HSCLOUD: CLOUD ARCHITECTURE FOR SUPPORTING HOMELAND SECURITY

M. Fazio, M. Paone, A. Puliafito and M. Villari

Faculty of Engineering

University of Messina

Messina, Italy

Email: mfazio{mpaone, apuliafito, mvillari}@unime.it

Submitted: Feb. 21, 2012 Accepted: Feb. 22, 2012 Published: Mar. 1, 2012

Abstract- Governmental institutions all over the world are trying to increase the level of security of their countries emphasizing the usage Information Technology solutions. We believe that Cloud Computing may strongly help Homeland Security, since it offers a very flexible support for organizing and managing heterogeneous systems, providing huge amount of processing, storing and sensing resources. In this work we introduce a new Cloud architecture able to virtualize different types of sensing environments in* virtual sensing elements*, logically belonging to* Cooperating Clouds*. It

represents a very flexible solution, which offers seamless, secure and advanced services to support Homeland Security.

Index terms: Homeland Security, Cloud Computing, Dangerous Goods, Virtual Sensing.

I. INTRODUCTION

Recent events that have mined the safety and security of our countries show the importance in increasing the defenses against terrorist attacks and intrusion detection [1][2]. The 9/11 attack, carried out by terrorists instructed by the Al Qaeda leadership in Afghanistan, has changed our perception of terrorism. It is not the isolated event that has upset our history in last years. In Detroit, the failed attempt to blow up an airplane on Christmas Day 2009 was planned in Yemen; Iran has been behind terrorist attacks carried out by its proxies, Hizballah and Hamas; the terrorist attack at Moscow's Domodedovo airport, 25 January 2011, killed 35 people and wounded 180; in the UK, we still face threats from dissident republicans in Northern Ireland; just over a year ago, several missions to the UN received hoax biological attacks in New York, emanating from Texas. According to an internal report of Department of Homeland Security issued on Friday, May 21 2010, the number of attempted terror attacks against the U.S. over the last nine months has surpassed the number of attempts during any previous one-year period. All these examples are a clear evidence that efforts and policies for Homeland Security need to be strengthened in all countries.

The US Department of Homeland Security has focused the main activities against terrorism on three goals:

1) prevent terrorist attacks;
2) prevent the unauthorized acquisition, importation, movement, or use of chemical, biological, radiological, and nuclear materials and capabilities within the United States;
3) reduce the vulnerability of critical infrastructure and key resources, essential leadership, and major events to terrorist attacks and other hazards.

Governmental institutions all over the world are enforcing their activities against the terrorism cooperating with computer scientist, in order to design and implement effective solutions to improve what

is commonly referred as Homeland Security (HS) [3][4][5]], i.e. a concerted effort to prevent terrorist attacks, reduce vulnerability to terrorism, and minimize the damage and recover from attacks that do occur.

Nowadays many human activities find a valid support in using IT (Information Technology) solutions, especially for massively collecting, processing and storing of data. Procedures automation, mobile device interconnections social data inter-exchange (i.e. Facebook applications [6]) are increasing our quality of life. In particular, the massive usage of sensors, mobile devices and wireless communication technologies is drastically transforming our habits, leading toward the deployment of autonomic systems in our houses, offices, hospitals and so on. We believe that IT can be also the keystone for future activities for HS.

In this paper we deal with the HS issue by recurring to the Cloud Computing paradigm. Cloud computing is a new computation paradigm, which aims to pursue new levels of efficiency in delivering services, representing a tempting business opportunity for IT operators of increasing their revenues. We believe that Clouds may really be useful in many application contexts, for several reasons: 1) it is oriented toward the virtualization of resources, enabling an easy management of heterogeneous systems, 2) it offers services over a world wide area, 3) it provides a huge amount of resources in terms of storage, processing and sensing, 4) it guarantees high flexibility and versatility. In the context of HS activities, it represents a very strategic technology to strengthen defense mechanisms, especially thanks to its ability in supporting cooperation and integration among different frameworks and entities. In our idea, all private companies, military or governative departments, civilian organizations operating in HS should merge their efforts, resources and strategies to create a complete and efficient shield against terroristic attacks. We point out that the novelty of our work is not about the introduction of new threats investigation or new data analysis methodologies or algorithms. We propose a new way for pursuing much more concrete results reusing consolidated approaches by leveraging Cloud-based technologies.

For this purpose, we have to uptake the capability of heterogeneous Clouds to collaborate each other for increasing their productivity and efficiency. Cooperating Clouds have to form a federation, where virtual

resources and services are shared. In this paper, we present a new Cloud architecture to support HS, called Homeland Security Cloud (HSCloud). It aims to reduce vulnerability to terrorism by coupling activities of monitoring and detection with advanced features in the treatment, filtering and provisioning of collected data. It offers a very flexible platform to develop algorithms and solutions for the prevention of possible attacks. Whenever terrorist attacks occur, HSCloud provides a communication infrastructure useful to integrate efforts and resources of different organizations involved in the disaster, giving a concrete support for a common planning of rescue activities. HSCloud represents a very flexible architecture and offers many advantages in terms of availability, security and dependability. The abstraction layer of HSCloud has been developed according to the Sensor Web Enablement (SWE) standard defined by the Open Geospatial Consortium [7], in order to introduce in the environment new virtual pervasive elements able to offer different types of service, in terms of infrastructures, platforms and applications.

To show the effective goodness of HSCloud, we discuss a specific and strategic scenario for HS, that is the Transportation of Dangerous Goods (TDGs) [8] over multi-modal ways, such as freeways, railways, air routes and sea routes. The area of goods tracking has attracting great interest due to the congenital high potential risk. Systems for TDGs capable to reduce risks of terrorist attacks make extensive use of sensing infrastructures to assess risks or to detect unusual events. To import such complex monitoring services into the Cloud, the sensing infrastructures have to be virtualized. In this paper we investigate how to implement virtual sensing infrastructures in HSCloud and discuss the advantages that our approach introduces in existing TDGs solutions.

The paper is organized as follow. In Section II we highlight the main issues in TDGs, presenting current solutions in literature to tackle them. In Section III, we present the Cloud computing paradigm, the state of the art on the integration of sensing technologies in the Cloud. In particular we focus our attention on the Cloud middleware called CLEVER, which has been used for implementing HSCloud. Then, Section IV discusses the actual benefits of applying Cloud computing in the management of TDGs. The HSCloud architecture is presented in Section V. Section VI describes in detail the core of

HSCloud, the SensCLEVER middleware, which is a minimal implementation of CLEVER oriented to the sensing service. In Section VII we give a description of users that can be interested in using HSCLoud, explaining their possible contributions and advantages. Section VIII describes the implementation of our first prototype of HSCloud. Finally, Section IX provides our conclusions and guidelines for future advances.

II. Transportation of Dangerous Goods (TDGs): problems, solutions and advances

Most of the times, people are not aware about the risks related to TDGs, even more if it becomes the target of terrorist attacks. Indeed, dangerous goods can cause terrible disaster if an accident occurs, producing uncontrollable effects in highly populated areas or during popular events. So, terrorist attacks increase the hazard of TDGs of hundreds of times.

The TDGs is a very complex problem, involving economical, legislative and technological aspects. Nowadays, however, advanced technologies in the field of ICT (Information and communications technology) promise a way to track in real time the entity of such transportations and efficiently manage the exposure to related risks. Innovative technologies can actively support goods tracking and provide valuable added value services to provide legally requested information and also to minimize risk in case of failures and accidents.

TDGs ask for a continuous monitoring of activities related to transportation. It is necessary not only to log the position of the vehicle and the status of the cargo, but also to understand how the environment interacts during the transportation of dangerous goods. Automatic vehicle identification techniques relying on Radio Frequency Identification (RFID) permit to electronically gather shipment information. Route planning can reduce the probability of disaster. It can be time-independent or reactive. In particular, route planning is reactive if real-time information about the conditions of the transport network are periodically updated in the management system. Such information are gathered by sensor networks and made available in real-time databases. Also, Geographic information System (GIS) will permit geospatial data management for decision making processes.

An analysis of the state of the art shows a great interest on the TDGs. This is demonstrated by the existence of many solutions, projects and business products falling in this area.

A. The state of the art on TDGs

Dangerous good transportation has gathering great attention from the research community and business companies. The main goal is to provide a framework for goods monitoring and activities planning, in order to prevent disaster and to manage activities if accident occurs.

MITRA [9] is a research project funded by the European Commission with the objective to prototype a new operational system based on regional responsibilities for the monitoring of dangerous goods transportation in Europe. It provides a real-time knowledge of position and contents of dangerous goods through the European Geostationary Navigation Overlay Service (EGNOS), that is a satellite based augmentation system developed by the European Space Agency, the European Commission and EUROCONTROL. In case of dangerous situations, GSM communications allow to alert the Security Control Centre, which is responsible to prevent accidents, manage crisis and enable quick intervention.

SMARTFREIGHT [10] is a European research project, partly funded by the European Commission under the 7^{th} Framework Program (7FP). The overall objective of SMARTFREIGHT is to address new traffic management measures towards individual freight vehicles by using open ICT services, with an emphasis on the interoperability between traffic management and freight distribution systems, and an integrated heterogeneous wireless communication infrastructure within the framework of CALM (Communication Access for Land Mobiles)

In [11], the authors propose a complete monitoring and tracking solution for truck fleets. The system exploits battery-powered environmental sensors (temperature, humidity, pressure, gas concentration and ionizing radiation levels), connected by a ZigBee-based Wireless Sensor Network. Collected data is then sent from the vehicle to a remote server via a GPRS link. The GPS positioning system is integrated by the use of an Inertial Navigation System, which guarantees a precise estimate of the position also when the GPS signal is weak or temporarily lost.

The solution proposed in [12] aims to improve the security of maritime container transport of dangerous goods by the real-time monitoring of container state. This system uses micro-sensor technologies and radio frequency communication technology to obtain the dangerous goods condition inside containers, as well as automatic positioning in the cargo hold. Information on the state of dangerous goods are transmitted to the shore monitoring center on land through INMARSAT stations. By comparing the different solutions for dangerous goods transportation, we have identified the following common goals: 1) localization and tracking means of freight transportation, 2) monitoring of goods according to several types of information (temperature, pressure, gas detection,...), 3) data collection and elaboration, 4) definition of policies for disaster prevention, 5) definition of policies for emergency management. However, the existing solutions differ a lot in terms of sensor technologies, communication infrastructures, design of the system organization and software support.

B. Open issues

Companies operating in the monitoring of dangerous goods have to use specific technologies that depend on several factors: the type of dangerous goods that are tracked, their geographical position and route, means of transport, legislation of the country and so on. International Regulations define standard procedures for the treatment of dangerous goods. However, from a technological point of view, they do not provide any specification with reference to the monitoring infrastructure installation. The result is that actually there is no compatibility between different monitoring systems managed by organizations or companies, both in terms of hardware and software.

Another important point is related to the transportation solution adopted. Each solution focuses on a specific method of transportation (such as ship, truck, airplane or railways) and the concept of multi-modal service is not faced at all. However, the aggregation of information from multi-modal ways can be extremely useful to predict terrorist attacks. Furthermore, in case of attacks, the management of different types of way out from the disaster area can save human lives.

A world wide solution is still missing. Recent events have shown the importance of collaboration

among different countries to fight against terrorism. So, we imagine a future HS system where efforts will integrate activities along the roads, highways, railways, harbors and airports at once. The integration will also include activities provided by different operators inside the same country and among different countries.

C. A Model for TDGs systems

The existing solutions to improve the security offered in TDGs are composed of two main components: 1) a Sensing Infrastructure (SI), able to monitor the state of the goods, their position and movements and additional information on the environment. It can be mobile or fixed, according to the involved technologies, and allows to deliver all the sensed information towards Information Data Centers (IDCs). 2) Several IDCs are responsible to manage activities for accident prevention and/or rescue operations. They host algorithms for HS management and policies to control actuators in the SI.

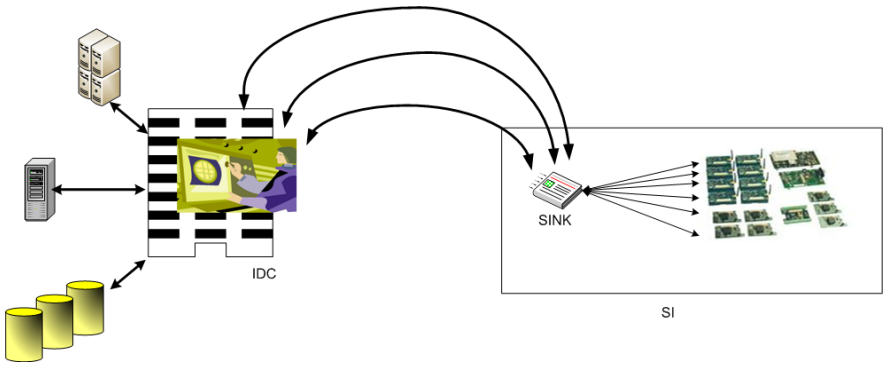

Fig. 1. Architecture of systems for monitoring and managing TDGs

The SI is usually composed of a set of sensors equipped with a communication interface, properly deployed in the monitored area. Sensors transmit sensing data towards a sink node. The sink is the connection point between the sensor network and one or more IDCs and is able to manage multiple connections in order to improve the availability and reliability of the system. It can provide sensing data to IDCs. Few solutions consider the possibility of active communications from an IDC to the sink, in order to actuate different functionalities on specific sensor nodes. IDCs track movements of dangerous goods, store sensing data in dedicated data centers and elaborate them in order to prevent possible attacks

and map up activities and strategies to handle crisis.

According to Figure 1, the activities carried on by the SI and the IDC can be separated. Each company or organization in HSCloud can be involved in a specific task, taking benefits from activities or infrastructures of different federated entities. For examples, business companies leader in producing sensors and hardware platforms will improve the HS by developing a SI regardless of how sensing data will be managed and without being aware of any implementation detail. In the scenario we have in mind it is possible to distinguish among companies involved in the SI, that we call SI Managers (SIMs), and organizations in charge of the IDC services, that we call IDC Managers (IDCMs).

III. Briefly on Clouds

Cloud Computing is a very challenging technology that many generally considered as one of the more challenging topic in the IT world, although it is not always fully clear what its potentialities are and which are all the involved implications. Many definitions of Cloud computing are presented and many scenarios exist in literature. In [13] Ian Foster describes Cloud Computing as a large-scale distributed computing paradigm that is driven by economies of scale, in which a pool of abstracted, virtualized, dynamically-scalable, managed computing power, storage, platforms, and services are delivered on demand to external customers over the Internet. Until now, such trend has brought the steady rising of hundreds of independent, heterogeneous Cloud providers managed by private subjects yielding various services to their clients. Services might be software, platform or infrastructure (SaaS, PaaS, IaaS), whereas clients might range from other Clouds, organizations, and enterprises to single users. ICT developers are exploring to improve the economy of scale, the efficiency of their systems and, also, for reducing the power consumptions and CO2 emissions. In order to provide a flexible use of resources, Cloud computing makes wide use of virtualization, allowing to treat traditional hardware resources like a pool of virtual ones. In addition, virtualization enables the resources migration, regardless of the underlying physical infrastructure. Using virtualization and aggregation techniques, Cloud computing offers its available resource *as a Service* rather than as physical product. In particular, according to the NIST formalization [14], it provides services at

three different levels:

- *Software as a Service* (SaaS): it represents the capability given to consumers of accessing provider's applications running on a Cloud infrastructure. Many Cloud services are already available at this level, such as Amazon Storage, DropBox Storage, Google Map, Google Docs and Microsoft Office Online.
- *Platform as a Service* (PaaS): it makes consumers able to deploy their own applications onto the Cloud infrastructure using programming languages and tools supported by PaaS Cloud providers. Typical examples of PaaS services are given by Social platforms as well Facebook, Twitter, LinkedIn, Google Apps.
- *Infrastructure as a Service* (IaaS): it provides consumers with computation, storage, networks data transfer and other computing resources. Consumers are able to deploy and run arbitrary software, which can include operating systems and applications. For example, this type of service is offered by Amazon EC2, Rackspace, Salesforce.

Cloud computing exploits whatever virtual technologies for making an abstraction on data, processing, and storage. Virtual Machines (VMs) represent the typical example of how virtualization technology can be used in Cloud. Cloud costumers are able to preconfigure VMs and to deploy them on the Cloud infrastructure, without any further configuration. VMs may collect data, execute their elaboration, migrate the data if necessary, expose APIs to be used from other VMs being executed in different Clouds and so on. To provide the effective integration of sensing technologies into the Cloud, specific virtualization techniques for monitoring systems need to be exploited. In Section III-A we present current solutions in literature to integrate sensing resources and Cloud computing. Then, in Section V we propose our solution, which is able to offer several benefits at the IaaS, PaaS and SaaS layers. It has been developed by using a very innovative Cloud Middleware, called CLEVER [15], which is briefly introduced in section III-B.

A. The state of the art on Cloud and sensing technologies

In [16], the authors propose the use of Cloud for collecting personal health data, in particular for monitoring ECG (ElectroCardioGram) values. Authors have designed a real-time health monitoring and analysis system that should be scalable and economic for people who require frequent monitoring of their health. They focused on the design aspects of an autonomic Cloud environment that collects health data and disseminates them to a Cloud based information repository, facilitating analysis of data by using software services hosted in the Cloud. Persons under their assessment have to wear PDAs with sensors able to catch some physiological data. Thanks to Wi-Fi/UMTS network connections PDAs are able to send to Clouds data they aggregated on board. One of the main concern in using public Cloud services (see Amazon storage), is about the level of security and privacy that Cloud companies can offer to customers. In particular the security aspect is crucial all the times personal health data are collected. In our point of view the Cloud middleware should be aware of the *value* of data it has to manage.

In [17] the authors present a framework that provides a semantic overlay of underlying Cloud-based Internet of Things. The framework they propose introduces the concept of sensor-as-aservice (SenaaS) to address the connectivity issue with various types of sensor nodes. They employ enhanced semantic access polices to ensure access to only authorized parties. In this work we noticed a weakness of their system in explaining how they use Cloud technologies, in fact their dissertation shows an unclear collocation of the developed framework within the Cloud stack. They stated what are the advantages on leveraging Clouds, but how they integrate its system in the Cloud remains not comprehensible. The Cloud computing technology is not well faced at all. However, we agree with authors that XML technology is useful in Cloud scenarios. Indeed we also think that it may help for encouraging inter-operability among Clouds where even sensing technology is part of them. In particular SensorML (used in [7]) that provides metadata model in XML format to describe sensor, its capabilities and measurement process, represents a valid example of XML utilizations useful for Clouds.

Another solution that merges sensor and Cloud concepts is presented in [18]. The authors describe

the *Integration Controller interaction Architecture (*IICiA), which enables users to easily collect, access, process, visualize, archive, share and search large amounts of sensor data from different applications. The architecture should support complete sensor data life cycle from data collection to the backend decision support system. They characterize the Cloud technology by using a Service Oriented Architecture (SOA).

In [19] the authors present a model for Smart Grid data management based on specific characteristics of Cloud computing, such as distributed data management for real-time data gathering, parallel processing for real-time information retrieval, and ubiquitous access. They gather the requirements by utilizing REST based APIs to collect and analyze set of data from well-known smart grid use cases.

The authors in [20] present a platform, called ubiquitous Cloud, to exploit Cloud facilities. They propose adaptive services to manage ubiquitous resources, which are able to dynamically adapt their behaviors to requirements and contexts of the ubiquitous computing. To facilitate the management of ubiquitous service resources, their paper should present a platform called ubiquitous Cloud, borrowing the concept of the Cloud computing. The ubiquitous Cloud is developed by using SOA with SOAP based technology and supports several types of stakeholders, which can use specific ubiquitous objects at the infrastructure, platform and application levels. We noticed the use of SOA with SOAP based technology, that makes the proposed solution less suitable with the current Cloud technology. Besides the last part of their dissertation along with the description of what they provided, shows that Cloud is correctly mentioned but in reality the architecture does not look like as Cloud based infrastructure, especially in the representation of SaaS, PaaS and IaaS levels. They spread their modules *the service resource registry, the adaptive resource finder, the context manager and the service concierge* into this Cloud stack, regardless the meaning of the Cloud stack layers.

The authors in [21] proposed a way for managing Physical Sensors with Virtualized Sensors on Cloud computing. Giving a look at the table they reported in the final part of their dissertation it is possible to see the *pro and cons* of the proposed infrastructure named Sensor-Cloud. The description of the work is focused on virtual sensors and provide a further description of the Cloud they have used. However, in the

proposed solution, Sensor-Cloud administrators have to prepare the templates for virtual sensors and the need of human (end-users) interaction for setup IT resources is a very big fault. Indeed in the framework they have shown, the part that should make up the Cloud environment able to elastically receive sensing data is totally missing.

Since the Cloud paradigm is used in many application areas, we want to exploit its features for solving also issues related to HS in TDGs.

Finally in order to remark our contribution in the context of Cloud and sensors, our experience in both areas help us to identify what are the capabilities of each and how to exploit them in a synergic scenario. We believe, that our solution is able to overcome the simplistic view of Clouds, in which concrete benefits are ease to reach. Cloud Computing currently represents a very famous *buzzy word*, and it is not enough to mention it without considering the meaningful utilization of its real functionalities.

B. The CLoud-Enabled Virtual EnviRonment (CLEVER)

The CLoud-Enabled Virtual EnviRonment (CLEVER) is a Cloud middleware [15] which aims at the development of a Virtual Infrastructure Manager (VIM) for the administration of private Clouds infrastructures. Its main capability is to setup an overlay network useful for allowing the interaction of more Clouds spread over the Internet.

CLEVER was originally conceived as part of the IaaS level [15] for implementing the Virtual Infrastructure Management (VIM) layer. In general, a VIM manages the physical resource of a datacenter (i.e., a cluster of machines), interacting with end-users, providing them Accounting, Service Level Agreements (SLAs), Billings, etc. It dynamically creates and executes Virtual Machines (VMs) on the CLEVER hosts, considering their workload, data location and several other parameters.

To meet the requirements of sensing environments, we have extended the VIM functionalities of CLEVER in the SensCLEVER module, in order to manage the physical resources of SIs. There, we have added new capabilities for exposing specific services for sensing resource management, hiding underneath technologies. This has meant the definition of new PaaS functionalities in our middleware

(i.e., data filtering and aggregation, on-demand messages, etc.).

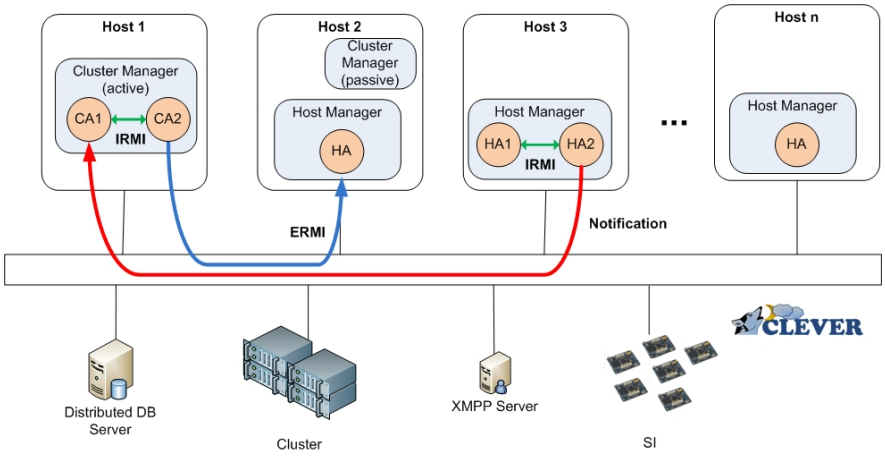

Fig. 2. CLEVER middleware

The middleware is based on a distributed clustered architecture, where each cluster is organized in two hierarchical layers, as depicted in Figure 2. CLEVER nodes contains an host level management module, called Host Manager (HM). A single node may also include a cluster level management module, called Cluster Manager (CM). The CM contains the intelligence for treating and analyzing all incoming data whereas the HM has simple characteristics at lower level. Indeed it represents the remote agent of the CM. Thus, we have in the cluster ay least one active CM at higher layer and, at lower layer, many HMs depending on it. A CM acts as an interface between the clients (software entities, which can exploit the Cloud) and the software running on the HMs. The CM receives commands from the clients, gives instructions to the HMs, elaborates information and finally sends results to the clients. It also performs the management of resources (uploading, discovering, etc.) and the monitoring of the overall state of the cluster (workload, availability, etc.).

An HM does not perform any data interpretation and/or evaluation, but it can be seen as a gateway towards the physical infrastructure. For example, it instantiates VMs and run them on the physical hosts, or gathers sensed data from the SNs forwarding them to the CM. Both CMs and HMs are composed by several sub-components, called *agents*, which are designed to perform specific tasks. To improve the readability of the paper, we name Cluster Agent (CA) an Agent in the CM and Host Agent (HA) an

Agent in the HM, but we emphasize that, from a technical point of view, there is not any difference in the design of CAs and HAs.

CLEVER supports three types of communication: Internal Remote Method Invoker (IRMI), External Remote Method Invoker (ERMI) and Notification. *An IRMI communication* refers to the message exchanging protocol among agents within the same manager (both in CMs and HMs). Since agents are separated processes running on the same host, IRMI communications are based on Inter Process Communications (IPCs).

ERMI communications allow to CA to exchange messages with HAs. In fact, each CA has knowledge of the agents in the HMs that depend on the CM itself. It is based on the XMPP protocol [22], which was born to drive the communications in the heterogeneous instant messaging systems, where it is possible to convey any type of data. In particular, the protocol has to guarantee the connectivity among different users even with restrictive network security policies (NAT transversal, firewalling policies, etc.). It is based on coupling of HTTP and XML, thus ensures the maximum level of flexibility. The XML versatility allows us to use the channel XMPP for the management, control data transfer in inter-site communications. The XMPP protocol is able to offer a decentralized service, scalability in terms of number of hosts, flexibility in the system interoperability and native security features based on the use of channel encryption and/or XML encryption, as it is furthermore described in the next section.

Unlike CAs, HAs do not have knowledge of the CAs at the upper-layer of the hierarchy. Even if this design choice seems to limit the interoperability among the agents, it allows to reduce the complexity of inter-module communications and to increase the scalability, fault-tolerance and availability of the system. To allows communications from an HM to the CM, CLEVER uses the *Notifications*, which are sent from an HA to the CM without specifying the CA interested in the communication. For example, in the provisioning of sensed data, an HA gathers data from a SI and forward them towards the Cloud independently from the particular services that will manipulate these pieces of information at the upper-layer. As the ERMI, also Notifications are based on the XMPP protocols, in order to benefit of flexibility

and versatility in communications.

Since both ERMI communications and Notifications are based on XMPP, the CLEVER architecture needs the presence of an XMPP Server, which guarantees a high fault tolerance level in communications and allows system status recovery if a crash of a middleware component occurs. The current implementation of CLEVER is based on the employment of an Ejabberd XMPP server [23].

The current implementation of our architecture is based on a specific plugin able to locally interact with the Sedna native XML database (see [24]). Sedna allows the possibility of creating incremental hot backup copies of the databases and supports ACID transactions. Although Sedna cannot be deployed in a distributed fashion, it has been preferred because it natively supports the XML data containers (as well SensorML and others). Thanks to XPath and XQuery capabilities of Sedna along with the flexibility for defining in run-time the XML DB Schema, Sedna simplifies the data storage along with the queries to perform on them. Hence we can consider our solution as a hybrid database, that is a DB that has a Relational Database behavior (we can make entity-relation-ships among the XML parts) but it can grow-up regardless a preconfigured and static schema (NoSQL approach: i.e. column-based as well the Cassandra DB [25]; the database used by Facebook).

CLEVER has been designed with an eye toward horizontal federation. The concept of federation has always had both political and historical implications: the term refers, in fact, to a type of system organization characterized by a joining of partially ”self-governing”’ entities united by a ”central government”. In a federation, each self-governing status of the component entities is typically independent and may not be altered by a unilateral decision of the ”central government”. The components of a federation are in some sense ”sovereign” with a certain degree of autonomy from the ”central government”: this is why a federation can be intended as more than a mere loose alliance of independent entities. The choice of using XMPP for the CLEVER module communication has been made thinking about the possibility to support in the future also interdomain communication between different CLEVER administrative domains. The interdomain communication is the base for the horizontal federation.

IV. Benefits of Cloud Computing in TDGs activities

Existing TDGs solutions can be seen as *Isolated Solutions* that might benefit of an interoperable framework able to address at once issues through the crossing analysis of data. Cloud Computing represents the *GLUE* to integrate and homogenize such heterogeneous systems. Governments are responsible to enforce the use of such *GLUE*. In this paper we also refer to multi-modal TDGs. In multi-modal systems, all data must be collected, organized and processed to provide an integrated knowledge base to build up strategies at the National Security level. Behind that, the HS system has to guarantee the control and management of multi-modal transportations activities even out of the country borders. It is not easy to develop a middleware for the cooperation of several different entities. By using the concept of virtualization, Cloud computing is the most suitable approach to guarantee the high level of interoperability requested.

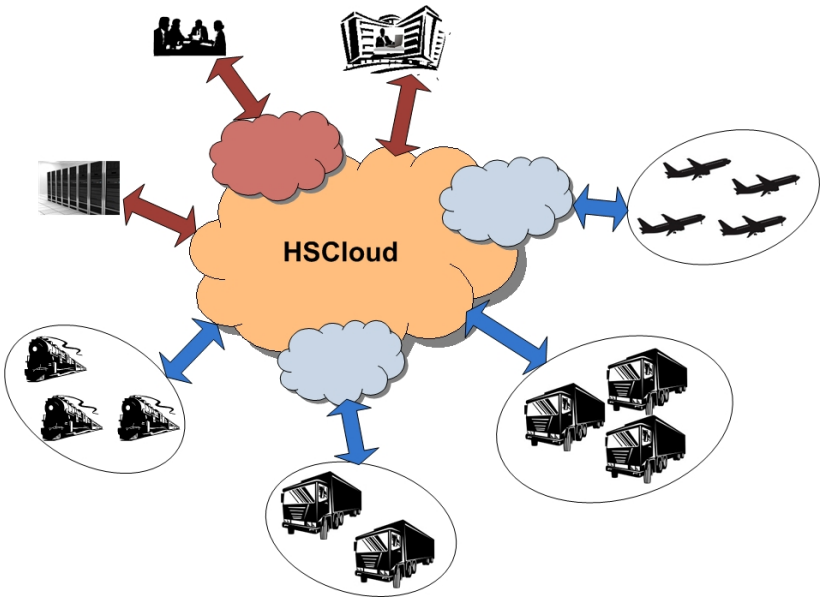

Fig. 3. How supporting TDG through the Cloud

In our idea, as shown in Figure 3, several companies and administrations work to monitor multi-modal TDGs and to manage them by using their own hardware and software systems. Cloud allows them to collaborate each other without any change in their heterogeneous infrastructures. It acts as an intermediary infrastructure spread over a wide geographical area, helping to easily overcome the

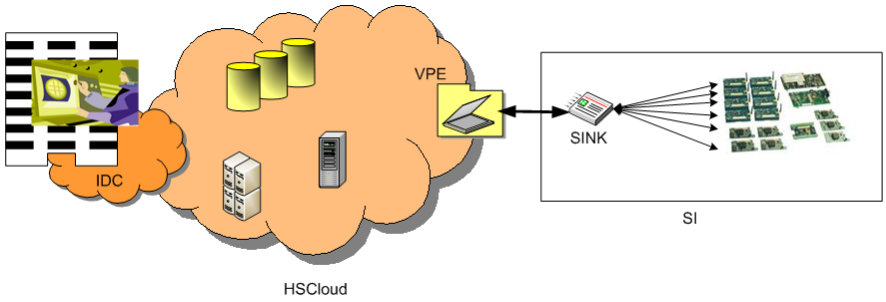

Fig. 4. Cloud-based architecture for monitoring and managing TDGs

management of distributed activities on territory. Each administration will see its SI extended thanks the SIs of the other components of the whole system. At the same time, each administration will be able to optimize the management policies of TDGs, through agreements with other administrations/companies and joint actions. In fact, Cloud can provide support for TDGs at different layers. As SaaS, user-friendly web service interface can be implemented to access information and activities available in the system. The plethora of applications already developed for planning efficient TDGs may convey on the Cloud, in order to increase their degree of accessibility and availability as PaaS. IaaS ensures high elasticity in the usage of available resources, thus determining higher efficiency to all the HS activities, shielding attacks and efficiently managing crisis conditions.

For the administrations that are already working using different Cloud environments, Cloud federation gives the possibility of sharing resources and services without any effort. We consider the Cloud as a constellation of hundreds of independent, heterogeneous, private/hybrid Clouds. In that context, one of the main challenges to address is the possibility that systems with administrative autonomy are able to interact if needed, maintaining separated their own domains and the specific administration policies. This requirement is particularly important in the HS area, because actors that interact to improve their HS do not intend to disclose their valuable owned data.

To provide an effective integration among the services offered by several monitoring systems, we need to recur to the virtualization of the SIs, i.e. abstracting each framework for secure goods transportation and provide it with a common interface toward the Cloud system. To this aim, we introduce the Virtual

Pervasive Element (VPE) in the HSCloud infrastructure, as shown in Figure 4. It is a PaaS necessary to integrate a SI inside the HSCloud architecture. It is able to gather information from the specific sensing infrastructure and transfer them into the Cloud federation in several formats: it can output data in JSON or XML formats to a remote Data Base Management System (DBMS) application. It can provide also a rich and extensible Application Programming Interface (API), creating an abstraction layer for data formatting. Furthermore, the VPE allows remote decision-making to actuate HS strategies. In fact, the IDC Managers in the HSCloud architecture can use remote automation methods to execute specific behaviors inside the SIs. This service is easily provided by CLEVER, which is based on a remote command-based communication paradigm. According to the physical features of the SI, the corresponding VPE publishes in the Cloud environment the interface for automation services. Thereby, only IDCs that have the license to act over the IS will be able to execute the commands provided in the SI interface.

V. The HSCloud Architecture

In this Section, we present the main functionalities of HSCloud, a new Cloud architecture for HS, which aims to reduce vulnerability to terrorist attacks and quickly respond if attacks occur by minimizing damages and recovery efforts. Video cameras, metropolitan sensors spread in a municipal area, vehicle traffic monitoring as well any kind of data acquisition generally allow to improve the coordination of human activities at any level. We are assisting to spasmodic autonomic control of Houses, Hospitals, Factories, etc (as an example give a look at the existing Smart Grids and Internet of Things initiatives). In this direction, sensor networks are becoming a pervasive technology that allows to monitor wide geographical areas to promptly detect critical operation conditions. Sensors are currently applied in many applications fields, such as in buildings construction,[26], cars traffic monitoring, [27]), environments analysis, [28], medical care assistance, [29]), weather forecast, [30], video surveillances, [31], etc. All the information are acquired by independent administrations, which deploy their own monitoring infrastructure and software architectures. In such a scenario we should start to figure out what the complexity is, especially in case of systems inter-connection allowing the cross correlating of sensing data.

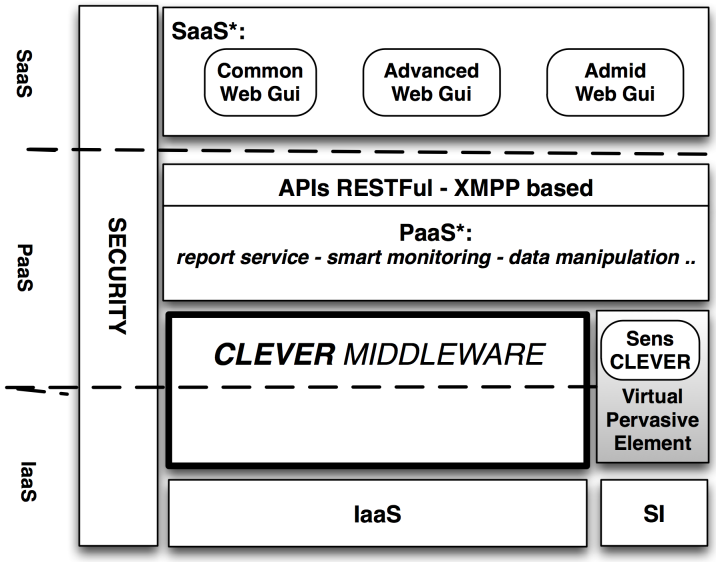

Fig. 5. Architecture of HSCloud

We believe that sensor networks should be integrated as part of our Cloud Infrastructure, virtualized if needed, and accessed as any other resource in the Cloud computing environment. In our view sensor networks have to cover an active role, and they represent the physical place where data are originally acquired and where possible actions might be forwarded. At the beginning of our discussion, we named some elements compounding Clouds as virtual.

In this work, we consider sensor networks as a virtual resource, which is accessible through a VPE. The collections of the VPEs is considered part of the Federated Clouds. Thanks to the sensing technology we can practically collect whichever data as we need: a wide choice of sensors are already available in the market with many different typologies of acquisition. It can also be noted the evolution of sensors towards Smart Sensors and Actuator Sensors. In the former version, customers might change on-demand the sensors behavior (i.e., data filtering, data aggregation, up to making local decision, etc.) thanks to customized programming code that can be instantiated on-demand in the sensor. In the latter version, that is Actuator Sensors, we consider such devices as elements able to perform remote actions on-demand.

Figure 5 shows the HSCloud architecture, which is based on the CLEVER middleware (see Section III-B). Since CLEVER provides a VIM to manage Cloud infrastructures, its implementation covers

some aspects at the IaaS layer. In fact, it has to manage the virtual resources executed on the Host Managers and the Cluster Managers by using low layer procedures. At the same time, its implementation covers aspects at the PaaS layer, as it has to offer the services that are necessary to make use of Virtual Machines in the system.

To have an efficent integration of heterogeneous SIs into the Cloud, we have implemented a modified version of CLEVER, called SensCLEVER. It has been designed to perform an intelligent provisioning of sensed data. This extend the functionalities of the CLEVER middleware to the PaaS layer. In Section VI, we describe the SensCLEVER internal architecture. SensCLEVER is part of the VPE, which represents the peripheral element necessary to endure the interaction between SIs and the Cloud services. VPEs have mainly two task. First, they gather information from the monitoring systems and provide advanced features to request, filter and retrieve them. Second, they act as a manager element, which coordinate and supervise monitoring activities, by wondering and comparing responses from different components in the monitoring environment. This task allows to improve the efficiency and robustness of monitoring systems, since it allows to integrate different kind of information over the controlled area and to detect misbehaviors in a subsets of monitoring nodes. In Section VIII, we describe our first implementation of a VPE.

On top of CLEVER and the VPE a new layer has been included that provides some specific platform oriented applications able to deliver advanced services for contributors of the HSCloud (PaaS*). They include: smart monitoring, report services, data manipulation and aggregation. However, PaaS* functionalities are out of the scope of this work. The SaaS* exposes the interfaces to access such services. Different profiles are conceived in order to allow the HSCloud users to give different types of contributions. In fact, we can have actors belonging to Governments, having more departments relaying on differentiate contexts (Security Dep, Environment Dep, Military Dep, etc.) business enterprises, up to research centers. They can provide different contributions in HSCloud that we can classify as *passive, active and hybrid*. With the term *passive,* we mean actors that have a low level of trustiness on Clouds and can only achieve data

from Cloud. *Active* actors fully adopt and use Cloud technologies, transferring their software and hardware resources into the Cloud. *Hybrid* contributors are actors knowing Cloud, likely they use internally in their IT systems, but they partially use external Cloud technologies. In Section VII we provide a more detailed description of different actors in HSCloud.

On the left side of Figure 5, we have included the security block. One of the main concern in using Clouds is related to security and the problem increases in our scenario since it deals with Public National Security rules. XMPP has natively some fundamental security mechanisms for guaranteeing, the confidentiality of communications, message integrity verification among the parties and non repudiation of messages senders. In the *Identity* subsystem of XMPP users and robots (i.e. plug-in modules) must authenticate to their host server and messages from that user (robot) cannot be spoofed by simply replacing headers text in the message. In addition the identity verification can be obtained by requiring clients to have valid security certificates that confirm their identity; indeed certificates X509 v3 allow to accomplish the Strong Authentication. In XMPP the communication can be encrypted. There are two types of encryption with XMPP. The first is encryption performed during connection establishment and authentication. SASL, a standard used by several protocols, is used during this phase. Once the connection is established, all transmissions between the client (robot) and server are encrypted using TLS. These properties mean that XMPP is secure since both the connection establishment phase and the communication phase of the protocol are encrypted. Finally in our scenario we are considering the overall benefits in having the federation of Clouds. XMPP uses the same principle to secure client-to-server connections extending the server-to-server connections used for federation. Like client-to-server connections, server-to-server connections can be configured so that only encrypted connections, using SASL and TLS, are accepted.

VI. SensCLEVER components

The design of SensCLEVER is described in Figure 6. It is compliant with the Sensor Web Enablement (SWE) standard defined by the Open Geospatial Consortium. The OGC-SWE framework [32] has taken important early steps towards enabling the web-based discovery, exchanging and processing of sensor

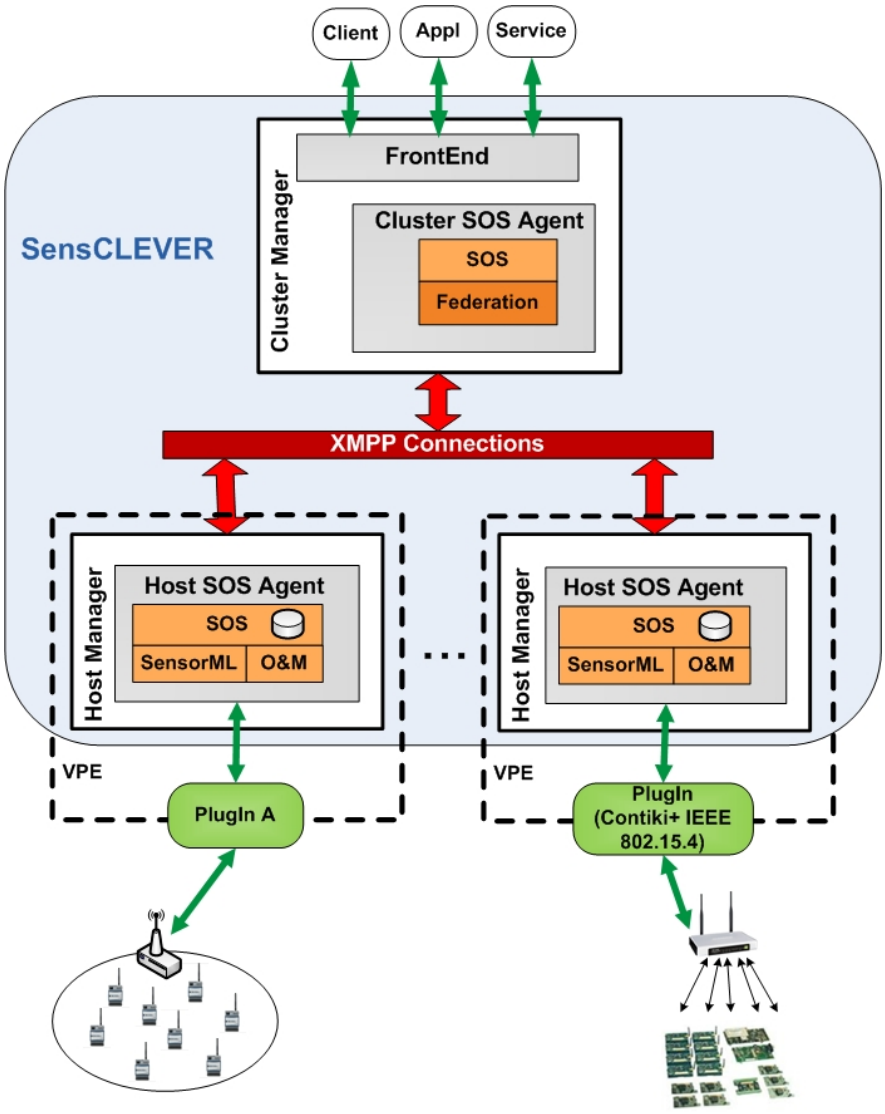

Fig. 6. The overall SensCLEVER architecture.

observations. Despite the OGC-SWE framework includes seven different standards, the SensCLEVER architecture refers only some of them. It implements the functionalities of the SOS (Sensor Observation Service), which specifies XML-based language and interface for requesting, filtering, and retrieving observations and sensor system information. It provides a mean to integrate data from heterogeneous sources in a standard format accessible from Cloud users by using a language and XML schemas for describing sensors systems and processes, that is the Sensor Model Language (SensorML), and models and XML Schema for encoding observations and measurements from a sensor network, that is the Observations

and Measurements (O&M) standard.

According with the hierarchical organization of CLEVER components discussed in Section III-B, we have two logical layers in the SensCLEVER architecture. At the higher level, in the CM, a Cluster SOS Agent (CSA) is activated to analyze clients requirements and gather data necessary to meet their expectations. The interaction with Cloud users, applications and/or other services is performed through a FrontEnd, which is common for all the agents in the CM. In the CSA, the SOS module is responsible for requesting and retrieving observations and sensor system information from all the HMs belonging to the cluster. In fact, to realize a very flexible system, many SIs deployed in the environment and managed from different administration can be aggregated in the Cloud. To merge data from all the HSAs, the CSA uses the *Federation* module. Organizations and administration that need to cooperate accomplishing a trust context, establish a federation according to a a *three-phase cross-Cloud federation model* [33]. Their infrastructures are exposed in HSCloud and the CSA Federation module manages the different data sources through polling policies, offering a seamless service to the FrontEnd.

The low-level services for the interaction of the system with the SIs are implemented in the HMs. We assume that an Host SOS Agent (HSA) is activated for each SI managed from an administration. The HSA has to support all the functionalities for the description of sensors and observations, setting of new observations and gathering of measurements from the SIs. To this aim, the SOS module in the HSA manages deployed sensors and retrieves sensor data and specifically observation data. It makes use of SensorML, for modeling sensors and sensor systems, and O&M standards, for modeling sensor observations. Each SI is virtualized in the Cloud through a specific HSA, but several HSAs can be placed in one or more HMs. At the same time, many CSAs can be present in the same CM. Each CSA offers a specific service according to a particular sensed data (e.g. traffic information, temperature measurements, forecast,...). The on-demand interaction allows the user to ask to the CSA a specific information. So, the user can choose the CSA of interest and perform on-demand queries according to its needs.

An HSA implements all the functionalities necessary to virtualize sensors, observations and measure-

ments. However, to make a prototype of the whole system, SensCLEVER has to interact with a sensing environment to gather data. Due to many available networking technologies for SIs, it is very important to guarantee a great adaptability of SensCLEVER to different technologies. To this aim, SensCLEVER implements a plug-in framework, where each plug-in implements calls of the APIs of the HSA, for the interaction with the Cloud, and knows the specific communication technologies of sensors, for the communication with the SI. For example, in our firs prototype that is described in detail in Section VIII, we have implemented a plug-in for a sensor network, where sensor nodes are equipped with IEEE 802.15.4 communication devices and work by using the Contiki operative system [34]. The HSA together with a specific plug-in forms the VPE, which represents the virtual element able to integrate a SI inside the CLEVER architecture, as shown in Figure 6.

VII. STOCKHOLDERS OF HSCLOUD

Users of HSCloud can have a *passive, active or hybrid* role, as specified in Section V. In particular, active actors may offer different types of contribution, such as functionalities to access sensed data (weather conditions, vehicle traffic congestions, airport air traffic, etc), other supply resources (i.e., computation, storage, etc) and/or specific applications for solving hard issues for managing dangerous (or catastrophic) situations. The type of contribution offered from a user in the HSCloud architecture depends on its type of accounting. HSCloud defines four types of users (see Figure 7: External users, Basic users, Advanced users and Admin. External users can just browse the web page of HSCloud, aiming to know its worth, potentiality and offered services. They can have information on the global behavior of the system through documents, reports and information pack available in the SaaS. They can not access tricky information or have knowledge of strategies necessary to support homeland security. To this aim, they do not need any authentication feature. On the contrary, basic users, advanced users and administrators need to authenticate themselves before acceding the system. Basic users can access SaaS and PaaS services in order to share information and algorithms and software prototypes. For example, basic users are private, military, governative or civilian organizations that develop specific strategies to fight back enemy attacks

and to provide help in case of disaster. Advanced users share, also, their physical infrastructure with the HSCloud system. Companies that monitor dangerous goods over multi-modal routes can be advanced users if they integrate their monitoring systems in HSCloud through the VPE module. They have access to PaaS services too. Also companies that are interest in provide hardware resources, such as clusters, multy-processor systems, storage solutions and so, can have an account as advanced users and give their contribution in HSCloud. Advanced users can offer their resources directly by integrating them in the HSCloud infrastructure or through other Cloud IaaS. Different Cloud providers and HSCloud federate each others and share their resources in order to provide a seamless infrastructure to PaaS and SaaS. Admin are administrators of the HSCloud architecture and they have access to all the resources and services of the system.

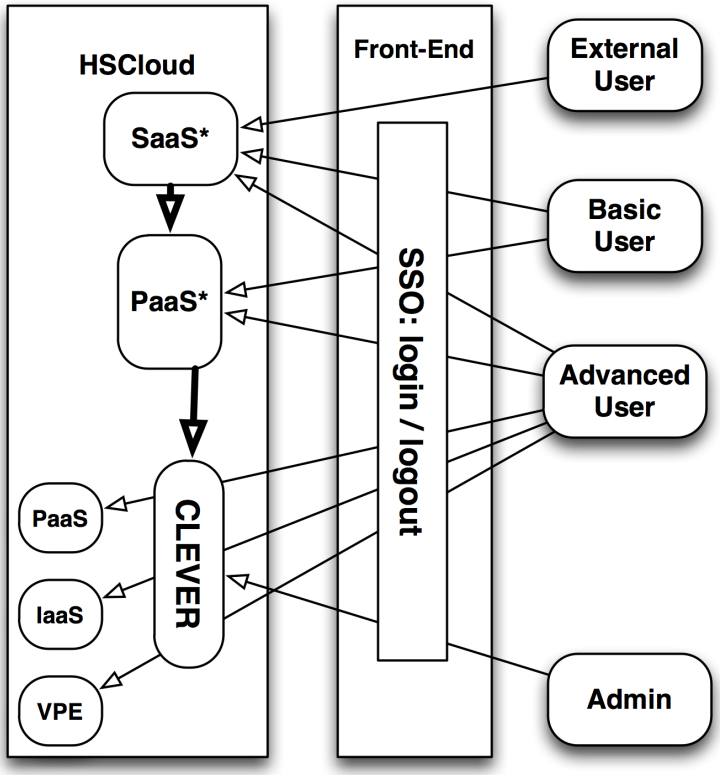

Fig. 7. Users access to HSCloud

External and Basic users belong respectively to the *passive* and *hybrid* topologies. In the early case they don't need to authenticate themself (External) for achieving simple services (SaaS*), wheres in latter,

after gaining the access, users can obtained some advanced services (SaaS* and PaaS*). The *Advanced Users* sub-category, is the *active* actors having their Cloud solutions and they are happy to use partner Clouds (federated Clouds) up to the use of public Clouds (i.e. deployment of more VMs on Amazon Cloud for executing algorithms that require an heavy computation).

VIII. System Prototype

We implemented a prototype of the VPE module to virtualize a sensor network in a federated Cloud environment. We used a NSLU2 (Network Storage Link for USB 2.0 Disk Drives), which is a Network-attached storage (NAS) device made by Linksys. Its main purpose is to serve as a network file server since it supports two USB ports along with an ethernet network connection. The added-values of such as device is its power consumption, in a few Watts it is possible to execute an entire Linux environment. The limited power consumption allow us to look at future scenarios in which VPEs may be deployed on territories. Indeed its power supply can easily guaranteed with batteries and photovoltaic cells. For our purposes, we have connected the NSLU2 device to a USB Flash memory disk to run a full Debian system ARM based (Lenny release) on it. Then, we have developed a minimal distribution of CLEVER and installed it on Debian, in order to configure the NSLU2 device as a Cloud node that is the VPE node. The second USB port is connected to the USB hub in order to increase NSLU2 connection capabilities. USB ports are used for connecting the VPE device to the Wireless sensor network. The WiFi, IEEE 802.15.4 and HSDPA/UMTS USB cards allow to use different technologies to access the SI. Thanks to these connectivity capabilities, the NSLU2 is able to work both as VPE and as the sink of the sensor network which monitors dangerous goods.

In our prototype, each node of the sensor network is a STMicroelectronics MB851 board, a device that has been designed as an IEEE 802.15.4 application-specific board for STM32W microcontrollers. The MB851 boards includes temperature (STLM20) and acceleration (LIS302DL) sensors. Furthermore, to implement automation services and application deployment in the SI, we have installed the Contiki open source operating system [34] on the STM32W devices. Contiki provides low-power networking

for resource constrained systems along with a development and simulation environment. It gives us the possibility to implement specific automation services on nodes in the SI, which we aim to develop as future work.

The main purpose of this prototype has been to develop a minimal version of *SensCLEVER*, in order to install it on the NSLU2. In SensCLEVER, we have maintained the basic functionalities of the XMPP communication and the Cloud federation management available in CLEVER in order to have the NSLU2 as part of the Cloud system. We have developed a runtime system to collect and represent sensor data as meta-data implementing the SWE specifications. In particular, we made reference to the SOS standard, which supports the functionalities for the description of sensors and observations, setting of new observations and gathering of measurements from a SI. To this aim, it makes use of the SensorML standard, for describing sensors, sensor systems, and sensed data, and the O&M standard, for modeling sensor observations. In the prototype, both the *Core* and *Transactional* profiles have been implemented. The first is responsible to produce information on the sensing system and on sensed data to clients. Such information are the result of specific queries on a local database into HMs. The mandatory operations of the Core profile we have implemented are:

- *GetCapabilities*: describes the abilities of the specific server implementation;
- *DescribeSensor*: retrieves detailed information about the sensors and processes generating the measurements;
- *GetObservation*: provides access to sensor observations and measurement data via a spatio-temporal query that can be filtered by phenomena.

The Transactional profile of SOS collects information from the SI, both in terms of properties of sensors entering the system and measurements performed by the sensors themselves. According to the SWE specifications, its mandatory functionalities are:

- *RegisterSensor*: allows to register new sensors available in the system;
- *InsertObservation*: allows to collect new observations coming from sensors.

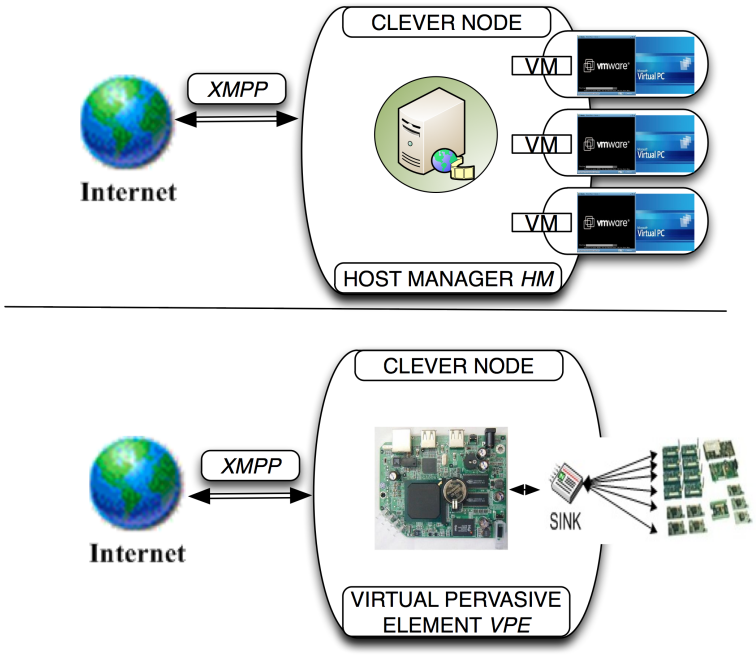

Fig. 8. CLEVER node: on the top side the original version aimed at Cloud Technology. On the bottom side the modified version able to support more type of Sensing Technology.

Figure 8 depicts how the original CLEVER node, conceived for Host Management has been modified in order to support different types of physical resources interactions. CLEVER represents a wrapper in which the overall sensing functionalities are mapped and translated into the XMPP communication channel. To test the architecture, we set up a simple environment where the temperature of our lab was continually measured thanks to the sensors mounted on board of the MB851 nodes. They send sensed data every 2 seconds to the sink (that is the MB851 board connected to the NSLU2 device). The sink transfers the data to the VPE (that is the NSLU2 device), which aggregates such information as the average temperature of the monitored area and formats it as a xml meta-data. In this way the VPE is able to provide an abstraction of the Cloud computing environment.

We remark the environment we designed and the prototype we developed have the opportunity to setup a system with an high level of security. The scenario we presented is quite complex and many actors need to interact each other in as secure way. XMPP is a standardized protocol in which many security capabilities are natively accomplished. All modules communicating in XMPP are strongly identified with X509 digital

certificates (private-public asymmetric keys). Whereas critical communications can be totally encrypted, using TLS/SSL protocols, for preventing fraudulent use of sensible data. One of the main concerns in adopting Cloud is represented by security and privacy, our solution tries to overcome these issues.

IX. Conclusions

In this paper we have presented HSCloud, a Cloud-based architecture able to deal with Homeland security problems. To show the main features of HSCloud, we have analyzed a specific use case, that is a TDGs scenario. We have discussed current solution for security in TDGs highlighting limitations and drawbacks. Then we have explained how Cloud computing can improve the world wide security against terrorist attacks. According with all these considerations, we have presented the main features of HSCloud, describing in detail its working core, that is a Cloud middleware called SensCLEVER. The virtualization techniques implemented in SensCLEVER for integrating sensing infrastructures in a Cloud environment guarantees efficient and secure services at the infrastructure, platform and application layers. Even if we have developed a first prototype of HSCloud, in the next future, we will improve it by including additional features related to the filtering of information according to users profile (see Section V). Furthermore, we intend to implement methods to execute remote control of the sensing infrastructure.

References

[1] L. Janczewski and A. Colarik, *Managerial guide for handling cyber-terrorism and information warfare*. Idea Group Publishing, 2005. [Online]. Available: http://books.google.com/books?id=vnjfMHWdKHQC

[2] T. Holt and B. Schell, *Corporate Hacking and Technology-Driven Crime: Social Dynamics and Implications*. Igi Global, 2010. [Online]. Available: http://books.google.com/books?id=LAIjG_OGuIMC

[3] D. Mortimer, "Homeland security public safety dive teams: how technology can help," in *OCEANS, 2005. Proceedings of MTS/IEEE*, 2005, pp. 178 – 183 Vol. 1.

[4] A. Koyuncugil and N. Ozgulbas, *Surveillance Technologies and Early Warning Systems: Data Mining Applications for Risk Detection*, ser. Premier Reference Source. Igi Global, 2010. [Online]. Available: http://books.google.com/books?id=l4Ir1nvyURIC

[5] C. Reddick, *Homeland security preparedness and information systems: strategies for managing public policy*. Information Science Reference, 2010. [Online]. Available: http://books.google.com/books?id=NaT_Fob6lBIC

[6] Warner Bros. Likes Facebook Rentals. http://online.wsj.com/article/SB10001424052748703386704576186913491751144.html.

[7] July 2011, Sensor Web Enablement. Available: http://www.opengeospatial.org/standards.

[8] M. Ortner, A. Nehorai, and A. Jeremic, "Biochemical transport modeling and bayesian source estimation in realistic environments," *Signal Processing, IEEE Transactions on*, vol. 55, no. 6, pp. 2520 –2532, june 2007.

[9] 2004-2006, MITRA: Monitoring and intervention for the transportation of dangerous goods. http://www.mitraproject.info/.

[10] 2009, SMARTFREIGHT project, FP7-216353. http://www.smartfreight.info//.

[11] F. Valente, G. Zacheo, P. Losito, and P. Camarda, "A telecommunications framework for real-time monitoring of dangerous goods transport," in *Intelligent Transport Systems Telecommunications,(ITST),2009 9th International Conference on*, October 2009, pp. 13 –18.

[12] Z. Yingjun, X. Shengwei, X. Peng, and W. Xinquan, "Shipping containers of dangerous goods condition monitoring system based on wireless sensor network," in *Networked Computing (INC), 2010 6th International Conference on*, may 2010, pp. 1 –3.

[13] I. Foster, Y. Zhao, I. Raicu, and S. Lu, "Cloud Computing and Grid Computing 360-Degree Compared," in *Grid Computing Environments Workshop, 2008. GCE '08*, 2008, pp. 1–10.

[14] NIST Cloud Computing Reference Architecture
http://www.nist.gov/customcf/get_pdf.cfm?pub_id=909505 December 2011.

[15] F. Tusa, M. Paone, M. Villari, and A. Puliafito., "CLEVER: A CLoud-Enabled Virtual EnviRonment," in *15th IEEE Symposium on Computers and CommunicationsS Computing and Communications, 2010. ISCC '10. Riccione*, June 2010.

[16] S. Pandey, W. Voorsluys, S. Niu, A. Khandoker, and R. Buyya, "An autonomic cloud environment for hosting ecg data analysis services," *Future Generation Computer Systems*, vol. 55, no. 6, June 2011.

[17] S. Alam, M. Chowdhury, and J. Noll, "Senaas: An event-driven sensor virtualization approach for internet of things cloud," in *Networked Embedded Systems for Enterprise Applications (NESEA), 2010 IEEE International Conference on*, nov. 2010, pp. 1 –6.

[18] V. Rajesh, J. Gnanasekar, R. Ponmagal, and P. Anbalagan, "Integration of wireless sensor network with cloud," in *Recent Trends in Information, Telecommunication and Computing (ITC), 2010 International Conference on*, march 2010, pp. 321 –323.

[19] S. Rusitschka, K. Eger, and C. Gerdes, "Smart grid data cloud: A model for utilizing cloud computing in the smart grid domain," in *Smart Grid Communications (SmartGridComm), 2010 First IEEE International Conference on*, oct. 2010, pp. 483 –488.

[20] K. Egami, S. Matsumoto, and M. Nakamura, "Ubiquitous cloud: Managing service resources for adaptive ubiquitous computing," in *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2011 IEEE International Conference on*, march 2011, pp. 123 –128.

[21] M. Yuriyama and T. Kushida, "Sensor-cloud infrastructure - physical sensor management with virtualized sensors on cloud computing," in *Network-Based Information Systems (NBiS), 2010 13th International Conference on*, sept. 2010, pp. 1 –8.

[22] The Extensible Messaging and Presence Protocol (XMPP) protocol:
http://tools.ietf.org/html/rfc3920.

[23] Ejabberd, the Erlang Jabber/XMPP daemon, http://www.ejabberd.im/ December 2011.

[24] Sedna, Native XML Database System:
http://modis.ispras.ru/sedna/ December 2011.

[25] The Apache Cassandra Project develops a highly scalable second-generation distributed database
http://cassandra.apache.org/.

[26] A. Kerrouche, J. Leighton, W. Boyle, Y. Gebremichael, T. Sun, K. Grattan, and B. Taljsten, "Strain measurement on a rail bridge loaded to failure using a fiber bragg grating-based distributed sensor system," *Sensors Journal, IEEE*, vol. 8, no. 12, pp. 2059 –2065, dec. 2008.

[27] Q. Li, T. Zhang, and Y. Yu, "Using cloud computing to process intensive floating car data for urban traffic surveillance," *Int. J. Geogr. Inf. Sci.*, vol. 25, pp. 1303–1322, August 2011. [Online]. Available: http://dx.doi.org/10.1080/13658816.2011.577746

[28] N. Sahli, N. Jabeur, and M. Badra, "Agent-based approach to plan sensors relocation in a virtual geographic environment," in *New Technologies, Mobility and Security (NTMS), 2011 4th IFIP International Conference on*, feb. 2011, pp. 1 –5.

[29] C.-T. Yang, L.-T. Chen, W.-L. Chou, and K.-C. Wang, "Implementation of a medical image file accessing system on cloud computing," in *Computational Science and Engineering (CSE), 2010 IEEE 13th International Conference on*, dec. 2010, pp. 321 –326.

[30] J. Wang, B. Huang, A. Huang, and M. D. Goldberg, "Parallel computation of the weather research and forecast (wrf) wdm5 cloud microphysics on a many-core gpu," in *Parallel and Distributed Systems (ICPADS), 2011 IEEE 17th International Conference on*, dec. 2011, pp. 1032 –1037.

[31] M. Saini, W. Xiangyu, P. Atrey, and M. Kankanhalli, "Dynamic workload assignment in video surveillance systems," in *Multimedia and Expo (ICME), 2011 IEEE International Conference on*, july 2011, pp. 1 –6.

[32] C. Reed, M. Botts, J. Davidson, and G. Percivall, "OGC Sensor Web Enablement: Overview and High Level Architecture," *IEEE Autotestcon*, pp. 372–380, 2007.

[33] A. Celesti, F. Tusa, M. Villari, and A. Puliafito, "How to Enhance Cloud Architectures to Enable Cross-Federation," in *IEEE 3rd International Conference on Cloud Computing (CLOUD'10)*, Miami, FL, 5-10 July 2010, pp. 337 –345.

[34] A. Dunkels, B. Grnvall, and T. Voigt, "Contiki - a lightweight and flexible operating system for tiny networked sensors," in *Proceedings of the 29th Annual IEEE International Conference on Local Computer Networks (LCN '04)*, 2004.