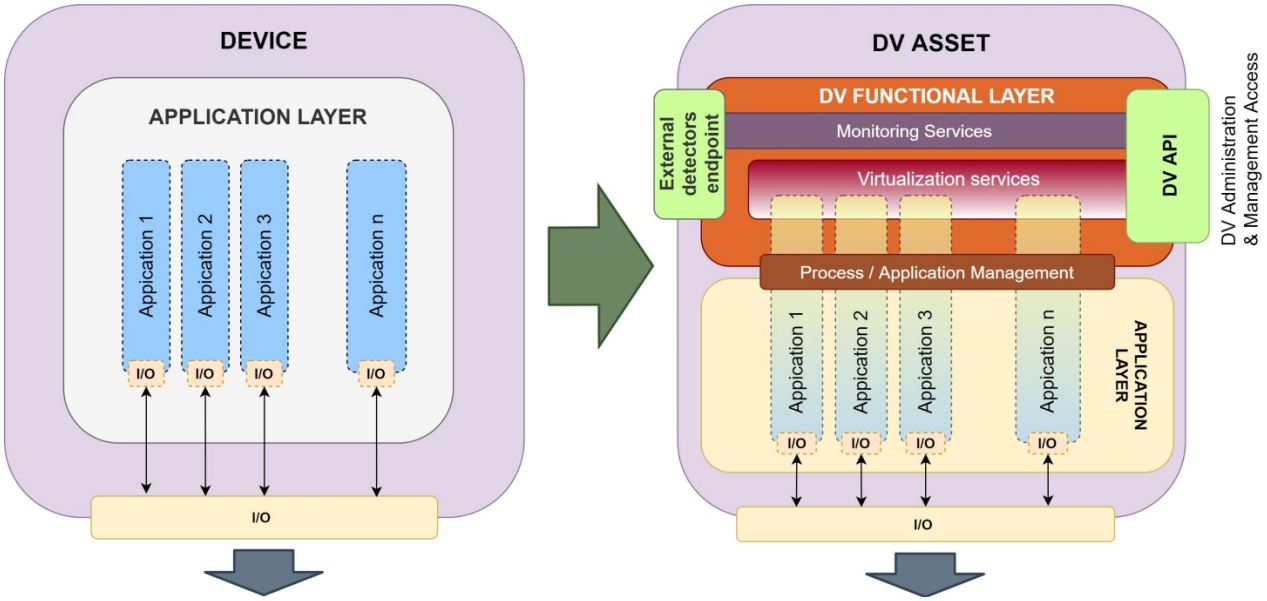

Fig. 4. Computational Device "Enhanced" as DV Asset Device.

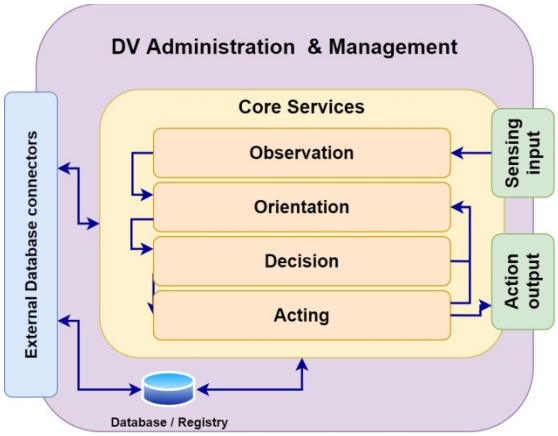

Fig. 5. DV Administration and Management Device.

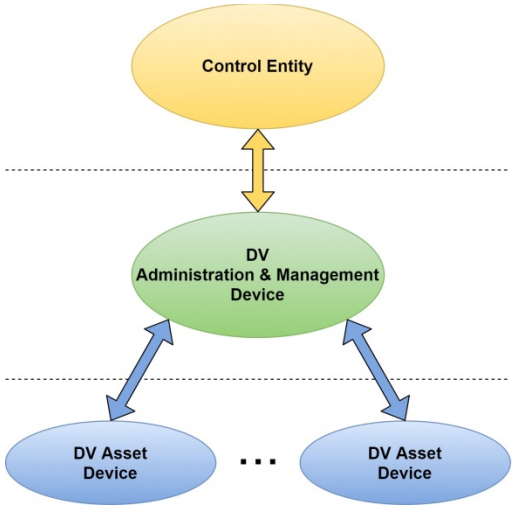

Fig. 6. DV Architecture with a Single DVA and M Device.

- *DV API*: it is an endpoint specifically created to allow *DVA&M* devices to remotely access the *DV Asset* devices for extracting and collecting the necessary information as well as for sending commands for triggering specific *DV* routines and actions (e.g. start, stop, reconfigure and/or move virtualized functions from one *DV Asset* to another).
- *Monitoring service*: it provides the ability to the *DV Asset* device gather all the necessary information about the resources attached to it and related functionalities. Some gathered information include: network connection status, CPU and memory usage, as well as any change in the virtualized functions. As part of the Monitoring service there are: *Heartbeat Service* and *Ping Service*. Furthermore, Simple Network Management Protocol (SNMP) services could also be included to enable deeper monitoring capability of the *DV Asset* device.

D. DV Administration and Management device

The *DVA&M* device is then structured to comply with the OODA loop as a way to fulfill the goals proposed by the DV, and adopting the "enhanced" system with the Control loop depicted in Fig. 7. So, the *DVA&M* devices are provided with the following functionalities and/or services:

- *Persistence Service*: it is a necessary condition for DV to have some type of database system for supporting registry functionality (e.g. register available *DV Asset* devices), as well as, storing temporary data necessary to run the *DV* logic. This service can be as simple as a file system or a more complex DB framework, either locally or remotely:
- *DVA&M Core*: combination of the DV resources and functionalities that constantly uses the available set of information provided by the *DV Asset* devices and any other external data source if needed. The core features of the *DVA&M* have been divided in four distinct stages, namely:

1) Observing: The DV acquires data from connected devices installed on the sub-station through several pre-defined connectors. The data can be obtained by using request/reply (e.g. Heartbeat, as used in the current implementation) or publish/subscription, or through the integration of connectors which may be implemented to cope with several protocols, as for example, REST endpoints. Also, the sources of data are mainly composed by the *DV Asset* devices working in the subsystem under the supervision of the *DVA&M* device, however, it has also been implemented connectors that enabled the integration of external detectors, which by their turn, also send data related to *DV Assets* devices.

2) Orienting: In this stage, each item of the received data is inspected through certain mechanisms, in order to be contextualized in respect to the system, so that possible scenarios can be envisioned. This means that for each incoming message, the content is deciphered in order to identify the source device and the type of information, compare it with previous stage, look for mismatching or unexpected values, and so on. This enables to identify if and which algorithms of the DV can possibly make use of the information to take some action. Through the help of dedicated observation services, sometimes it is easy for the DV to identify the type and source of the data. For example, receiving successful or failed heartbeat messages on dedicated HTTP(s) request will rapidly lead the DV to change the possible future scenario to take in consideration.

3) Decision: The data and information provided by the *Orientation* stage are used during this stage by the *DVA&M* device to decide about DV actions, i.e. decisions computed during this stage are then translated into actions to be executed in the *Acting* stage. Moreover, decisions are made through algorithms that possess a configurable set of rules, limits and thresholds. Considering the former heartbeat example, depending on the configuration in use in the system, one failure message may be enough to activate a mitigation action, while on other system is may be needed two consecutive heartbeat failure messages. In the end of the stage, the possible decisions are: “no action”, or “start migration” to trigger the execution of a mitigation action that involves at least two *DV Asset* devices. The actors of the decision are also established during this stage.

4) Acting: It is the practical execution of the decisions of the *Decision* stage, i.e. the decisions taken by the *DVA&M* device are translated into an action to take effect on *DV Asset* devices, if necessary. This is accomplished by establishing dedicated channels, which were designed as part of the DV communication and management features, for sending the necessary information and control data required by the involved *DV Asset* devices to execute the actions. Moreover, while actions are being deployed, the *DVA&M* device continuously observes the status of the operations while providing this information as internal feedback to the loop.

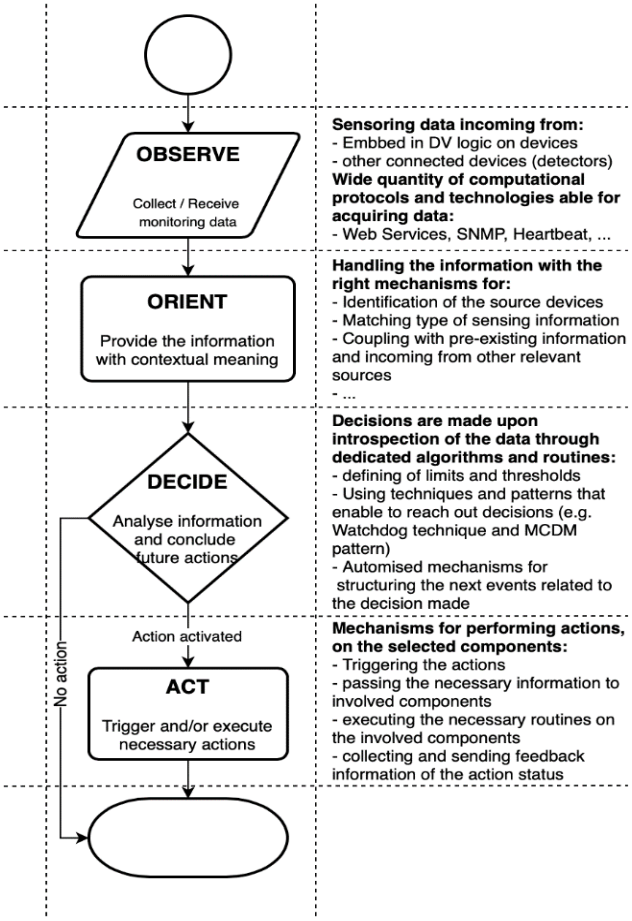

Fig. 7. DV Control Flow Loop.

V. Application Scenario and Results

A. Exemplary Application Scenario

A simple exemplary application scenario that involves the execution of a mitigation action – Migration – is described to show the potential of DV strategy. The scenario includes the following stages:

a) Monitoring of DV Asset devices connectivity and detection of a connectivity.

b) Deciding about the detected status of the *DV Asset* device, i.e. identify the best action to be taken (migration in the example) as well as the involved actors.

c) Acting: the *DV&AM* device triggers the action by sending information to the involved *DV Asset* devices, while these execute the action according to the information received.

The system configuration is based on the System Model presented in Section III, where PDC devices are connected to PMU devices that – in turn – are connected to a Real Time Digital Simulator (RTDS) to deliver measurement data to the PDC. PDC executes internal applications (simulation scripts) for collecting and processing data from the PMUs and data provisioning to the Control Center. All these components together represent a substation (Fig. 8).

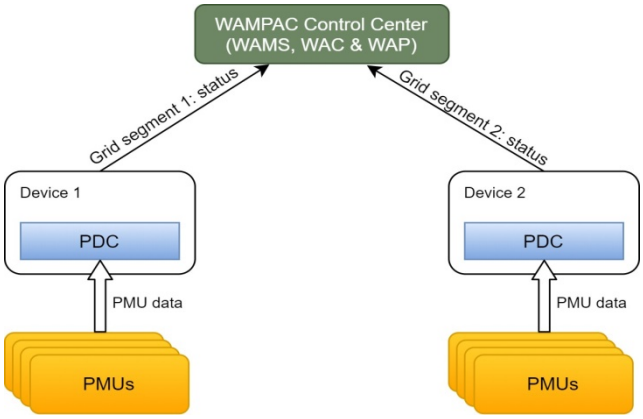

Fig. 8. Simple Grid Monitoring use-case using PMU Measurements.

In this scenario, PDC devices are "enhanced" with DV logic in order to act as *DV Asset* devices. Next to the PDC a *DVA&M* device is added to monitor and act on the *DV Asset* devices. At this point the initialization of the system can take place and *DV Asset* device applications and/or virtualized functionalities are communicated to the *DVA&M* together with connections details to enable the execution of the *Monitoring services (*Watchdog and Heartbeat services). A third *DV Asset* device is also included and used as "supporting" device during the execution of the migration action (see Fig. 9). Once initialized and all the conditions are settled, the DV logic can take place.

Conceptually the OODA loop is executed by the overall system. In particular, the *Observation* stage runs with the *DV Asset* devices (i.e. "enhanced" PDC) that use the internal *Heartbeat* service to communicate their connectivity status. The status is monitored by the *DVA&M* device that in turn uses the *Watchdog* internal service for detecting any connectivity failure of the related *DV Asset* devices. At some point in time a connectivity failure (connection timeouts, communication delays, etc.) of a specific *DV Asset* device is detected by the *DVA&M* device, the *Decision* stage can start. During this stage, the *DVA&M* is responsible for deciding the most suitable mitigation action and, in particular, to start the migration process which involves the following steps:

1) Identification of the best suitable *DV Asset* device that can host the virtualized functions of the faulty *DV Asset* device;

2) Move the virtualized functions of the faulty *DV Asset* device to the new selected *DV Asset* device; and.

3) Track the migration process while keeping updated the tasks of the *Observation* stage.

In the current scenario the system configuration evolves from the one depicted in Fig. 9 to a newer configuration (see Fig. 10), where the "supporting" *DV Asset* device is now running the virtualized functionalities of the faulty *DV Asset* device, i.e. it shows the same behavior of the faulty *DV Asset* device.

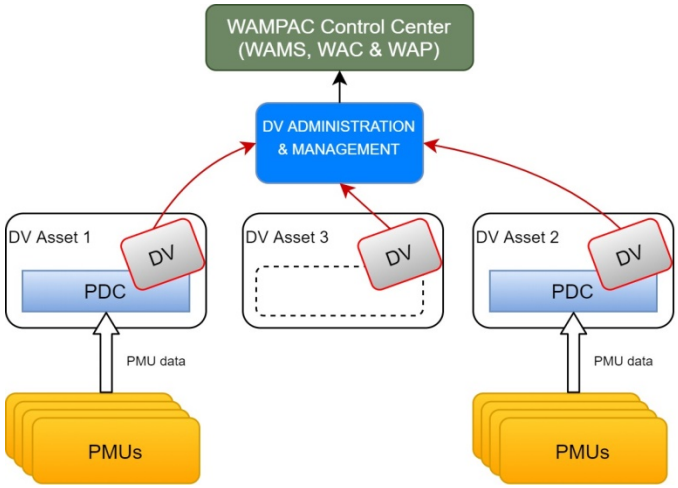

Fig. 9. Grid Monitoring use-case using PMU Measurements with DV Components Integrated.

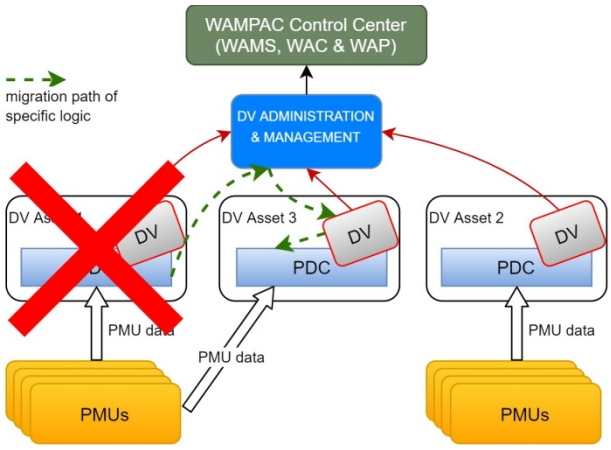

Fig. 10. System State after Migration Process Triggered by a Faulty Connection.

B. Results and Discussion

To measure and quantify the impact DV strategy has in the normal operation of the system the following metrics are considered:

- The time instant a connectivity failure is detected by the *DVA&M* device by using the internal *Watchdog* service.
- The time instant a decision is taken by the *DVA&M* device, i.e. a suitable *DV Asset* device is identified and all the virtualized functions of the faulty device are ready to be moved.
- The time instant the action is practically executed by the *DVA&M* device, i.e. the *DVA&M* device connects to the selected *DV Asset* device (through the *DV API*) and send the virtualized functions and configurations.
- The time instant the action is concluded, i.e. when a notification is received by the *DV Asset* device.

These time instants are shown in Table I, where each row identifies the execution of a migration action.

TABLE I. TIMESTAMPS RELATED TO THE DV AND AM MITIGATIN PROCESS

detection	decision	action	mitigation end
1578421500738	1578421500740	1578421500744	1578421501216
1578422203744	1578422203746	1578422203750	1578422204211
1578422358906	1578422358907	1578422358912	1578422359369
1578423527297	1578423527298	1578423527301	1578423527758
1578423642621	1578423642624	1578423642627	1578423643088
1578423779547	1578423779547	1578423779549	1578423779906
1578423886704	1578423886705	1578423886709	1578423887170
1578424038372	1578424038374	1578424038377	1578424038842
1578424374310	1578424374312	1578424374316	1578424374779
1578424506782	1578424506783	1578424506789	1578424507247

From the values of Table I, it is possible to determine the time consumed on the distinct stages, once a failure is detected (i.e. decision, action preparation and execution times), as well as, the total time consumed, from the detection to the end of the mitigation action. These times are shown in Table II.

The values show an extremely reactive system, where the decision and action preparation times are less than 10ms. On the contrary the action execution time is where the system consumes most of the total time. This time consumed is mainly due to the technological constrains (REST connections and implemented interaction protocols). However, the total time needed for a complex mitigation action is still on the order of milliseconds. Finally, during the execution of a mitigation action (in the example migration) there is always some data that is lost. This loss of data is directly related to the time needed for executing the mitigation action (in the example the mean time of 457ms as shown in Table II). It is possible to quantify the loss of data too by considering that PMUs devices publish measurements each 20ms or in other words with a 50 frames per second rate. Therefore, by capturing the time instant of the last data frame received by the *DV Asset* device before the connectivity fault and the time instant of the first data frame after the migration action is executed the loss of data can be estimated (see Table III).

TABLE II. TIME CONSUMED BY DVA AND M DEVICE DURING MIGRATION

	Decision time (ms)	Action preparation time (ms)	Action Execution time (ms)	Total time (ms)
	2	4	472	478
	2	4	461	467
	1	5	457	463
	1	3	457	461
	3	3	461	467
	0	2	357	359
	1	4	461	466
	2	3	465	470
	2	4	463	469
	1	6	458	465
Mean	2	4	451	**457**

TABLE III. TIME ELAPSED OF DATA FRAMES ACQUIRED BETWEEN FAULTY AND NEW DEVICE

	timestamp from last frame before fault	Timestamp from first frame after migration	time lapse (ms)	lost frames
	1578421501400	1578421500540	860	43
	1578422204400	1578422203600	800	40
	1578422359560	1578422358700	860	43
	1578423527940	1578423527100	840	42
	1578423643280	1578423642480	800	40
	1578423780100	1578423779320	780	39
	1578423887360	1578423886540	820	41
	1578424039000	1578424038200	800	40
	1578424374960	1578424374100	860	43
	1578424507420	1578424506620	800	40
Mean			**822**	**41**

Finally, in Table IV, the approximated time consumed during the deployment, initialization, and configuration of the migrated functionalities in the new *DV Asset* device is given. This time is calculated by correlating the values gathered from the previous tables.

TABLE IV. APPROXIMATION OF *DV ASSET* DEVICE TIME ELAPSED WHILE LAUNCHING FLOWS AND INITIALIZING THE APPLICATION SCRIPTS

	Migration duration on DVA&M (ms)	Time elapsed of lost frames (ms)	Approximated time of processing and initializing received flows on the *DV Asset* device (ms)
	478	860	382
	467	800	333
	463	860	397
	461	840	379
	467	800	333
	359	780	421
	466	820	354
	470	800	330
	469	860	391
	465	800	335
Mean	457	822	**366**

The approximated time of processing and initializing flows on the DV Asset device strictly depends on the technology chosen and on technological constraints (such as type of communication channels and interaction protocols), and, thus, this is the first parameter the authors are working on for DV strategy performance improvement.

VI. CONCLUSIONS AND FUTURE DEVELOPMENTS

In this paper the authors analyzed the importance of WAMPAC to gain a real-time awareness of the current state of the smart grid while ensuring the correct operation. Securing WAMPAC is – thus – a key priority. In this landscape, the authors presented a new and improved technological environment for developing and deploying the novel approach/strategy – i.e. "Double Virtualization" – to ensure the robust and resilient design and management of critical energy infrastructures based on CPSs. The proposed approach/strategy deeply relies on current trends in internet technologies and advanced computing technique where virtualization of the

resources is demanded. Furthermore, the evolution of current systems and processes into networks of CPS, and the related separation between "cyber" and "physical" dimensions creates the foundations for "Double Virtualization". As a matter of fact, this separation allows the creation of a highly dynamic and evolvable environment where functionalities (that lives in the "cyber" dimension) are detached from the specific runtime, protocols and communication needs. By appropriately exposing WAMPAC physical devices (PDCs) in terms of their functionalities it is possible to migrate functionalities from one device to another in the presence of cyber-attacks. Finally, the paper provides an exemplary application scenario to validate the proposed approach/strategy while summarizing how cyber-attacks on WAMPAC can trigger mitigation actions consisting in the migration of the functionalities from one runtime to another. However, further experiments need to be conducted to optimize the "Double Virtualization" i.e. optimization of the mitigation strategies, new monitoring algorithms for detecting abnormal behaviors and cyber-attacks within WAMPAC, integration of multi-criteria decision making (MCDM) algorithms to expand and improve decision agility, and investigate a distributed management strategy to handle any failure of the DVA&M component.

ACKNOWLEDGMENT

This work has been developed with the support of funds provided by the European Commission in the scope of H2020 DEFENDER project (grant nr. 740898) and PROPHESY project (grant nr. 766994).

REFERENCES

[1] Q. Zhu and T. Başar, "Robust and resilient control design for cyber-physical systems with an application to power systems," in 2011 50th IEEE Conference on Decision and Control and European Control Conference, Dec. 2011, pp. 4066–4071, doi: 10.1109/CDC.2011.6161031.

[2] C. Y. T. Ma, N. S. V. Rao, and D. K. Y. Yau, "A game theoretic study of attack and defense in cyber-physical systems," in 2011 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), Apr. 2011, pp. 708–713, doi: 10.1109/INFCOMW.2011.5928904.

[3] A. Ashok, A. Hahn, and M. Govindarasu, "Cyber-physical security of Wide-Area Monitoring, Protection and Control in a smart grid environment," J. Adv. Res., vol. 5, no. 4, pp. 481–489, Jul. 2014, doi: 10.1016/j.jare.2013.12.005.

[4] Y. Mo et al., "Cyber–Physical Security of a Smart Grid Infrastructure," Proc. IEEE, vol. 100, no. 1, pp. 195–209, Jan. 2012, doi: 10.1109/JPROC.2011.2161428.

[5] S. Majumder, A. Mathur, and A. Y. Javaid, "Cyber-Physical System Security Controls: A Review," in Cyber-Physical Systems: Architecture, Security and Application, S. Guo and D. Zeng, Eds. Cham: Springer International Publishing, 2019, pp. 187–240.

[6] P. Leitão, A. W. Colombo, and S. Karnouskos, "Industrial automation based on cyber-physical systems technologies: Prototype implementations and challenges," Comput. Ind., vol. 81, pp. 11–25, Sep. 2016, doi: 10.1016/j.compind.2015.08.004.

[7] R. Baheti and H. Gill, "Cyber-physical systems," Impact Control Technol., pp. 161–166, 2011.

[8] Acatech, Cyber-Physical Systems - Driving force for innovations in mobility, | acatech | Springer. 2011.

[9] Steering Committee, "Strategic R&D Opportunities for 21st Century Cyber-Physical Systems, Connecting computer and Information systems with physical world," 2013.

[10] M. Cengarle, S. Bensalem, J. McDermid, R. Passerone, A. Sangiovanni-Vincetelli, and M. Torngren, "Characteristics, capabilities, potential applications of Cyber-Physical Systems: a preliminary analysis," D2.1, Nov. 2013.

[11] D. Bytschkow, A. Campetelli, M. V. Cengarle, M. Irlbeck, and K. Schorp, "Reference Framework for the Engineering of Cyber-Physical Systems: A First Approach," 2014.

[12] G. Candido, C. Sousa, G. Di Orio, J. Barata, and A. W. Colombo, "Enhancing device exchange agility in Service-oriented industrial automation," in 2013 IEEE International Symposium on Industrial Electronics (ISIE), May 2013, pp. 1–6, doi: 10.1109/ISIE.2013.6563808.

[13] A. Rocha et al., "An agent based framework to support plug and produce," in 2014 12th IEEE International Conference on Industrial Informatics (INDIN), Jul. 2014, pp. 504–510, doi: 10.1109/INDIN.2014.6945565.

[14] G. Di Orio, A. Rocha, L. Ribeiro, and J. Barata, "The PRIME Semantic Language: Plug and Produce in Standard- based Manufacturing Production Systems," presented at the The International Conference on Flexible Automation and Intelligent Manufacturing (FAIM2015), Wolverhampton, UK, 23 - 26 June 2015, 2015, Accessed: Feb. 12, 2016. [Online]. Available: http://www.diva-portal.org/smash/record.jsf?pid=diva2%3A858181&dswid=8588.

[15] J. Soldatos, S. Gusmeroli, P. Malo, and G. Di Orio, "Internet of Things Applications in Future Manufacturing," in Digitising Industry - Internet of Things Connecting the Physical, Digital and Virtual Worlds, River Publishers, 2016.

[16] G. D. Orio, P. Maló, J. Barata, M. Albano, and L. L. Ferreira, "Towards a Framework for Interoperable and Interconnected CPS-populated Systems for Proactive Maintenance," in 2018 IEEE 16th International Conference on Industrial Informatics (INDIN), Jul. 2018, pp. 146–151, doi: 10.1109/INDIN.2018.8472041.

[17] ZVEI, "The Reference Architectural Model RAMI 4.0 and the Industrie 4.0 Component," Industrie 4.0, 2015. http://www.zvei.org/en/subjects/Industry-40/Pages/The-Reference-Architectural-Model-RAMI-40-and-the-Industrie-40-Component.aspx (accessed Nov. 11, 2015).

[18] ZVEI, "Examples of the Asset Administration Shell for Industrie 4.0 Components – Basic Part." Apr. 2017, Accessed: Nov. 01, 2018. [Online]. Available: https://www.zvei.org/en/press-media/publications/examples-of-the-asset-administration-shell-for-industrie-40-components-basic-part/.

[19] M. P. Barrett, "Framework for Improving Critical Infrastructure Cybersecurity Version 1.1," NIST Cybersecurity Framew., Apr. 2018, doi: http://dx.doi.org/10.1002/https://dx.doi.org/10.6028/NIST.CSWP.04162018.

[20] R. D. Alexander and S. Panguluri, "Cybersecurity Terminology and Frameworks," in Cyber-Physical Security: Protecting Critical Infrastructure at the State and Local Level, R. M. Clark and S. Hakim, Eds. Cham: Springer International Publishing, 2017, pp. 19–47.

[21] Y. Tada, H. D. Chiang, H. Li, A. Ishibashi, and Y. Serizawa, "A hierarchical WAMPAC system: Demonstration and evaluation," in 2013 IEEE Power Energy Society General Meeting, Jul. 2013, pp. 1–5, doi: 10.1109/PESMG.2013.6672543.

[22] A. Sadu, L. Ostendorf, G. Lipari, F. Ponci, and A. Monti, "Resilient design of distribution grid automation system with CALVIN," in 2018 IEEE International Energy Conference (ENERGYCON), Jun. 2018, pp. 1–6, doi: 10.1109/ENERGYCON.2018.8398833.

[23] J. R. Boyd, "The essence of winning and losing," Unpubl. Lect. Notes, vol. 12, no. 23, pp. 123–125, 1996.

[24] A. Sadu, G. Lipari, D. Gyorgy, and Z. Peiyue, "The Resilience by Design Concept, V2," D2.5, 2018. [Online]. Available: https://success-energy.eu/files/success/Content/Library/Deliverables/700416_deliverable_D2.5.pdf.

[25] P0_C37.118.2_WG - Synchrophasor Data Transfer for Power Systems, "IEEE C37.118.2-2011 - IEEE Standard for Synchrophasor Data Transfer for Power Systems," 2011. https://standards.ieee.org/standard/C37_118_2-2011.html (accessed Apr. 15, 2019).

[26] "Handbook of System Safety and Security - 1st Edition." https://www.elsevier.com/books/handbook-of-system-safety-and-security/griffor/978-0-12-803773-7 (accessed Apr. 11, 2019).