Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 177 (2020) 450–455

Procedia Computer Science

www.elsevier.com/locate/procedia

The 6th International workshop on Big Data and Networks Technologies (BDNT 2020)
November 2-5, 2020, Madeira, Portugal

A new Collaborative Platform for Research in Smart Farming

Olivier Debauche[a,b,*], Saïd Mahmoudi[a], Pierre Manneback[a], Jérôme Bindelle[c], Frédéric Lebeau[d]

[a] *University of Mons, Faculty of Engineering - ILIA / Infortech, Place du parc 20, Mons 7000, Belgium*
[b] *University of Liège - GxABT, Terra, Passage des déportés 2, Gembloux 5030, Belgium*
[c] *University of Liège - GxABT, Precision Livestock and Nutrition, Passage des déportés 2, Gembloux 5030, Belgium*
[d] *University of Liège - GxABT, BioDynE, Passage des déportés 2, Gembloux 5030, Belgium*

Abstract

The sharing of experimental dataset and benchmark between researchers is particularly import for the cross validation of models and algorithms that have been developed. In practice the sharing of data is difficult because certain legislations must be respected such as protecting privacy, copyrights, information confidentiality... etc. In this paper, we propose a scientific collaborative platform to share, exchange, and transfer data, applications, and models between researchers. This is only possible if we implement a high-level of encryption and security. The choice of encryption algorithms is crucial to ensure a long-term high level of protection against theft, willful alteration, and falsification. Our platform has been experimented within a community of researchers interested in cow behavior analysis based on Inertial Movement Unit and GPS data acquired by iPhone at high frequency (100 Hz).

Keywords: Animal Behavior; Behavior Analysis; Sharing Platform

1. Introduction

Nowadays, in the context of open data and open science, the sharing data is becoming crucial for researchers; nevertheless, we notice, in fact, that there are many difficulties involved regarding the control which the owners must apply in terms of data, protection against hacking, falsification, and theft. Moreover, it also essential to guarantee the preservation of data integrity. Several communities of researchers have developed platforms to share, archive, publish data, and application such as Galaxy project in genetics [3, 7, 8], Globus Platform [2, 1, 4], iPlant collaborative [9], Cancer Imaging Archive (TCIA) [5] new needs of their respective communities. These needs in term of collaborative

* Corresponding author. Tel.: +32-65-374-059 ; fax: +32-71-140-095.
E-mail address: olivier.debauche@umons.ac.be

1877-0509
10.1016/j.procs.2020.10.061

platform, in smart farming, are still weakly addressed, particularly the sharing data for review with colleagues or paper reviewers, or the use as input in other analysis pipeline to produce new derived data. The reproducibility is also an important aspect which can be attempts only if the traceability is ensured between raw, intermediate, derived data. Moreover, the use of a short persistent identifier which stay associate with data independently of its location and the association of domain specific metadata facilitate the discovery of it.

In this paper, we propose an open source platform to address the needs in Smart Farming, as well as research in terms of sharing and exchange of benchmark dataset, models, and applications. This platform is built based upon our own needs on two use cases: the first concern the farm animal' behaviors and the second focusing on the digital phenotyping in containers.

The remaining of this paper is organized as follow: In section 2, we analyze existing works achieved in other disciplines in order to identify key factors of theses platforms or infrastructures. Then, in section 3, we identify the main needs in smart farming research and then, on this basis, we conceptualize and implement our raw data, dataset benchmark, and sharing applications platform. Afterwards, in section 4, we test our platform in production. Finally, in section 5, we conclude and describe future developments.

2. Related Works

This section is organized following three parts: In the first subsection, important conceptualization aspects and issues must be clearly identified in order to addressed them. The second part focuses on existing platforms developed by the community used to share research sensitive data. Finally, the third part studies how to ensure an optimal security and privacy to sensitive data.

2.1. Platform Conceptualization

Dong et al. [6]have highlighted four factors aspects which must taken into account to guarantee the safety: (1) The upload of data between the owner and the platform; (2) Security problem a storage and computing at platform level; (3) Internal issues presented in cloud platform; (4) Secure data destruction.

Niu et al. [10] have proposed a framework to secure a big data platform using cloud computing restful on 8 aspects of security: network, server, storage, virtualization platform, platform software, application, and management.

We develop our platform on basis of this framework and four security factors.

2.2. Existing research platforms

Tremendous number of platforms exist in various research domains responding to specific needs. We describe the major contributions in the subsequent paragraphs.

Galaxy [3, 7, 8] is an accessible and an open-web platform to address, in genomic research, the lack of computational tools easily includable in an analysis chain and runnable without previous programming experience. Moreover, Galaxy is based on the concept of Reproducible Research System (RSS) providing an environment for performing and recording computational analysis.

Globus [2, 1, 4] is platform-as-a-service based on REST API providing identity, profile, group management, , synchronization, and sharing data operations. It also provides robust data publication capabilities such as persistent identifier, data access management, and metadata indexation for the discovery.

The **iPlant** Collaborative [9] is a cyber infrastructure for plant biology designed to help researchers to use easily and efficiently tools and data, gain access to High-Performance Computing (HPC) and provide interoperable software analysis.

The **Cancer Imaging Archive** [5] is an open-source, open-access information resource to develop research using advanced medical imaging of cancer. All proposal must be approved before submission on the Internet.

2.3. Best ways to secure sensitive data

A firewall is placed between the user and the cluster. UFW is a command line tool to manage the firewall implemented by default in Ubuntu. This firewall has no control of the activity inside the cluster, so it is necessary to

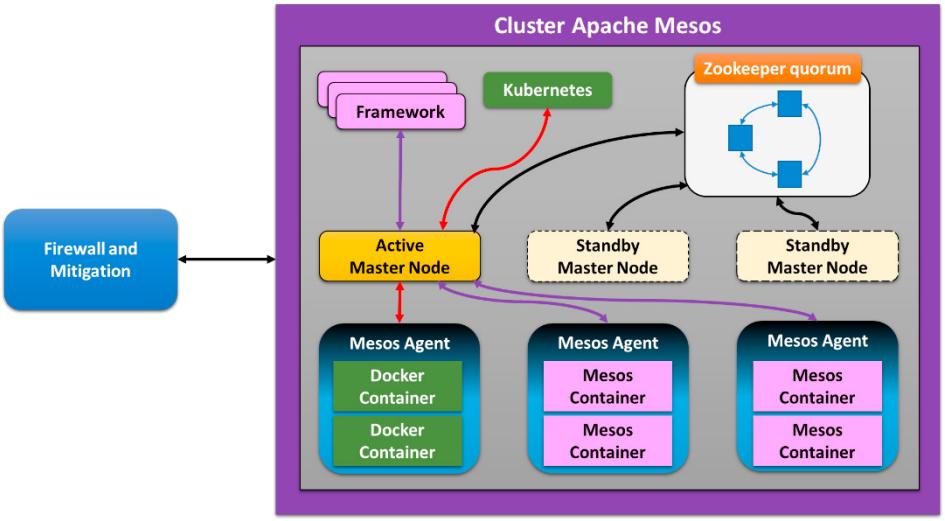

Fig. 1. High level of our architecture.

implement complementary software to monitor the activity inside the cluster. Snort [12] is an open-source network intrusion detection and prevention system capable of real-time traffic analysis, packet logging and allowing to detect emerging threats.

Paul et al. [11] have developed a testbed to generate internal and external attacks, and dataset creation for big data cluster. They used hing3 tools to generate attacks such as DOS, ICMP Flooding, TestDFSIO to generate an important I/O operation, and YARN REST API to generate an important network activity.

3. Our proposition

Our architecture is built around Apache Mesos, an open-source cluster manager which allows resources allocations inside the cluster (Fig. 1). Mesos uses a master node and two standby master nodes linked with a quorum of zookeeper that elects a standby master node to become the new master node in case of failure of the active master node. The master node distributes requests coming from different frameworks schedulers and Kubernetes that we use to orchestrate the management of docker containers. When a framework request is sent to the active master node, it attributes this request to a worker node having available resources that are necessary for the execution of this task. If the executor of the framework is absent on the work node. It is therefore installed and then the task is executed in a Mesos container. In the flip side, if a Kubernetes request arrives to the master node, it is redirected to a worker node using docker containerization to execute the Kubernetes request.

In this architecture, Mesos is used to host on one hand services of the platform, and on other hand models and applications developed by researchers. The use of containerization allows us to propose different versions of the same model or application and hence retest old versions with new datasets and compare results with the new version.

In the upcoming paragraphs, we describe milestones of our sharing, publishing, and hosting platform.

3.1. Platform security

Security aspects are important and must be implemented at different levels to be secured against any theft, voluntary or accidental deterioration or falsification of data and hacking attempt. We have implemented Snort as intrusion detection system and prevention of emerging threats. Moreover, we use also Clamav is an open source antivirus to detect threats such as trojans, malware, viruses, viz. Additionally, Fail2ban provides a protection to SSH against brute force intrusion attempts.

3.2. Data management

The deposit of data is achieved by the filling of a form and afterwards the dataset and companion files are uploaded by means of secure file protocol (SFTP) on a storage created specifically for this dataset. Files upload are then identified and associate with a file category (metadata, annotation, license, etc.) and their privacy access (private, public, or shared) is specified. Dataset files are crypted with Advanced Encryption Standard (AES) 256-CBC algorithm provided with Open SSL library with an individual symmetric key for each file. All symmetric keys and associate nonce of the user are encrypted with XChaCha20Poly1305; a hardened version of ChaCha20Poly1305 against nonce misuse provided by the Sodium library. In public mode, proposed dataset must be firstly validated by an administrator before publishing while in private and shared mode, in this scenario, dataset is directly available.

3.3. Applications management

Users of the platform have two possibilities to propose their model and/or application. They can propose a download link or deploy publicly their model or application on the platform using docker containerization, containerization system of Apache Mesos or virtualization. Apache Mesos is used to host and fined grained manage dynamically cluster resources. As for dataset management, public deployment must previously validate by an administrator to be achieved.

3.4. Description and publishing

Dataset is described by means of metadata which allow the discovery and the indexing in catalog. It is also referenced of Minid such as Digital Object Identifier (DOI), Archival Resource Key (ARK) or handle. While applications are proposed in the form of docker images that can be downloaded or deployed directly on Apache Mesos which. This latter has virtualization capability allowing to host all types of application.

3.5. Access management

The access to public content is free and does not require to register and log in beforehand. However, access to restricted content requires a valid user account and access rights to private content. Shared content (dataset or application) are accessible by means of a fixed IP and a password. This access mode allows to provide easily access to a team without having to manage access rights individually.

4. Experimentation and Discussion

Our platform proposition has been tested on the animals' behavior research and deployed on http://cow.engineering.

Figure 2 shows the pipeline used to publish a dataset. The access management is used to authenticate a user (and eventually create a new one). An identified user can propose a new dataset by filling in a description form. Then a storage is created, and ACL are applied. Afterwards, a specific SFTP access associated with the create storage is activated. It allows to transfer at same time data and companion files such as image, license file, annotation file, etc. In the next step all files uploaded are listed and the user identify the category of each ones. Metadata are requested to describe the dataset. The publication system get metadata and generate the dataset before publishing in the catalog.

Figure 3 shows the pipeline used to publish or deploy applications on the platform. The deposit or the deployment of an application begins by getting credentials and by continuing with the creation of data storage on which it is possible to upload a docker file from a SFTP access or import it from an external repository. Its integrity is evaluated, and the deployment is tested and the descriptive metadata of the application are then required. These metadata are retrieved by publishing and/or deployment module.

The deployment module use metadata to deploy all needs components and install the model or the application. Afterwards, ACL are applied on the instance and an URL is generated to access to this one. The URL and metadata are published in the catalog.

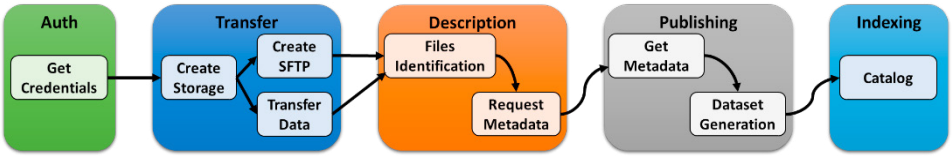

Fig. 2. Dataset publishing pipeline.

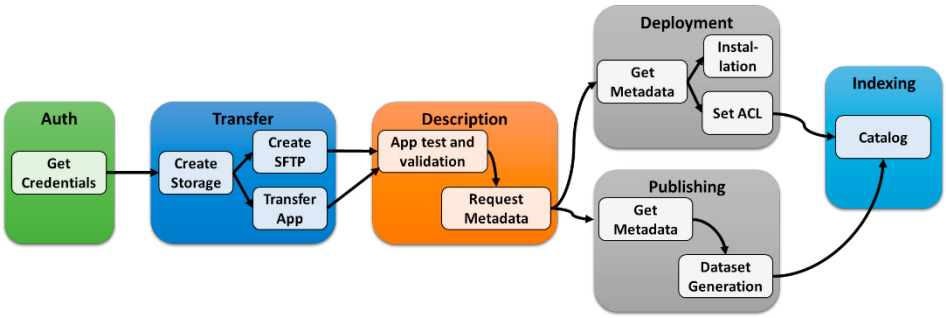

Fig. 3. Application publishing and deployment pipeline.

The publishing module exploit meta data to generate a dataset bundle which will be indexed in the catalog.

The infrastructure has been tested with 4 research teams located in Brazil, Belgium, Ecuador, and Colombia that have exchanged dataset of measures achieved with Inertia Movement Unit (IMU) of iPhone acquiring. This data is logged at 100Hz and produce files with a size of 1.8 Gb for 24 h. This data is completed by video sequences used to correlate IMU measures with tagged behavior identified on video. In addition, behavior data and video are completed with georeferenced images taken during drones' flies achieved punctually to evaluate the grass stock available for cows' grazing. Our platform has been tested with several TB of various data. This sharing and exchange of datasets have allowed to develop more robust models built on more varied dataset instead of local models whose applicability is much more restricted.

5. Conclusion and Perspective

Our platform is emerged from our own needs in matter of data, models and applications sharing for research in Smart Farming specially to develop and train specific Artificial Intelligence algorithms, to provide geospatial analysis, and to build new statistical or mathematical models.

The use of docker containerization in association with a Mesos cluster allows us to host multiple versions of the same model or applications and ease the comparison between versions of a same model / application for different datasets. In addition, our approach allows to develop and validate models / applications based on wide and various datasets.

In future works, we will propose to monetize their deployed models and applications at usage or in hours of use.

Acknowledgements

The authors would like to express their gratitude to Mrs Meryem Elmoulat for English editing of this paper.

References

[1] Ananthakrishnan, R., Blaiszik, B., Chard, K., Chard, R., McCollam, B., Pruyne, J., Rosen, S., Tuecke, S., Foster, I., 2018. Globus platform services for data publication, in: Proceedings of the Practice and Experience on Advanced Research Computing, pp. 1–7.

[2] Ananthakrishnan, R., Chard, K., Foster, I., Tuecke, S., 2015. Globus platform-as-a-service for collaborative science applications. Concurrency and Computation: Practice and Experience 27, 290–305.
[3] Blankenberg, D., Kuster, G.V., Coraor, N., Ananda, G., Lazarus, R., Mangan, M., Nekrutenko, A., Taylor, J., 2010. Galaxy: a web-based genome analysis tool for experimentalists. Current protocols in molecular biology 89, 19–10.
[4] Chard, K., Pruyne, J., Blaiszik, B., Ananthakrishnan, R., Tuecke, S., Foster, I., 2015. Globus data publication as a service: Lowering barriers to reproducible science, in: 2015 IEEE 11th International Conference on e-Science, IEEE. pp. 401–410.
[5] Clark, K., Vendt, B., Smith, K., Freymann, J., Kirby, J., Koppel, P., Moore, S., Phillips, S., Maffitt, D., Pringle, M., et al., 2013. The cancer imaging archive (tcia): maintaining and operating a public information repository. Journal of digital imaging 26, 1045–1057.
[6] Dong, X., Li, R., He, H., Zhou, W., Xue, Z., Wu, H., 2015. Secure sensitive data sharing on a big data platform. Tsinghua science and technology 20, 72–80.
[7] Giardine, B., Riemer, C., Hardison, R.C., Burhans, R., Elnitski, L., Shah, P., Zhang, Y., Blankenberg, D., Albert, I., Taylor, J., et al., 2005. Galaxy: a platform for interactive large-scale genome analysis. Genome research 15, 1451–1455.
[8] Goecks, J., Nekrutenko, A., Taylor, J., Team, G., et al., 2010. Galaxy: a comprehensive approach for supporting accessible, reproducible, and transparent computational research in the life sciences. Genome biology 11, R86.
[9] Goff, S.A., Vaughn, M., McKay, S., Lyons, E., Stapleton, A.E., Gessler, D., Matasci, N., Wang, L., Hanlon, M., Lenards, A., et al., 2011. The iplant collaborative: cyberinfrastructure for plant biology. Frontiers in plant science 2, 34.
[10] Niu, X., Zhao, Y., 2019. Research on big data platform security based on cloud computing, in: International Conference on Security and Privacy in New Computing Environments, Springer. pp. 38–45.
[11] Paul, S., Saha, S., Goswami, R., 2020. Testbeds, attacks, and dataset generation for big data cluster: A system application for big data platform security analysis, in: Progress in Computing, Analytics and Networking. Springer, pp. 545–554.
[12] Roesch, M., et al., 1999. Snort: Lightweight intrusion detection for networks., in: Lisa, pp. 229–238.