H2020 – ICT-13-2018-2019

Machine Learning to Augment Shared Knowledge in Federated Privacy-Preserving Scenarios (MUSKETEER)

Grant No 824988

D8.6 Evaluation and impact assessment

November 21

Imprint

Contractual Date of Delivery to the EC: **31 November 2021**

Author(s): **Antoine Garnier (IDSA), Laura Kasterke (IDSA), Angel Navia Vázquez (UC3M), Jesús Cid Sueiro (UC3M), Manuel Vázquez López (UC3M), Luis Muñoz-González (IMP), Roberto Díaz Morales (TREE)**

Participant(s): **Chiara Napione (COMAU), Giacomo Fecondo (STELLANTIS), Mark Purcell (IBM), Joao Correia (B3D), Petros Papachristou (HYGEIA), Christina Kotsiopoulou (HYGEIA)**

Reviewer(s): **Susanna Bonura (ENG), Emil Lupu (IMP)**

Project: **Machine learning to augment shared knowledge in federated privacy-preserving scenarios (MUSKETEER)**

Work package: **WP8**

Dissemination level: Public

Version: **1.0**

Contact: **Antoine Garnier – antoine.garnier@internationaldataspaces.org**

Website: **www.MUSKETEER.eu**

Legal disclaimer

The project Machine Learning to Augment Shared Knowledge in Federated Privacy-Preserving Scenarios (MUSKETEER) has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 824988. The sole responsibility for the content of this publication lies with the authors.

Copyright

Executive Summary

The document provides a full-scale evaluation and impact assessment of the MUSKETEER Industrial Data Platform. It integrates the evaluation of technical aspects of the platform (efficiency and quality gains resulting from the MUSKETEER solution, with specific respect to: privacy and confidentiality of data, scalability, compute efficiency (amount of data processed, number of queries handled at any given time, power consumption, etc.)) and its impact from the developer and end users' point of view.

Document History

Version	Date	Status	Author	Comment
1	31 Aug 2021	For internal review	IDSA	First draft TOC
2	15 Oct 2021	For internal review	IDSA, UC3M, IMP, TREE	Version 0.5
3	05 Nov 2021	For review	IDSA	Version 0.9
4	29 Nov 2021	Ready for submission	IDSA	Version 1.0
5	29 Nov 2021	Final review	IBM	Final

Table of Contents

List of Figures

List of Tables

List of Acronyms and Abbreviations

Abbreviation	Definition
3D	Three-Dimensional
AFA	Adaptive Federated Averaging
AI	Artificial Intelligence
AMQP	Advanced Message Queuing Protocol
B3D	Biotronics 3D
CA	Certification Authority
CN	Crypto Node
DICOM	Digital Imaging and Communications in Medicine
DoA	Description of Actions
DPIA	Data Protection Impact Assessment
DVE	Data Value Extraction
EU	European Union
FBSVM	Fuzzy Binary Support Vector Machine
FL	Federated Learning
FML	Fibre-Metal Laminate
GA	Grant Agreement
GDPR	General Data Protection Regulation
HE	Homomorphic Encryption
HL7	Health Level Seven
HTTPS	Hypertext Transfer Protocol Secure
HYGEIA	Hygeia Hospital
IBM	International Business Machines Corporation
ICT	Information and Communication Technology
ID	Identification Document
IDSA	International Data Spaces Association
IMP	Imperial College London
IoT	Internet of Things
IT	Information Technology
JSON	JavaScript Object Notation
KPI	Key Performance Indicator
KUL	Katholieke Universiteit Leuven
ML	Machine Learning
MMLL	MUSKETEER Machine Learning Libraries
MN	Master Node
MNIST	Modified National Institute of Standards and Technology
MQ	Message Queuing

Abbreviation	Definition
oT	(Internet) of Things
POM	Privacy Operations Modes
RAM	Reference Architecture Model
RMD	Random Matrix Disguise
SDP	Secure Dot Product
SMC	Secure Multi-party Computation
SME	Small and Medium-sized Enterprises
SVM	Support Vector Machine
TRL	Technology Readiness Level
UK	United Kingdom
US	United States
WN	Worker Node
WP	Work Package

1 Introduction

1.1 Purpose

This document aims to present a complete evaluation and impact assessment of the MUSKETEER platform. As mentioned in the task description T8.3 “Evaluation and impact assessment”, the evaluation should consider the successive iterations of the platform and shows how it has improved in its different aspects. This concerns the efficiency and quality gains resulting from the MUSKETEER solution, privacy and confidentiality of data, scalability, computational efficiency (amount of data processed, number of queries handled at any given time, power consumption, etc.). As mentioned in the deliverable description, D8.6, this work should provide quantitative information, and also identify areas for further improvement and investment. Therefore, the partners providing their assessment for the different aspects of the platform, have also added in the deliverable their recommendations for future improvements. This includes recommendations for possible further development and investment but also ideas for further research and emerging research questions.

The task attached to deliverable T8.3 includes a second objective. The impact assessment conducted over this deliverable should provide core material for dissemination activities. In this respect, and knowing the partners started their assessment in other deliverables, this document focuses on recommendation for improvements to the platform, but also long-term views on possible transformations of the industry in light of the MUSKETEER project.

1.2 Related Documents

As an impact assessment of the complete platform, this deliverable deals with a lot of findings from work previously done in the project. For that reason, it has many relationships with other project deliverables.

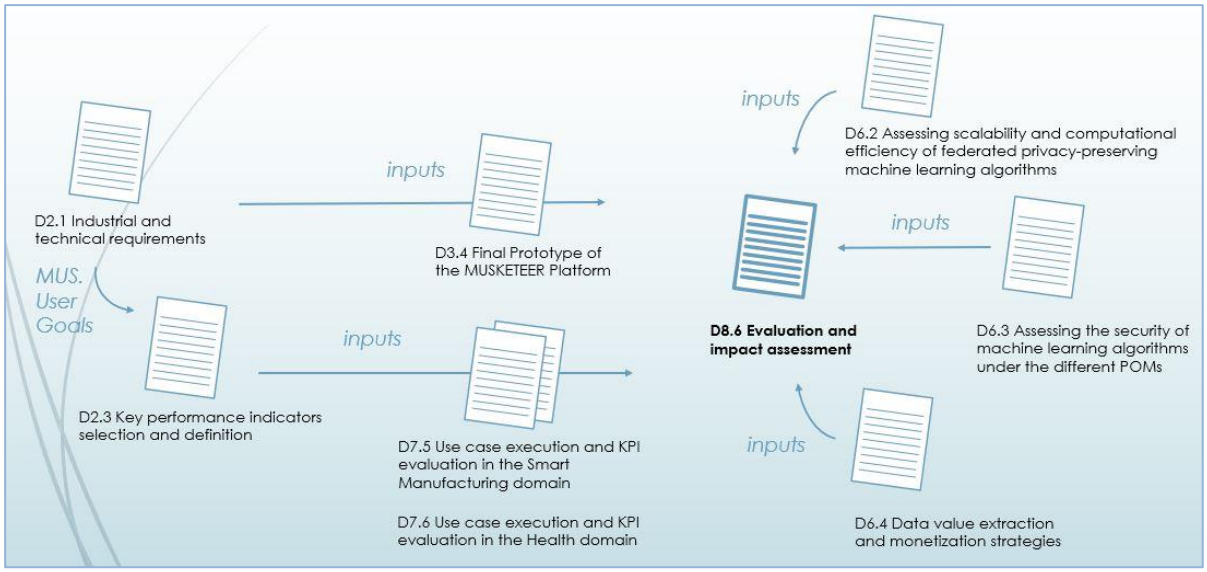


Figure 1. Relationships between D8.6 "Evaluation and impact assessment" and relevant documents in the project

As presented in Figure 1, many documents include relevant information related to the assessment of the platform. On one hand side, it includes inputs dating back to the beginning of the project with the requirements from D2.1 “Industrial and technical requirements” (M4) that included a list of functional, non-functional, and technical requirements and a first analysis of the MUSKETEER user goals in D2.3 “Key performance indicators selection and definition” (M6). The list of requirements was analysed, and its completion eventually assessed in D3.4 “Final Prototype of the MUSKETEER Platform” (M26) while the list of metrics defined under D2.3 is currently being used in D7.5 “Use case execution and KPI evaluation in the Smart Manufacturing domain” (M36) and D7.6 “Use case execution and KPI evaluation in the health domain” (M36). The rest of the inputs for this document come from WP6 “Assessment of data quality, scalability, computational efficiency and security” specifically targeting the assessment of different computational aspects of the platform. D6.2 “Assessing scalability and computational efficiency of federated privacy-preserving machine learning algorithms” (M30) provides information about the scalability and computational efficiency, two aspects directly mentioned in the task description. D6.3 “Assessing the security of machine learning algorithms under the different POMs” (M36) will provide information about the security of the platform, another aspect mentioned in the task description. Finally, D6.4 “Data value extraction and monetization strategies” (M30) provides information about the topic “data monetization”. Given its many relationships with other documents, the deliverable will synthesize the different assessments of the platform, from technology to business. It relies on the work done in common and on the feedback from the partners.

1.3 Document Structure

After a brief introduction, this document follows with a chapter about the MUSKETEER objectives. It intends to remind of the MUSKETEER objectives as per the Description of Actions (DoA) and gives some background, explanation and context for the following chapters. This deliverable is then divided in two main chapters: on one hand side chapter 3 provides a platform perspective and on the other hand, chapter 4 provides the use case perspective. In this way, chapter 3 especially tackles the assessment of technical aspects of the project (security and robustness, privacy and confidentiality, data value extraction and monetization, computational efficiency assessment and scalability assessment). Chapter 4 focuses on the assessment of the business aspects, mainly through the assessment provided by the end-users of the project from the smart manufacturing use case and the health domain use case. Chapter 5 concludes the deliverable.

2 MUSKETEER Objectives

Based on the ICT-13-2018-2019 call description, the MUSKETEER objectives are twofold: on one side it includes technical objectives about advancing "the state of the art in the scalability and computational efficiency of methods for securing desired levels of privacy of personal data and/or confidentiality of commercial data, particularly when they are combined from multiple owners", in "privacy/confidentiality threat models and/or [for] incentive models for the sharing of data assets" and business and societal-related objectives:

- Personal data protection is improved, and compliance with the General Data Protection Regulation (and other relevant legislation) is made easier for economic operators
- Citizens' trust is improved as privacy-aware transparency and control features are increasingly streamlined across data platforms and Big Data applications.
- Better value-creation from personal and proprietary/industrial data.
- 20% annual increase in the number of data provider organisations in the personal and industrial data platforms
- 30% annual increase in the number of data user/buyer organisations using industrial data platforms
- 50% annual increase in number of users (data subjects) in the personal data platforms
- 20% annual increase in volume of business (turnover) channelled through the platforms

These objectives are directly translated in the MUSKETEER proposal as defined in the DoA:

O1. Machine Learning over a high variety of different privacy-preserving scenarios.

O1.1. Definition of several Privacy Operation Modes (POMs) to provide compliance with the legal and confidentiality restrictions of most industrial scenarios, so to get scalable architecture design (D3.1-2) and prototype (D3.3-4) with some Privacy Preserving Modes. In order to assess such result, it will be expected a distributed efficiency (speedup/number of users) superior to 0.8 (this result will be included in report D6.2), while preserving privacy.

O1.2. Creating predictive models without directly exposing them to the data consumers (training data remains in the installations of data providers), so to obtain Federated data normalization and alignment algorithms (D4.2.1) and some of the training procedures in (D4.3.1-2, D4.4.1-2). The federated training will achieve comparable accuracy to the traditional local computing (decentralization will not affect the accuracy; this result will be included in report D6.2).

O1.3. Correct combination of different concepts of federated machine learning, differential privacy, homomorphic encryption, secure multiparty computation and distributed computing to improve the scalability of machine learning algorithms over every POM. The output will be some of the training procedures in the ML library (D4.3.1-2, D4.4.1-2), and it will be evaluated by expecting faster than current Secure Multiparty Computation (SMC) privacy-preserving alternatives: PySyft, SecureML (this result will be included in report D6.2).

O1.4. Complete library of algorithms, having algorithms of different complexity levels (D4.3.1-2, D4.4.1-2). The number of implemented algorithms has to be significant. In supervised learning, the library will contain at least a classification and a regression alternative of linear models, kernel methods, trees and deep neural networks. It will also include one unsupervised technique for clustering and data decomposition.

O2. Providing robustness against external and internal threats.

O2.1. Providing analysis and requirements for secure federated machine learning algorithms. We will consider vulnerabilities during training and at runtime, including the possibility of abuse from the users of the platform. Thus, the goal is having a Threat model and taxonomy of the possible attacks and weaknesses for federated machine learning algorithms (D5.1).

O2.2. The POMs will be designed to allow a secure information exchange among the platform users, so to have an architecture flexible enough to handle the 8 POMs.

O2.3. Including defensive mechanisms for the federated machine learning algorithms against poisoning and evasion attacks by detecting and mitigating the effect of such attacks (D5.4 and D5.5). The defensive mechanisms will be capable of reducing the effect of poisoning (for

reasonable levels of data poisoning, e.g. less than 20% of poisoning in the training dataset) and evasion attacks, compared with unsecured federated machine learning algorithms.

O2.4. Providing mechanisms to detect and mitigate the effect of abusive users in the platform trying to compromise the learning process, so to get algorithms to detect and characterize malicious users colluding to compromise the learning algorithms in the platform (D5.6, D5.7). The defensive mechanisms will be capable of mitigating colluding users' attacks for reasonable scenarios (e.g. assuming a maximum of 20% of malicious users colluding to manipulate the platform), compared with unsecured federated machine learning algorithms.

O2.5. Providing strong cyber-security against external data hackers by integrating robust and secure access and transportation protocols into the communication layers. Architecture with cyber-security mechanisms fully implemented with zero filtration of data in the validation process by surpassing.

O2.6. Developing a framework to test the security of federated machine learning against data poisoning, evasion attacks, and users' colluding attacks. This testing framework will enable the design of more secure learning algorithms and will provide an estimation of the worst-case performance of the system against different attacks with different levels of strength. They will be delivered a report and an implementation with the testing methodology to assess the security of the machine learning algorithms used in the platform against poisoning and evasion attacks (D6.3), and to evaluate the robustness of the system against malicious users (D5.6, D5.7).

O3. Enhancement of the Data Economy

O3.1. Enhancing data providers to share their datasets thanks to the ability of creating predictive models without explicitly giving their datasets (using the FML concept), thus avoiding any possibility of personal/private information robbery (Algorithmic (D4.1) and architectural (D3.1) design). Eight different privacy operation modes will be implemented to cover the different privacy needs given in industry.

O3.2. Allowing to measure the impact of every data owner on the accuracy of the predictive models, thus allowing to monetize their contributions as a function of their real data value (Data value extraction and monetization strategies (D6.4). Different data value estimation methods (one for every POM/algorithm) will be delivered.

O3.3. European SMEs involvement (D8.5, D8.6), through more than 10 industrial diffusion events, 3 workgroups attendance, 5 workshops.

O4. Providing a standardized and extensible architecture.

O4.1. Integration with other European initiatives related with data platform, by granting the compliance with the Industrial Data Space Association reference architecture.

O4.2. Allowing interoperability with Big Data frameworks by providing portability mechanisms to load and export the predictive models from/to other platforms. The predictive models will be obtained with the ML library (D4.3.1-2, D4.4.1-2), so MUSKETEER will be capable to export the predictive models to be loaded at least into the most extended ML libraries.

O4.3. Fostering the creation of a community of developers and researchers that can extend the platform with new algorithms and attack detection mechanisms after the life of the project. Special focus will be given to Open Source Licenses. Reports with scientific dissemination (D8.1, D8.2), reports with community engagement and technology transfer (D8.2, D8.4) will be delivered.

O4.4. Fast deployment, installation and use. Architecture based on containers will ensure that applications deploy quickly, reliably, and consistently regardless of deployment environment. Software component accessible in open source repositories.

O5. Industrial demonstration of the technology advances in operational environment (TRL6)

O5.1. Demonstration that MUSKETEER will be applicable on different privacy application domains will drive the project research and developments. They will be delivered: Report with privacy confidentiality impact assessment (D2.4, D2.5), Data ownership and governance recommendations (D2.6), and correct application of MUSKETEER into two different sectors (smart manufacturing and health) will be ensured.

O5.2. Continuous monitoring and feedback process during the whole project using realistic conditions to ensure demonstration of ready-to-use technology at the end of the project (report with the technical and legal requirements (D2.1, D2.2), and industrial KPI definition (D2.3)). There will be at least one monthly meeting involving technical partners and use cases to facilitate communication and report potential problems that may arise.

O5.3. Benchmark execution, evaluation and impact assessment to ensure that the innovative technology is applicable in a wide variety of problems (report with the evaluation of every use case (D7.5, D7.6)). At least 8 correlations will be identified among the variables that characterize the welding process. The knowledge associated to these correlations will help to improve the process in time, cost, efficiency, etc. A reduction of 12% in the false alarm probability in the health use case thanks to the combination of datasets (based on use case partner estimations).

Aside of these objectives, the DoA also includes two Evaluation scenarios objectives for the 2 use cases mostly attached to objective 5.

For the smart manufacturing use case:

- An improvement of the welding process with positive impacts both on the quality of the welding process and on the final product associated with it;
- A reduction in the numbers of person hours needed to configure the robots;
- A reduction in the robot maintenance cost.

For the heath domain use case:

- Improve accuracy of AI algorithms by sharing knowledge from distinct organisations and data repositories, supporting cooperation keeping security and privacy of health data;
- More accurate clinical decision support tools for diagnosis and prognosis of diseases, conducting to better patient outcomes;
- Increasing productivity of services and more studies and patients diagnosed;
- More accurate clinical decision support tools for diagnosis and prognosis saving lives in emergency cases;
- Enable the growth of the level of research in medical imaging AI tools supported by distributed data repositories;
- Enable clinical practices to access medical imaging AI tools with gains of productivity and better patient outcomes;
- Improve Biotronics3D commercial offer, enabling partners to access its market

Beyond this comprehensive list of objectives, the main project tasks further refined and/or developed their own methodology to assess their results. This is especially true for technical aspects of the platform related to WP6 “Assessment of data quality, scalability, computational efficiency and security” activities. WP6 outcomes, also reflecting the objectives stated above are discussed in sections 3.2 “Security and robustness of the algorithms in the platform”, 3.3 “Privacy and Confidentiality of data”, 3.4 “Data value extraction and monetization strategy”, 3.5 “Computational efficiency assessment” and 3.6 “Scalability assessment” of this deliverable. For the development of the platform and the use cases, as it represents interesting dissemination material, where one could easily adopt an end user point of view and present the benefits of the platform, but also deal with long term perspectives, social impact, interviews were conducted with the developers of the platform and the partners involved in the smart manufacturing and heath domain use cases. As the

assessment of the platform and the use cases has already been conducted respectively in D3.4 “Final prototype of the MUSKETEER platform” (for the requirements completion) and in D7.5 “Use case execution and KPI evaluation in the Smart Manufacturing domain” and D7.6 “Use case execution and KPI evaluation in the health domain”, these interviews provide an interesting tool to complete the assessment performed there, with a strong focus on the impact and the long-term perspective. The objectives listed above were used to build the list of questions for the interviews. These interviews are presented in section 3.2 “General assessment of the MUSKETEER platform” for the platform and chapter 4 for the two use cases.

3 Platform perspective

3.1 General assessment of the MUSKETEER platform

To provide an overview of the MUSKETEER platform in terms of impact and assessment, we organized an interview[1] with IBM who is leading the development of the platform. Considering different assessments were performed during the project (requirements completion in D3.2 and D3.4, WP5 and the following sections 3.2, 3.3, 3.4, 3.5 and 3.6 of this deliverable for different technical dimensions of the project), this interview focuses more on the high-level impact, exploitation, and future research areas. Besides, these “lessons learnt from developers” provide strong dissemination material as expected by the description of the deliverable D8.6. It will be used for the MUSKETEER blog and for partners’ dissemination.

3.1.1 Completion of requirements

About the completion of user requirements defined in D2.1 “Industrial and Technical Requirements”, what worked well and what didn’t?

Mark Purcell

Prioritization among requirements was very useful. What you can figure is that there are less important requirements and having a list of requirements with the most important ones was useful to start working on the integration. It helped us to figure out what is required for the platform and **start integrating across the whole consortium early**. By integration, we mean specifically, the various software packages provided by various consortium partners and how they interoperate. That way we had enough people using the platform as early as possible and to see how to combine the work of different partners (like e.g., with

[1] The interview was recorded on October 15, 2021. The speaker is Mark Purcell from IBM. The interview was conducted by Antoine Garnier & Nora Gras from IDSA.

Engineering and the client connector). This led to being able to prototype the first integration back in November 2019, in M11, when we met in Dortmund at the time of the second GA and then to an actual live first demo at the mid-term review. What you always see in software projects spanning over multiple years is that what you think you need on day one, coming from the requirements gathered in the early phase of the project, isn't necessarily what you're going to need/end up with. For most of the requirements, the participating companies' business won't change, but a lot of **the technical requirements will be subject to change**, because people may not fully understand the topology of what we're building upon until it's partly built. Therefore, I would strongly recommend in any project **agile software development**.

I have nothing to highlight about parts that didn't work because I don't think we had many problems regarding requirements completion. **Not having the ability to meet in person** was an issue, as it was taken away from us by the pandemic, not easy for our collaborative projects. It is much more difficult when you meet people you've never met before only virtually in conferences, and Zoom calls. Fortunately, in the MUSKETEER case we met several times before the crisis. This helped to establish mutual understanding and trust between partners.

Any recommendation for improvement?

As I said, **don't treat the requirements phase as being in isolation**. After the first six months of a project you will not just write a deliverable report and then it's closed, cast in stone and doesn't change. **Ideally, requirements are never closed until the project is finished**. Which leads to my other point, integrate early, explore requirements in real setting and validate or invalidate them.

3.1.2 Completion of relevant MUSKETEER objectives

How successful was machine learning over a variety of different privacy preserving scenarios (i.e. various Privacy Operation Modes POMs)? Objective 1 (O.1)

We created a public GitHub repository accessible to all the partners with some public and some private repositories. It was **transparent about what people were doing along the project**. You could see how the code was going. It certainly helped people to understand what was going on especially with the different POMs (e.g. with GitHub commits). So that all worked well. I would recommend using a collaborative software for such collaborative project, a source control platform is a good thing, definitely.

Creating predictive models without directly exposing them to the data consumers (O1.2)

So, yes certainly, we did it. **This is a design decision**; we have an access control model in MUSKETEER. Whoever was involved in the process of building a model, would have access to

the final model to download from the platform. Users who are not involved in the task would not have access to the model. And **no one would have directly access to each other's data (only to the results)**.

Providing robustness against the external and internal threats, what would you say (O.2)? Does it allow a secure information exchange among the platform users (O2.2)?

This can be split into two kinds of threats: **traditional cybersecurity threats and AI threats**. I'll start with the cybersecurity threats. One of the main goals, when we started this work, was to minimize, I wouldn't say to eliminate, it's probably impossible to eliminate, but certainly **minimize the attack surface** from a traditional cybersecurity perspective. When you look at most federated learning platforms, everybody who wants to participate in the federated process, exposes an Internet facing service or at least a service within the network in which they are engaged, like in MUSKETEER, therefore it had to be on an "Internet scale". When you are talking about providing an Internet facing service, with all of the risks that go along with that, your attack surface is huge. You can do various things to minimize it, but it's still significant. So, we didn't want to do that in MUSKETEER. Therefore because of the security considerations, we planned from the start to **have a brokering mechanism where no participant in the Federation is ever directly connected to any other participant**. Participants send their model updates to an aggregator (via a broker) and there is no direct contact between participants. So, everybody's essentially separated by at least an extra network. It's far more secure. That's a very important thing from the data privacy point of view too. Because, of course, if you had an internet facing service, and that service wants to do training, that service has access to the data. If that service is attacked potentially your data is exposed.

The other side to it is AI robustness. The models themselves could be attacked by malicious users during the training phase. The work was done by Imperial and IBM did some of that as well. We did some things like adversarial training, routines, and how can you detect and minimize the effect of malicious participants. **That's still very much an open area for research**, and it's still progressing.

Enabling data provider to share those data sets, thanks to the validity of creating predictive models without explicitly giving their data set (O.3.1)? So, do you consider this objective fully achieved?

This question is related to the fundamental *raison d'etre* of federated learning. The whole purpose of federated learning is to do exactly that: to allow users to participate in a modelling exercise and not have to share data. In a traditional way, all the data would have to be centralized in a given location, and then you train there. Federated learning prevents that. I think we have fully achieved that in MUSKETEER, and **we've shown that we can do**

that across geographies right now. Not just even across departments in the same local area network. We can do this even across continents. What was envisioned in 2018 is still valid today and will be so going forward. That's exactly what people want to do. They want their data to remain on their premises, but still to be able to extract value from it to contribute to a model. What we haven't done in MUSKETEER, and it was beyond the scope of the project, is to offer an accountability perspective on the process. In other words, being able to retrospectively say everything that happened across a federated learning task "actually happened". We've no way to prove that currently on the platform. **But it's something to think about for the future, the whole accountability of Federated Learning.**

About involvement of SMEs, have you noticed the interests for the platform from external stakeholder, especially? (O3.3)

SMEs involvement has been complex, because of the global situation, dissemination was less easy. I can certainly talk about, the other side of it, so within IBM, we are looking to move some of the assets that we developed here into an IBM product called IBM Federated Learning. It's still under development. In MUSKETEER, the server side that we built at IBM is a collection of microservices. A refactored version of some of them is ongoing work and we plan to add that to IBM Federated Learning.

About providing a standardized and extensible architecture? (O4)

There is a section on this in Deliverables 3.2 and possibly 3.4. But, in short, from IBM point of view, we have tried to use as many open standards as possible "Publish, Subscribe" Standards (AMQP standard, HTTPS, JSON for your Message Payload, etc.). And you can even see some of that in the open-source contributions that we've made. We've tried to do that as much as possible. And even some of the cloud services that we use are open source, so we use S3 for model storage and RabbitMQ, these are all license free or open-source products.

About the compliance with IDSA standards (where we know a pillar of the framework is the mutual check of participants IDs)?

Privacy has been a very important aspect in the project, privacy of all things. Not just the privacy of the data, but the privacy of your identity. So, **it's a feature of the platform but it's not an irrevocable feature**. Everybody has a user account on the platform. The information is obviously available, it's just not revealed to any other user, it doesn't mean it couldn't be, and it is just to fulfil the privacy concerns that we have. On the other hand, from a general perspective, **a lot of the concepts in the IDSA reference architecture are, in fact, what we use**. Through the MUSKETEER client connectors, what we have inside is essentially data applications. Communications are not direct either, it's a brokered system. So again, that's

an important point. In a lot of ways, we followed the reference architecture as much as we could.

GAIA-X compliant, would you be interested in that (in a potential future)?

This is not that concrete obviously at the moment, but put it in general terms, using as many standards as we could is a good point. Particularly **having a messaging-based platform makes it, I think, easier to interoperate with other platforms**. Message formats are all described in deliverable documents and open-source software. So, it should be easy to construct messages to pass between platforms. No framework is going to work with another out-of-the box perfectly but broadly speaking, the effort here shouldn't be massive.

3.1.3 Completion of ICT13 2018-2019 objectives

Do you think that personal data protection was improved through the project? Compliance with the General Data Protection Regulation (and other relevant legislation) is made easier for economic operators?

This is the general concept of federated learning, right? With data remaining on premises. This is **the whole idea of MUSKETEER where even the privacy of your identity is protected**. So, a lot of things we did were very much done with a view to solving some of the questions that are posed by GDPR.

As a citizen, you're seeing that?

Citizens' trust is improved as privacy-aware transparency and control features are increasingly streamlined across data platforms and Big Data applications. MUSKETEER concept is different but is the citizens' trust reinforced?

We live in the era of misinformation. Social media platforms now exist as well as are traditional routes to misinformation. This is always going to be a problem. That's why accountability might be a key here. **Having a way of showing that what happened on the platform makes it fully accountable**. **This is an open area for new research** as we discussed before. At the moment, on most platforms, you just have to sign up, understand the terms and conditions, and that's it, but there's no actual accountability. On the other hand, it heavily relies on the level of trust you have in the network. If you're working across the departments. You might say, you completely trust everybody in the department. But then, if you're operating across the Internet, you might say, well, I have zero trust, you're happy staying completely private, while sharing your anonymized information. This is in the end the best way to prevent leakage. In our situation in MUSKETEER, you might even not trust the aggregator completely, it depends on the situation.

Finally, about a better value creation from personal, appropriate data, do you feel this was the case?

I definitely believe that federated learning, or federated analytics, is definitely going to be bigger in the future. We will be using more of it. There's no doubt about that. And our project becomes a good reference point on the route of federated learning. There was also some work on data value estimation in the project. That would lead us, let's say for future research, to see how we could maybe **build a marketplace based on federated learning mechanisms and see how people could be rewarded for providing good contributions**, valuable contributions from their datasets while still keeping the data secure and private. There is certainly a **balance to find between trust and reward for contributors in such ecosystems**.

3.2 Security and robustness of the algorithms in the platform

Machine learning algorithms are vulnerable and can be the objective of attackers who can exploit those vulnerabilities, both at training and at test time. One of the main objectives of MUSKETEER is to understand the vulnerabilities of federated learning and to include mechanism in the platform to mitigate the effect of possible attacks. This work has been conducted across the different tasks in WP5. For this, in deliverable D5.1 we described the threat model, including the different types of attacks that could be performed against the federated learning algorithms across the different POMs in the platform. This threat model served as a reference to test the robustness of the unprotected algorithms in the platform by crafting both attacks at training and test time (see deliverables D5.2 and D5.3) and to develop and integrate different mechanisms to mitigate these attacks (see deliverables D5.4 and D5.5). The complete security evaluation of the security of the algorithms developed and integrated in the MUSKETEER platform will be included in deliverable D6.3. In the remainder of this section, we will describe more specifically the work performed in WP5 for the security of federated learning in MUSKETEER both during training and at test time.

3.2.1 Poisoning Attacks

In general, machine learning algorithms can be vulnerable to poisoning attacks, where the attackers aim to subvert the learning process and manipulate the behaviour of the algorithm to produce errors when the resulting machine learning model is deployed. Typically, these attacks occur in settings where the data collected to train the machine learning algorithm is not trusted or in cases where the source code to train the algorithms has been deliberately manipulated by attackers. In the first case, attackers can inject a set of malicious training points in the training set of the victim to perform the attack.

These attacks are also plausible in federated learning settings, where the attack surface is bigger compared to standard (centralized) machine learning algorithms. Thus, in federated

learning attackers can be categorized as outsiders or insiders (see deliverable D5.1 for a more complete explanation of these concepts) and, depending on their capabilities can perform both data and model poisoning attacks. In the first case, attackers can manipulate the training data for some clients in the platform and try to manipulate the resulting aggregated model. On the other side, some attackers may also be able to manipulate directly the parameters of the model that some clients send to the aggregator. These are known as a model poisoning attacks.

In deliverable D5.2 we showed the necessity of protecting the federated learning algorithms in the platform. Thus, the attacks devised and performed against the unprotected (standard) machine learning algorithms in the MUSKETEER platform proved to be very effective to completely compromise the performance of the resulting machine learning models during the training phase. These attacks were performed for both supervised and unsupervised learning algorithms across different POMs in the platform. More advanced attacks where multiple malicious clients collude towards the same objective were analysed more carefully in deliverable D5.6.

To mitigate this threat, we have developed an implemented different defensive techniques that are described in deliverable D5.4. These defences fall into two categories: robust aggregation methods and data-prefiltering. In the first case, we have developed novel techniques such as Adaptive Federated Averaging (AFA) [Muñoz-González et al. 2019] which allow to detect attempts of compromising the performance of the model at each training iteration, and thus, mitigate the effect of possible poisoning attacks without compromising the models' performance when they are not under attack. However, robust aggregation techniques are not applicable for all POMs, as there are certain operations that are needed to implement them that are not supported when operating in the encrypted domain (see deliverable D5.4 for more details). To sidestep this difficulty, we have implemented different data pre-processing and outlier detection methods that also help to mitigate or reduce the impact of poisoning attacks for all POMs in the platform. The evaluation of all these techniques against attacks involving users' collusion will be described in deliverable D5.7 (M36).

The two hackathons organized as part of MUSKETEER project were dedicated to the security of federated learning against data poisoning attacks. In the first hackathon, the participants implemented their own defences against different poisoning attacks using the MMLL library developed in MUSKETEER. In the second hackathon, participants tested the defences developed for the Robust MUSKETEER MMLL library implementing their own data and model poisoning attacks under both white and black box settings. None of the participants was capable of bypassing our defences, providing an additional indicator of assurance on the quality of the solutions implemented in the library.

The analysis of more targeted attacks, such as backdoor attacks, was not considered in the work plan described in the Consortium Agreement. This aspect represents a challenge and an opportunity for future work beyond the MUSKETEER project.

3.2.2 Attacks at Test Time

At test time, machine learning algorithms are vulnerable to evasion attacks, where attackers can introduce small manipulations to the data to produce errors in the targeted machine learning algorithm. These modified data points are commonly referred to as adversarial examples. This vulnerability also affects federated learning algorithms, once the resulting collaborative machine learning model is deployed. These evasion attacks are analysed in the context of supervised learning algorithms for Neural Networks under POMs 1-3 in the MUSKETEER library.

Deliverable D5.3 shows that the unprotected (standard) machine learning algorithms implemented in the MMLL library are vulnerable to evasion attacks, both under white and black box settings. The impact of these vulnerabilities was also assessed on the smart manufacturing use case considered in MUSKETEER project.

To defend against this threat, we have implemented a technique to perform federated adversarial training, where adversarial examples are crafted during the training of the collaborative learning model to enhance the robustness against evasion attacks at test time. The results shown in deliverable D5.5, including again the smart manufacturing use case, endorse the usefulness of this approach to mitigate this threat.

3.3 Privacy and confidentiality of data

One of the main goals of the MUSKETEER platform is to provide a wide variety of machine learning algorithms under different Privacy Operation Modes (POMs), such that the final user has a direct access to those training methods without worrying about the deployment of specific protocols, as all the operations needed are provided by the platform.

The confidentiality of the training data is provided by design, since the implemented training algorithms rely on existing privacy preserving protocols and methods with proved security. For instance, in the Federated Learning approaches, training data does not leave the data contributor facilities, and only model updates are exchanged, as described in the Federated Learning literature. Therefore, this approach is considered as secure since from the aggregated parameters (models, gradients), it is not possible to infer individual training samples. Anyhow, as analysed in the literature, if any information leakage can be observed from the shared averaged quantities that are exchanged with the aggregator, then a noise perturbation method (also implemented in MUSKETEER) must be used to further protect the

training data, if necessary, in the line with Differential Privacy approaches. Other POMs rely on cryptographic technologies to protect the data (e.g. homomorphic encryption), such that operations take place in the encrypted domain. The security of these approaches is well studied in the literature, and they are considered as secure from a cryptographic point of view unless the encryption mechanism itself is broken (the secret key is revealed). Anyhow, the security of the encryption method can be augmented by selecting a larger encryption key, but at the cost of also increasing the computation and storage/transmission needs. Since the encryption library is external to MUSKETER, and the key length can be set arbitrarily large by the end user, the methods relying on encryption can be as strong as desired, by simply providing MUSKETEER with stronger encryption libraries or larger keys. Finally, one of the implemented POMs (POM 6) does not rely on encryption but on existing Secure Two-party protocols to obtain some partial results needed by the training algorithms, such as dot products between training data and model parameters. In this case, the security analysis is detailed in the original protocol description (e.g. any Secure Dot Product protocol, or any Random Matrix Disguise approach). Under this POM 6 approach, to alleviate the computational and transmission costs of other POMs, we need to expose some partial information during the training process, for instance, the number of provided training patterns, the model outputs, etc. If the partial information exposure is not acceptable for a given end user application, then a more restrictive POM (e.g. 4, 5) should be used instead. In what follows, we provide a more detailed discussion about the security of the implemented POMs under the semi-honest (a.k.a. honest-but-curious) assumption, e.g., POMs 4, 5 and 6. A more detailed description of these POMs and the implemented algorithms is included in Deliverable 4.7.

3.3.1 POM 4

In this POM we have three main parties:

- Master Node (MN, a.k.a. Aggregator). It controls the training process and obtains the final trained model.
- Crypto Node (CN). It provides the encryption keys and helps to solve some operations on the encrypted domain.
- Worker Node (WN). It provides some training data.

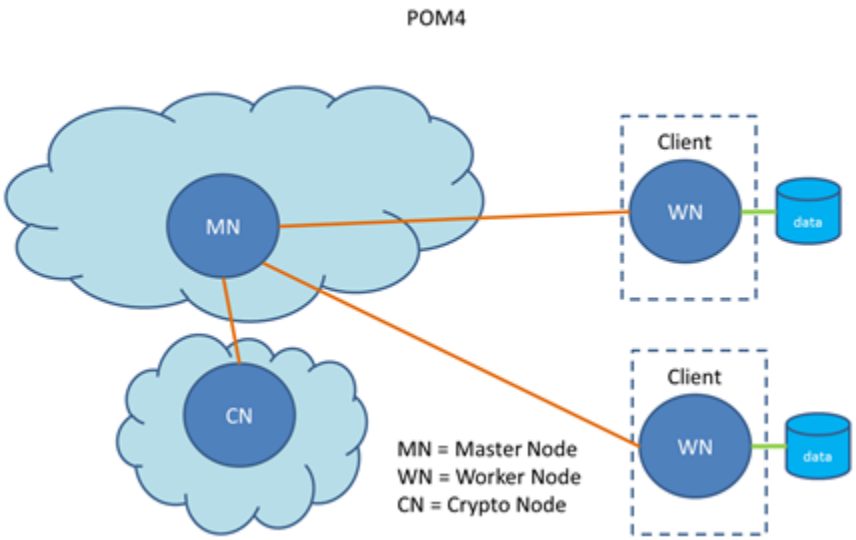


Figure 2. POM 4 general setup.

The operation under this POM is inspired from [Gonzalez_2017] and it is cryptographically secure whenever the MN and the CN do not collude. It relies on an additive homomorphic encryption (HE) scheme. Note that the cryptographic methods are external to MUSKETEER. The HE included in the current MUSKETEER release is based on the Paillier system with the key length as a parameter, but any other additive HE cryptosystem can be used by MUSKETEER if it is wrapped into a python class with the same functionalities as the provided one. The summarized POM operation is as follows:

- The CN generates public and private keys. It shares the public key with the other participants and keeps the secret key.
- Every WN encrypts the training data and sends the whole of it to the MN (only once).
- The MN operates on the encrypted data using HE properties, but it may need some help from the CN to implement the unsupported operations. For that, it first blinds the operands (adding or multiplying by a random number in the encrypted domain), sends the blinded encrypted operands to the CN that decrypts them and computes the operation on the blinded operands. The CN encrypts the result and sends back the blinded encrypted result to the MN, who removes the blinding in the encrypted domain to obtain the encrypted result.

- The MN is able to update the model using these procedures, but it is not able to decrypt the specific contribution (e.g. gradient) from every WN.

This POM is as secure as the cryptographic system that is being used, always under the assumption that the CN and MN do not collude. If stronger security is needed, a longer key can be defined (the cryptosystems are harder to crack for longer keys), or even any other HE cryptosystem with improved characteristics can be used by the MUSKETEER algorithms. The CN only sees randomly altered data when it decrypts the blinded cyphers, and the MN is not able to decrypt the encrypted training data.

3.3.2 POM 5

In this POM we have two main parties:

- Master Node (MN, a.k.a. Aggregator). It controls the training process and obtains the final trained model. The MN uses encryption to protect the model confidentiality, when it is shared with the workers.
- Worker Node (WN). It contributes with some training data to the training process, but the training data is not encrypted and it does not leave the WN.

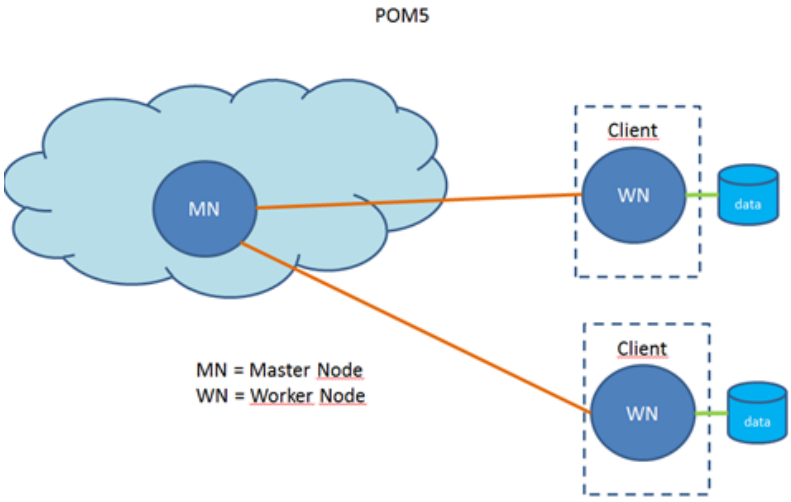


Figure 3. POM 5 general setup.

The operation under this POM is also partially inspired from [Gonzalez_2017], but in this case the MN plays the role of CN and we only encrypt the model parameters. It relies on an additive homomorphic encryption (HE) scheme. Note that the cryptographic methods are external to MUSKETEER. The HE included in the current MUSKETEER release is based on the Paillier system with the key length as a parameter, but any other additive HE cryptosystem can be used by MUSKETEER if it is wrapped into a python class with the same functionalities as the provided one. The summarized POM operation is as follows:

- The MN generates public and private keys. It shares the public key with the WNs and keeps the secret key.
- The MN encrypts the model and sends it to the WNs, to compute operations (model updates) using the training data.
- The WNs may need some help from the MN to implement the HE unsupported operations. For that, it first blinds the operands (adding or multiplying by a random number), sends the blinded operands to the MN that decrypts them and computes the desired operation, encrypting again the result. The MN sends back the blinded encrypted result and the WN removes the blinding to obtain the encrypted result, such that the encrypted model can be updated at the WN (using HE properties) and finally decrypted at the MN.

This POM is as secure as the cryptographic system that is being used, always under the assumption that the MN and WN do not collude. If higher security is needed, a longer key can be defined (the cryptosystem is harder to crack for longer keys), or even any other HE cryptosystem with improved characteristics can be used by the MUSKETEER algorithms. The MN is able to decrypt the updates from every WN (e.g. aggregated gradients, updated model, much in the line of standard Federated Learning), so if additional confidentiality is needed, then some form of Differential Privacy (e.g. adding noise to the training data) has to be used, as in the standard Federated Learning approach. MUSKETEER also provides a noise injection method during the data pre-processing stage, but at the cost of potentially reducing the performance of the resulting models.

3.3.3 POM 6

POMs 4 and 5 require a lot of computation and transmission resources, since they rely heavily on costly cryptographic operations. In POM 6 we maximally simplify and reduce the computational requirements by using lightweight protocols at the expense of revealing some intermediate results (model outputs, partial statistics). We do not use encryption and we rely instead on Secure Two-Party protocols that provide security to both model and training data. We have two main parties:

- Master Node (MN, a.k.a. Aggregator). It controls the training process and obtains the final trained model. The model does not leave the MN.
- Worker Node (WN). It provides some training data. The training data does not leave the WN, only intermediate results are exposed.

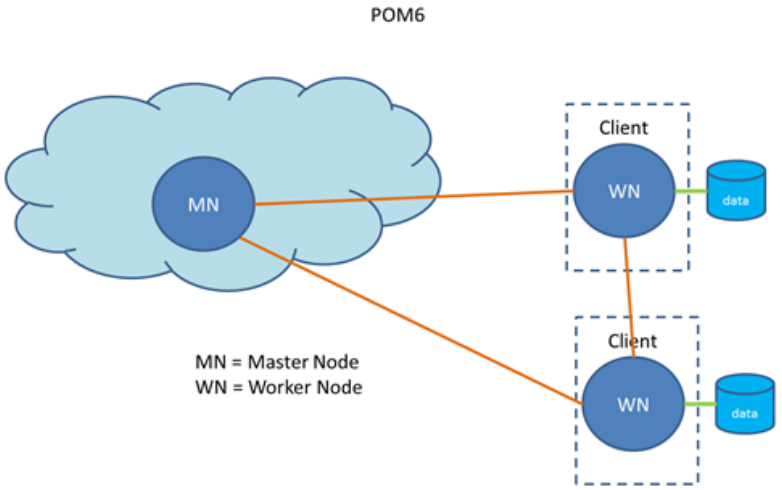


Figure 4. POM 6 general setup.

The operation under this POM is inspired by the Secure Multiparty Computing literature [Yao_1986][Goldreich_1987][Du_2002]. However, in the most popular SMC approach (Arithmetic Secret Sharing), the information must be split and shared among a set of servers (usually more than three), and again we need to guarantee that they do not collude, otherwise the information can be exposed. Also, some costly protocols like Oblivious Transfer are needed to complete the computations. In POM6 we have decided to mainly rely on Two-Party protocols, since they are free from the collusion problem (both parties are interested in following the protocols, otherwise their own information can be exposed) and also allow the exposure of some intermediate results, but always under the premise that the training patterns cannot be obtained from those intermediate results. Under this scheme, the model does not leave the MN and the training data does not leave the WN. The operations needed to update the model take place using Two-Party computing protocols, without needing to exchange neither the model nor the training data. The main used protocols in POM6 are:

- **Secure Dot Product** (SDP). We have implemented the method described in [Zhu_2015], but any other SDP approach could be used.
- **Random Matrix Disguise** (RMD): Relies on the use of a random matrix only known every worker to randomly permute and blind the elements of the original data matrix [Mohassel_2011][Wang_2011] [Wang_2015]

The above mentioned methods only expose the information described in the corresponding security analysis of the protocols published in [Mohassel_2011][Zhu_2015], [Wang_2011] [Wang_2015].

Some of the ML training algorithms may need to reveal some intermediate results, such as model outputs, average gradients, covariance matrices or cross-correlation vectors for POM6 to operate. If it is not acceptable that this information is exposed/revealed to the MN, then another POM must be used, or alternatively, some form of Differential Privacy (e.g. adding noise to the training data) has to be used, as in the standard Federated Learning approach. MUSKETEER also provides a noise injection method during the data pre-processing stage, but at the cost of potentially reducing the performance of the resulting models.

3.4 Data value extraction and monetization strategy

One of the aspects covered by the MUSKETEER platform is the ability to estimate the value of the contribution from every participant, task known as Data Value Estimation. Although we will summarize here some of the findings, the complete description of the implemented approaches and the experimental results can be found in D6.4.

The standard “de facto” approach to estimate the contribution of different participants to a given machine learning task is the Shapley Data Value estimation approach, which has been extensively studied in the context of cooperative game theory [Shapley1953][Ghorbani2019]. This approach offers some attractive features: it is task-dependant in the sense that the data is valued only if it allows to improve the performance of the model, the reward is fully distributed among the participants, equal data contribution means equal reward, and the addition of several contributions gets a reward equal to the sum of the individual rewards. The calculation of Shapley values is quite simple but it implies a large computational cost (a large set of models needs to be trained) and it also requires that all the training data is located at the same place, something that does not hold in the MUSKETEER context. Some alternatives, like those proposed in [Song2019], suggest using the information exchanged during the federated learning process (models, gradient vectors) to reconstruct the variety of models needed to estimate the Shapley values. In this way we can calculate estimates of the different models that would be obtained if different combinations of data sets were used, without the need to train them from scratch. Obviously, an exact reconstruction of all models is not possible and we only get estimates, but according to the experiments in [Song2019], it seems that good approximations of the data value are possible. In what follows, we will name these approaches as “on-line a posteriori DVE”, since a single training of the Federated model is needed to obtain the Data Values estimates, and the Data Values are obtained after the training takes place. In the

context of the MUSKETEER implementation, we provide this "on-line a posteriori DVE" approach for the implemented machine learning methods, but also proposed a new approach to the DVE problem, which we describe as "a priori" estimation. Under this "a priori" scheme, we want to estimate the data value *before* actually training any model. The proposed "a priori" scheme relies on the assumption that the statistics of a good dataset should be close to those of the reference one (e.g. a validation set). In a traditional (centralised) setup, where all data is available for analysis without any kind of restriction, this task could be stated as the problem of verifying that the different datasets share a common (or at least similar) input-output joint distribution, but in the distributed setup we have devised a method to extract simple statistics and computing distances among them to estimate the Data Value. This approach is described at length in D6.4, where we show that good estimates can be obtained, that closely mimic the "brute force" Shapley values, as shown in the two Figures below (more examples are included in D6.4).

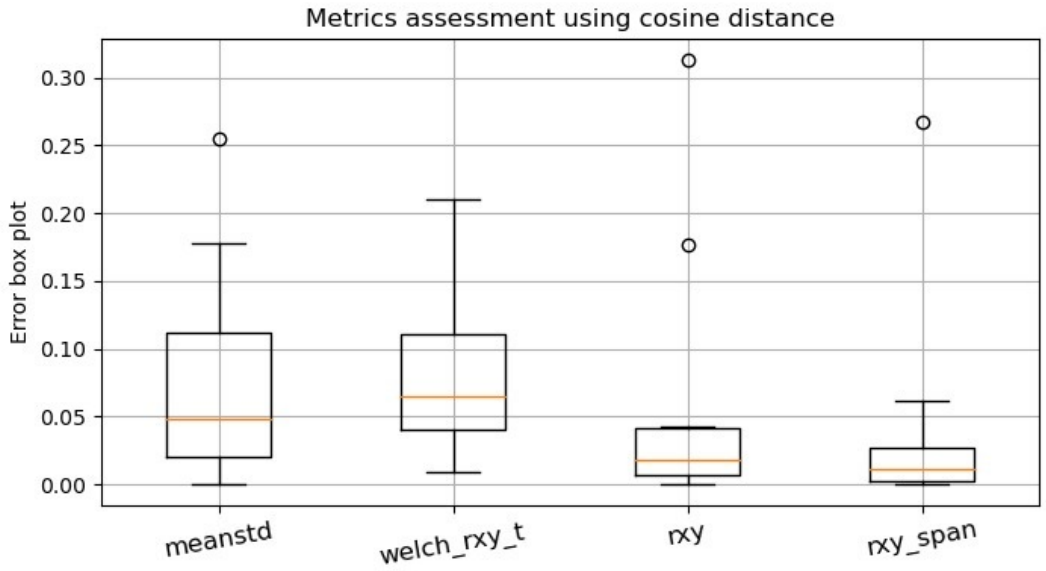


Figure 5. Boxplot of errors obtained when computing the cosine distance in the different scenarios using the proposed statistical data characterizations for "a priori" DVE. The "rxy" approaches provide the lowest error with respect to the Shapley value estimation.

Figure 6. Benchmark result for case 10: "a priori" estimation vs. Shapley values. Also shown the Task Alignment estimation values when they are below the threshold.

The existence of these Data Value estimation methods facilitates the implementation of a variety of monetization strategies, as also described in D6.4. When only "a posteriori" estimation methods are available, the number of possibilities is reduced basically to training a model with all possible data contributions (possibly weighting their contributions to the model proportionally to the observed DVEs) and finally reward them proportionally to their final observed Data Values:

- **Direct reward distribution**: the aggregator has a total budget and redistributes it among all the participants in proportion to their finally estimated Shapley value (share), after the model training process has been completed. Both aggregator and participants initiate their interactions blindly, in the sense that the aggregator has no clue about the expected data quality provided by a given set of participants, and the participants do not know whether they are going to receive any payment in the end.

However, the proposed "a priori" estimation method opens up new venues for data monetization approaches, since the "a priori" DVE allows a preliminary negotiation between the aggregator and the participants before they engage in the actual training process or expose their training data in any form. Inspired by this new approach. We have proposed the following monetization approaches:

- **Market model**: In this case every participant offers to the market their data (also publishing the statistics needed for the "a priori" Shapley estimation), and freely fixes a price, that anyone wanting to use that data has to pay in advance to get access to or interact with. The aggregator is able to see all the available data offers, it can evaluate their ``a priori" Shapley value based on the published statistics, and choose

the combination of participants that minimizes the expenditure while maximizing the expected[2] benefits (maximal model performance).

- **Two-step reward**: In an attempt to balance the risks incurred by the aggregator or participants, we have proposed a hybrid approach, named as "two-step reward". In this case, the aggregator has a budget to spend on the model training and splits the reward in two parts, not necessarily equal. A preliminary reward can be assigned relying on the "a priori" Shapley estimates. This allows the aggregator to filter out unwanted contributions (those with very low estimated value) and also gives the participants a preliminary earning (paid in advance to training), proportional to the estimated share value and to be received after their "willingness to cooperate" is confirmed. As already mentioned, the "a priori" Shapley estimation is not necessarily optimal, since it does not take into account the performance of the final obtained model. Therefore, in a second stage, the aggregator runs a training process with the selected participants and uses any available online Shapley value estimation method that presumably provides more accurate estimations than the "a priori" ones, since they are obtained after the model training is completed. Upon these new more accurate DV estimates (task/model dependent), the second part of the reward is distributed.

We have also provided a method to take into account the final achieved performance, since the Shapley values only reflect the share and not the actual benefit:

- **Performance-based reward modulation**: the Shapley values represent an estimation of the contribution of every participant in a "cooperative game", but they do not reflect by any means the quality of the obtained result. As an example, in a FL scheme with two workers providing equally good data, their respective -ideal- Shapley "share value" would be of 0.5, and the resulting reward values could be reasonable. On the other hand, in a FL scheme with two workers providing equally bad data (e.g. random), their respective -ideal- Shapley share value could also be of 0.5, but the resulting reward proposal is wrong, since those users should not receive any reward. We propose to define an "achievement" function, such that the achievement value is zero if the final performance (P) is below a given reference threshold, takes unit value when the goal is achieved and intermediate values in between. This achievement value is used to estimate the final rewards, under different "trust" assumptions, as described in D6.4

[2] Note that "expected" means here that the ultimate model improvement can be achieved or not.

As a further improvement/research, it would be interesting to explore new statistics and metrics that could provide better "a priori" Data Value estimations. Also, from the monetization point of view, the complete study of the potential scenarios or dynamics that may arise during am interaction among an aggregator and different participants exceeds the scope of MUSKETEER. It could be of potential future interest to model these interactions from a dynamic market and game theory perspective.

3.5 Computational efficiency assessment

We will summarize here the observed results for every algorithm and POM until the training process is completed. A more detailed analysis of the assessment can be found in deliverable D6.2.

3.5.1 Federated POMS

Transmission costs:

We have measured the bytes sent and received by the master node. The following pictures belongs to the FBSVM algorithm, but the behaviour is very similar in every algorithm. The datasets used were:

- Diabetes (Dataset S: Small size)
- Income (Dataset M: Medium size)
- MNIST (Dataset L :Large size)

Dataset	Number of Patterns (train)	Number of Patterns (validation)	Number of Patterns (test)	Number of Features
Diabetes	500	100	168	8
Income	26,049	6,512	16,281	107
MNIST	50,000	10,000	10,000	784

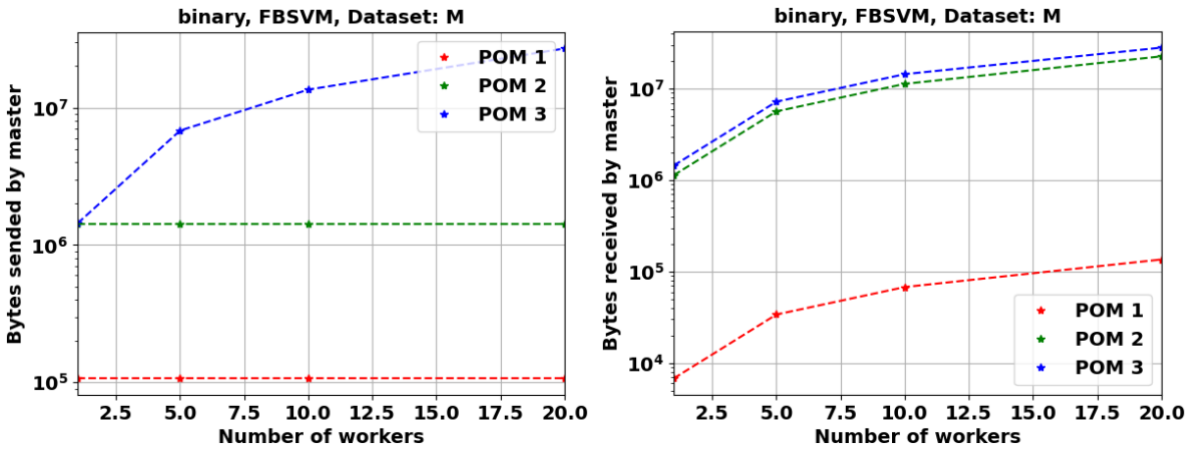


Figure 7. FBSVM algorithm

In every iteration of the training process, the aggregator sends the ML model to every worker.

Information sent from master to workers:

However, in POM1 and POM2, the master node broadcast the data, so it only needs to submit the data once to the *pycloudmessenger* library and the bytes sent by the master node do not depend on the number of workers and remains constant.

In the case of POM3, the training process is sequential. The master node sends and receive information from the workers one by one, for that reason we can observe a linear dependency of the information sent with the number of workers.

POM2 and POM3 make use of homomorphic encryption (and encrypted SVM weights take up more memory), for that reason, in training processes with 1 worker, the information sent by the master is lower in POM1 than in POM2 and POM3 and is similar in both methods with encryption.

In every training iteration, the worker nodes send a copy of their local SVM weights to the master node. For this reason, we can observe a linear dependency of the information received by the master node with the number of worker nodes.

POM2 and POM3 make use of homomorphic encryption (and encrypted weights take up more memory and these encrypted weights must be transmitted in every iteration). For this reason, the slope of the linear dependency is higher in POM3 and POM2 than in POM1 and when a single worker is running, the information received by the master is lower in POM1 than in POM2 and POM3.

Information sent from workers to master:

In every training iteration, the worker nodes send a copy of their local SVM weights to the master node. For this reason, we can observe a linear dependency of the information received by the master node with the number of worker nodes.

POM2 and POM3 make use of homomorphic encryption (and encrypted weights take up more memory). For this reason, the slope of the linear dependency is higher in POM3 and POM2 than in POM1 and when a single worker is running, the information received by the master is lower in POM1 than in POM2 and POM3.

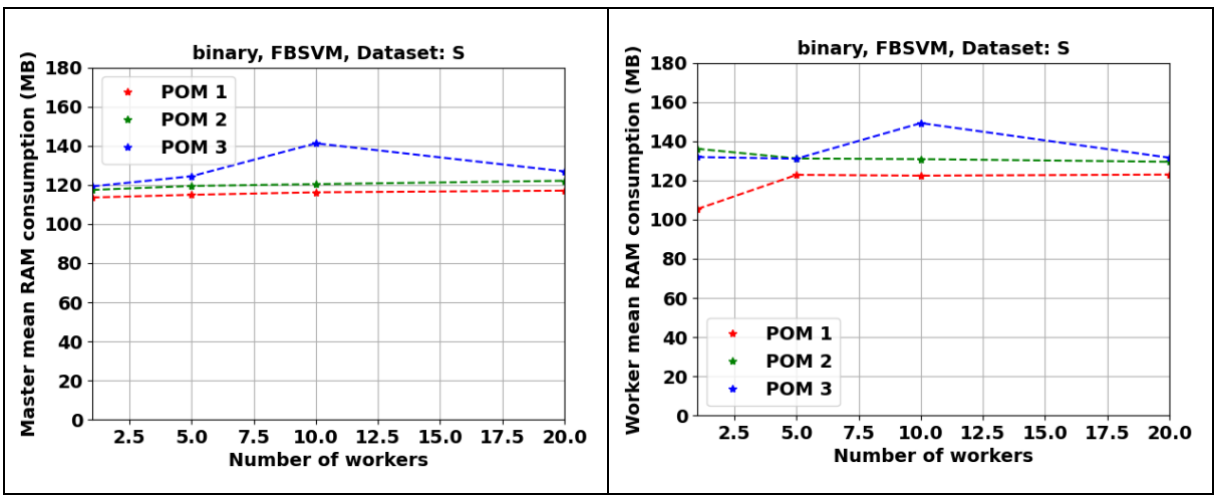
Memory usage:

We have measured every two seconds the memory consumption of the Docker container that contains the master node and one of the Docker containers that contains a worker node. Since the memory usage vary along the training process, to obtain a single metric, we average the results over time.

The behaviour is very similar for every algorithm. For small datasets, the RAM memory used by the processes is negligible with respect to the RAM memory used by the complete container. The influence of number of workers and the POM cannot be observed.

For medium size datasets, the RAM memory used by the processes is still more or less negligible with respect to the RAM memory used by the complete container. Since POM1 makes no use of encrypted information in memory, we can observe a lower memory consumption than in POM2 and POM3.

For large size datasets, we can observe the influence of the POM and dataset size in the memory consumption. In POM1 we can observe a lower memory consumption (no encryption has been used). Since we split the dataset among the different workers, the memory consumption in every worker decreases as we increase the number of workers. The master node of POM3 receives the centroids of every worker, so the memory increases as we increase the number of worker nodes.



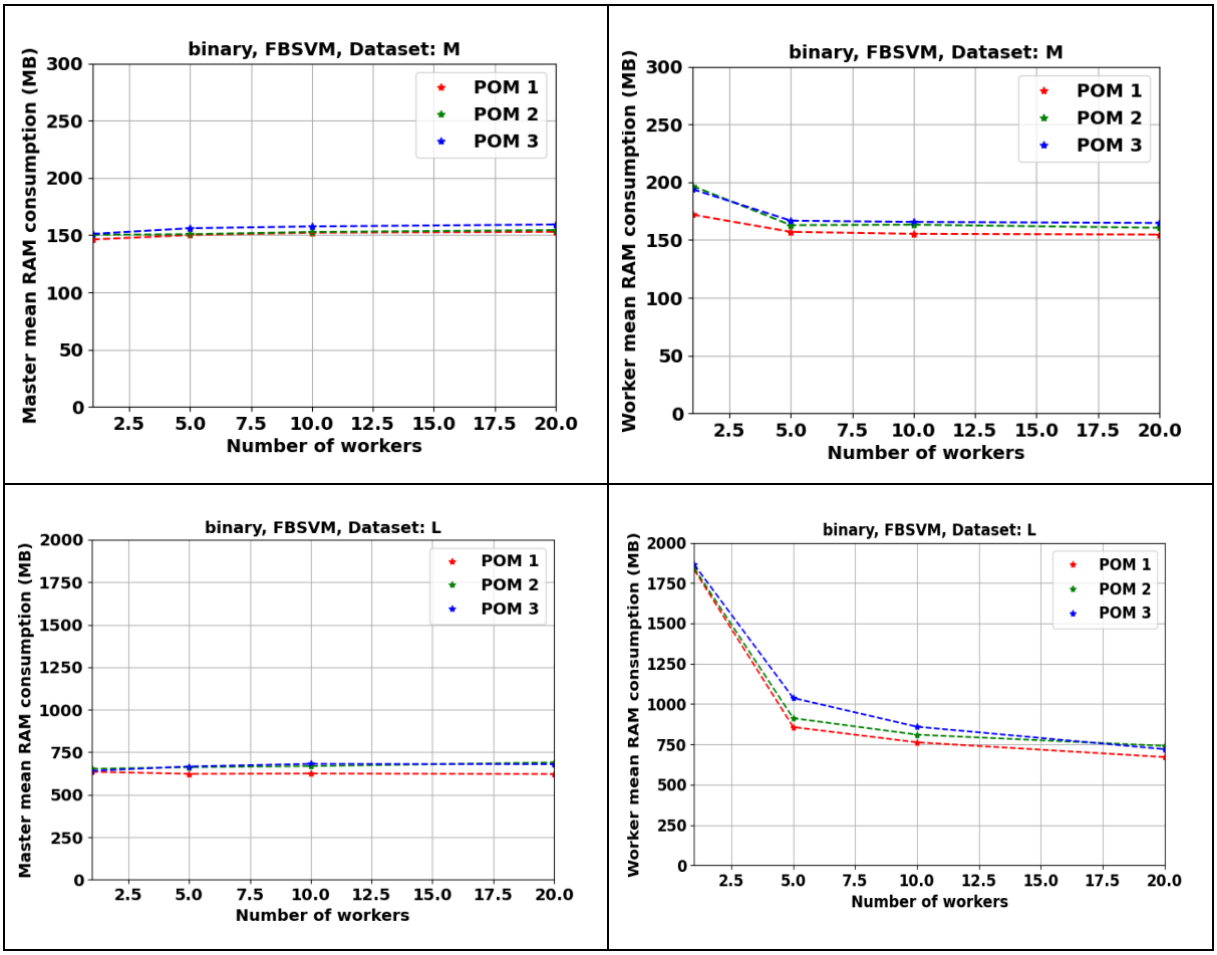


Figure 8. Influence of the POM and dataset size in the memory consumption

Although the memory in master node increases with the number of workers, in this picture seems similar since is an average over time and the time spent since the master node receives the model from every worker and updates the model is despicable from the total training time (most of the time, the master node waits for information). And concretely, the FBSVM algorithm only needs to send an array (main part of the model remains constant and is not sent). For more information about other algorithms, see D6.2.

3.5.2 Semi Honest POMs

In POM 6 data is not encrypted, so it basically uses the same (order of magnitude) storage space as in the centralized situation. However, the POMs relying on data encryption (POMs 4 and 5), require extra storage and transmission capacity. To illustrate the scale of needed storage/transmission, we have computed the size in Mbytes of every one of the used datasets, and compared the plain size with the encrypted one, as shown in Figure 9 below.

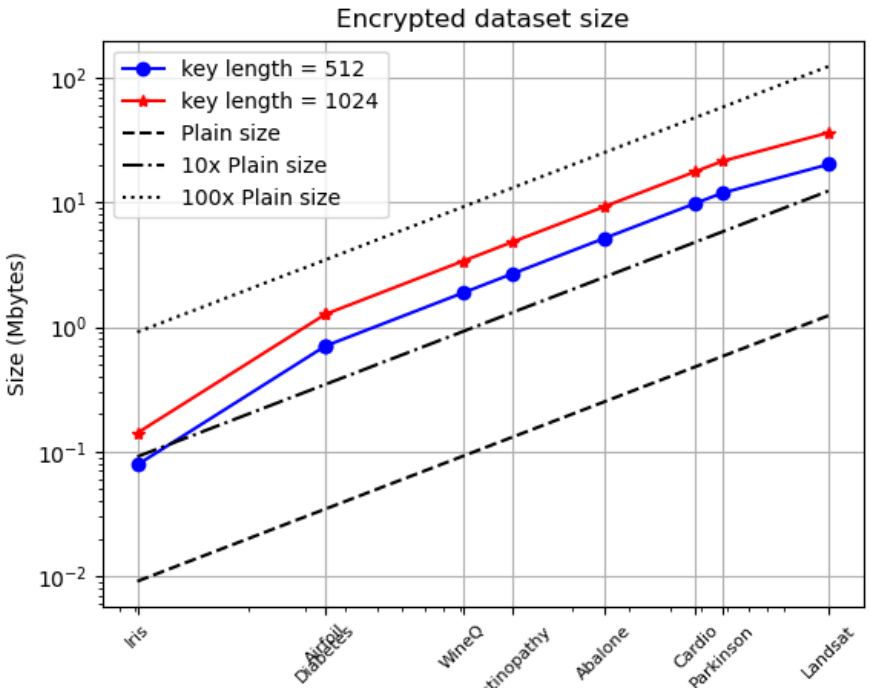
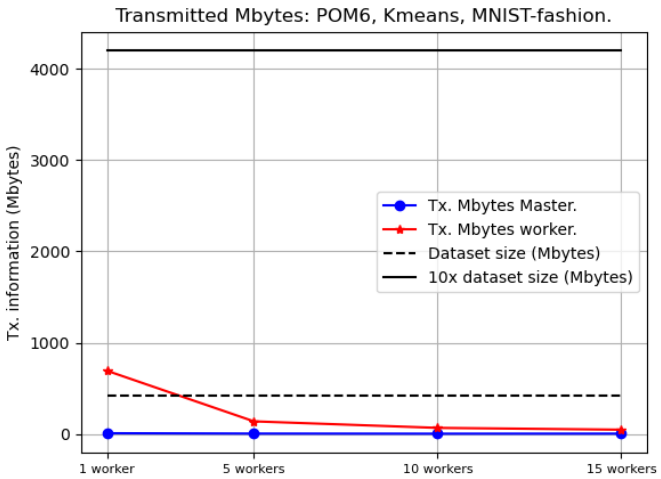
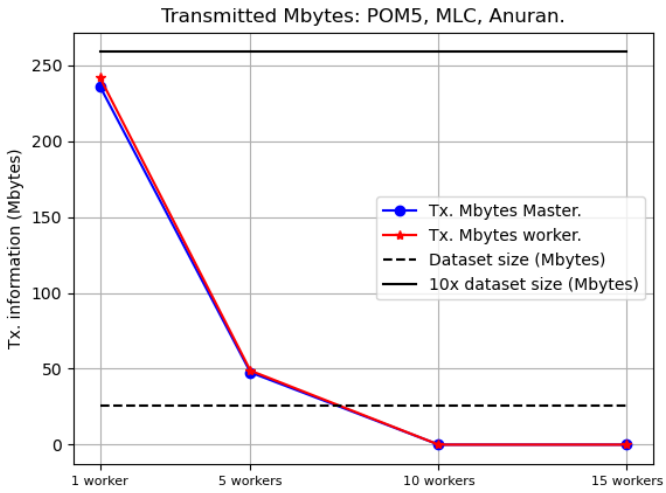


Figure 9. Size of the assessment datasets, plain vs. encrypted.

We observe that the size of the encrypted datasets is between 10-100 times larger than the plain data. This is also a factor that affects the total training time, when encrypted models or patterns need to be transmitted. We have estimated that, in our particular experimental conditions, the transmission rate is about 2.7 Mbytes per second on a steady regime. Once again, the communication means used by MMLL could be replaced by faster ones (dedicated transport networks) if needed for a specific application, but when the Internet is used as the communication platform, rates like this one are expected. We have measured the total amount of transmitted information as depicted in the following sample Figures (the complete set is available in D6.2):



(a)



(b)

Figure 10. Examples of transmitted information, processing and transmission times as a function of the No. of workers.

We observe that the amount of transmitted information by both the master and workers is below the reference thresholds (10x the dataset size[3]). In other cases, we have observed that, when the dataset is very small, the overhead transmission costs dominate and the ratio is higher, but for average or large sized dataset, the test is passed since the amount of transmitted information is mainly lower than 10 times the dataset size.

[3] In POMs 4 and 5, that operate in the encrypted domain, the reference dataset size is defined using the encrypted version of the dataset, using the corresponding key length.

3.6 Scalability assessment

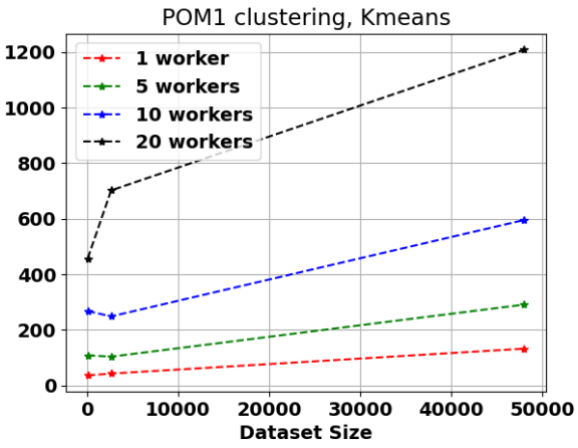
A thorough analysis of the scalability has been performed and the results are detailed in Deliverable 6.2. Here we summarize the results.

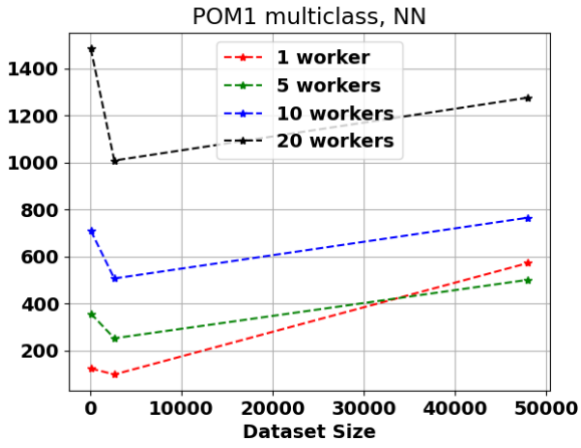
3.6.1 Federated POMS

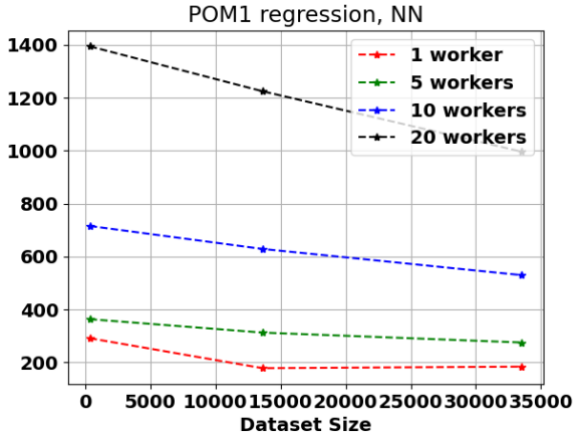
Scalability in terms of amount of data

We have tested every algorithm using 3 different datasets that contain different numbers of training samples.

The next pictures represent the training time as a function of the number of training samples in POM 1. According to the number of features we can observe different behaviours in terms of scalability.







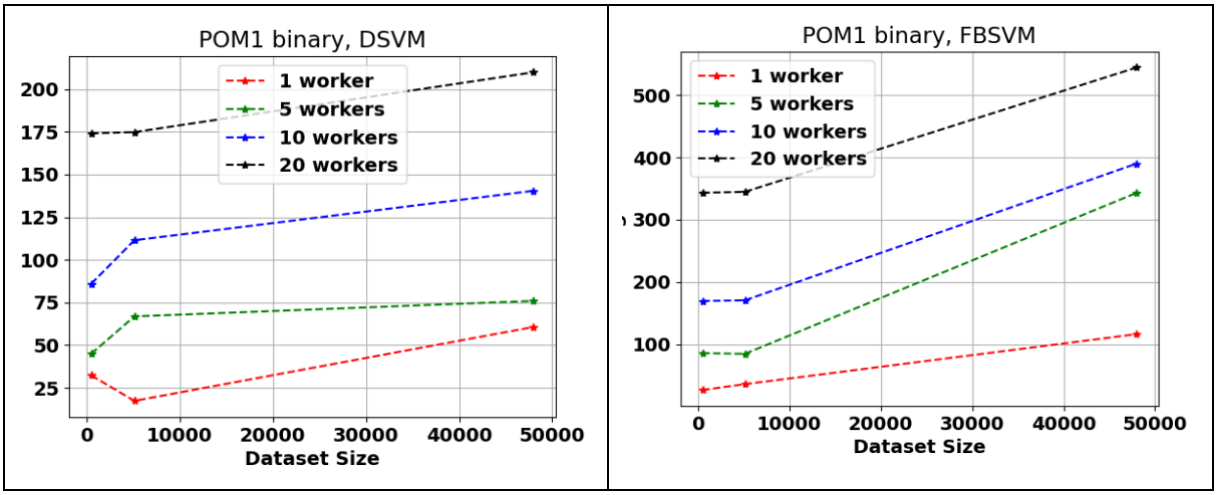
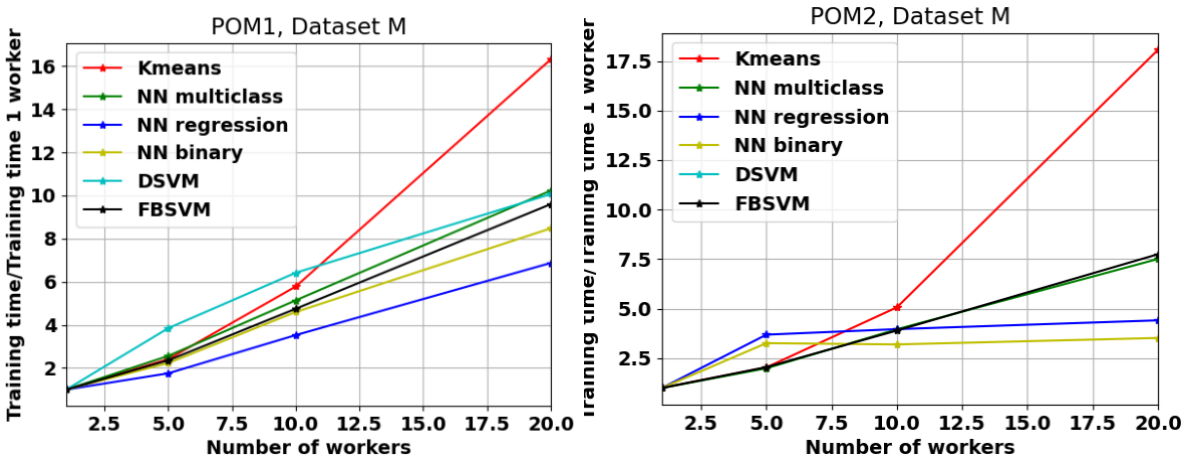


Figure 11. Training time as a function of the number of training samples in POM 1

Typically, the runtime increases with the number of training samples. However, every dataset is different and has associated a different number of training steps until convergence. For this reason, in some cases such as the Neural Networks for regression, we can observe how the runtime decreases because an increase of input features has motivated a decrease in the training iterations.

Scalability in terms of data owners:

The following pictures contain the training time when N workers (data providers) are running divided by the training time when there is only 1 data provider.



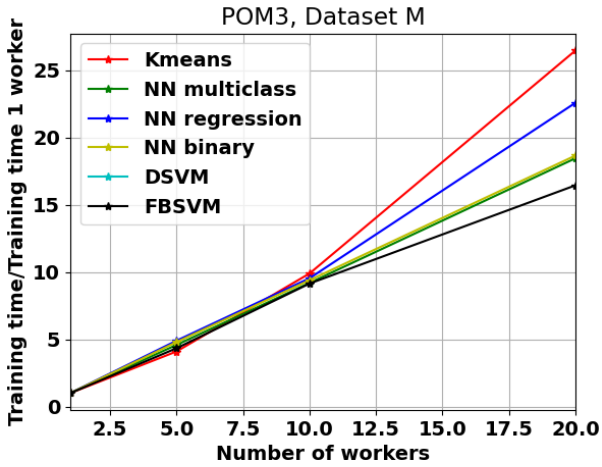


Figure 12. training time when N workers (data providers) are running divided by the training time when there is only 1 data provider

POM1 scales better. It doesn't need encryption and the master node can broadcast the information to the worker nodes at the same time in every iteration. We can see how the training time increases linearly with the number of workers due to the bottleneck associated to receive the information from every worker after every iteration.

In POM2 the scalability decrease respect to POM1. The master node encrypts the information once (the encryption key is the same for every worker) but it needs to decrypt the information received by every worker.

POM3 cannot make use of the broadcast command (and the communication is one of the bottlenecks in federated learning). That is why the runtime is higher and achieves poorer scalability than POM1 and POM2 since in every iteration POM3 has to encrypt and decrypt the information for every worker.

Scalability in terms of input features:

We have tested every algorithm using 3 different datasets that contains different number of input features.

The next pictures represent the training time as a function of the number of input features in POM1. According to the number of features we can observe different behaviours in terms of scalability.

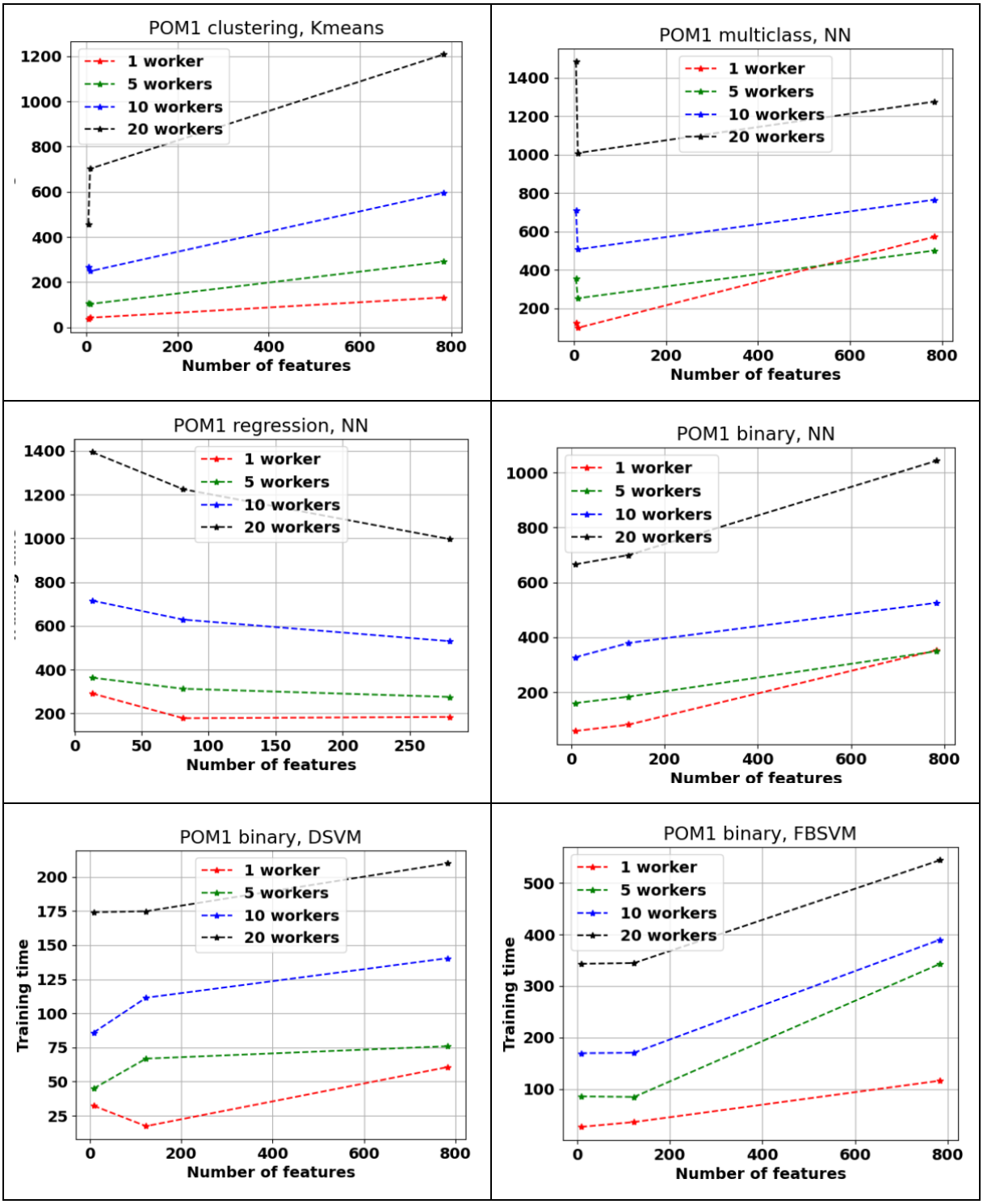


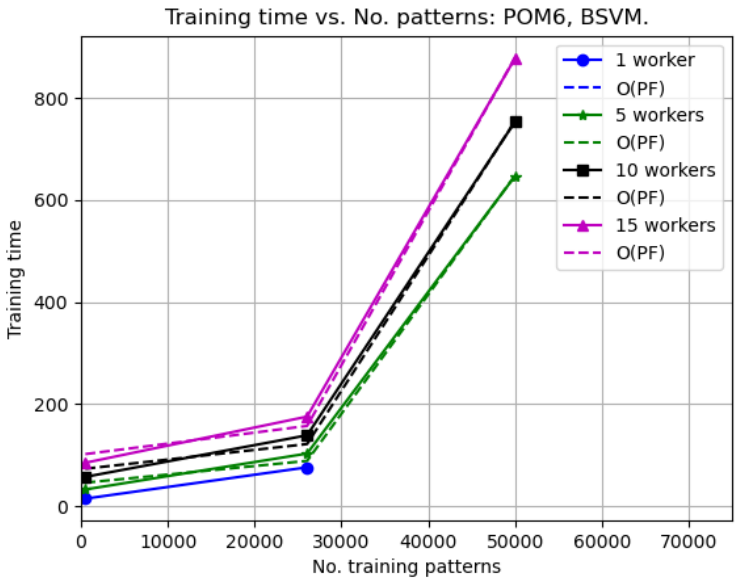
Figure 13. Training time as a function of the number of input features in POM1.

Typically, the runtime increases with the number of input features. However, we are working with different datasets and problems and in the case of Neural Networks for regression, we can observe how the runtime decreases because an increase of input features has motivated a decrease in the training iterations.

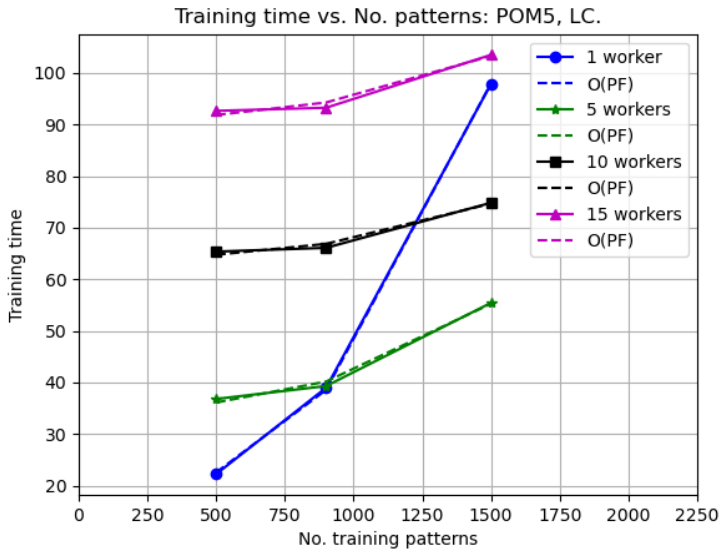
3.6.2 Semi Honest POMS

Scalability w.r.t. No. training patterns

We have measured the training time in datasets with different number of training patterns, as indicated in the next sample Figures (the complete set is available in D6.2):



(a)



(b)

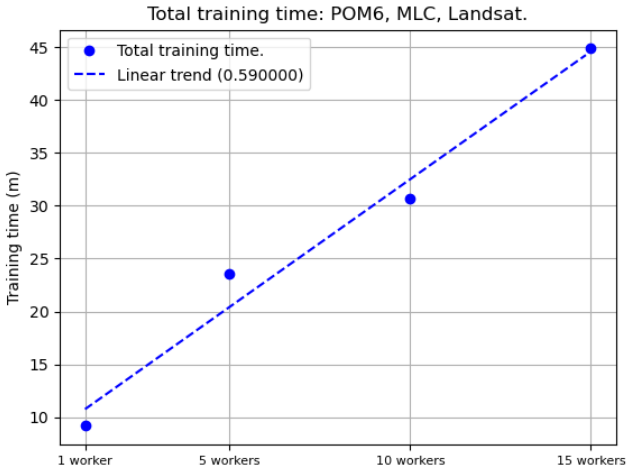
Figure 14. Examples of training time as a function of the No. of training patterns.

Although it may seem at first glance that a quadratic growing may be observed, actually the best fit to these measurements is obtained using a polynomial curve which is a linear

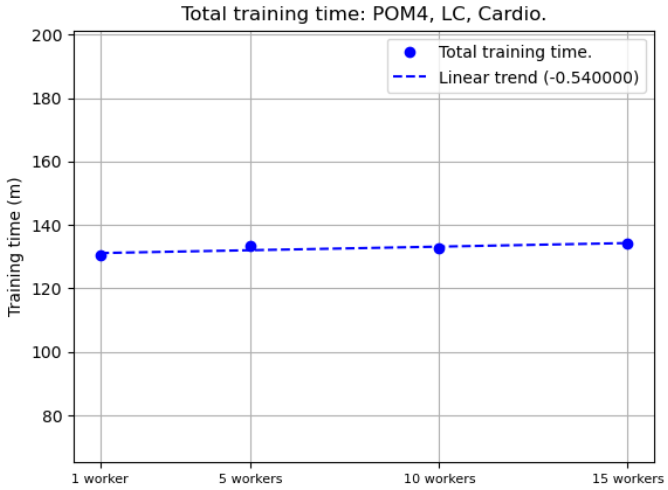
function of both the number of training patterns (P) and the number of input features, i.e. (F), as in Figure 14 (a) and (b). In these cases the complexity is O(PF) and therefore the behaviour is linear with respect to the number of training patterns. For instance, the observed large growth in Figure 14 (a) from 26.000 patterns (Income) to 50.000 (MNIST) is mainly due to the fact that the number of features grows from 107 to 784. In other cases, the complexity has a better fit with respect to the model size (number of centroids, C), as in the case depicted in Figure 14 (c), where be observe a good fit with a complexity estimation of O(PC). Analogous reasoning can be applied to the corresponding figures for most of the algorithms, all of them depicted in D6.2.

Scalability w.r.t. No. workers

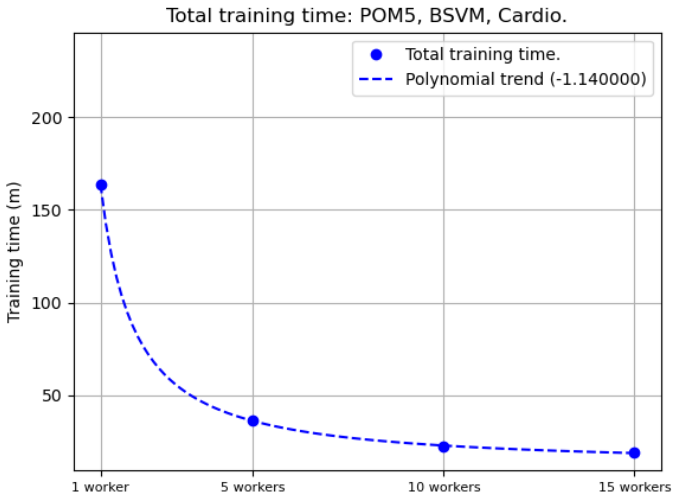
We have plotted the training time of every dataset when an increasing number of workers are used. Some typical observed evolution of training times is depicted in the Figures below:



(a)



(b)



(c)

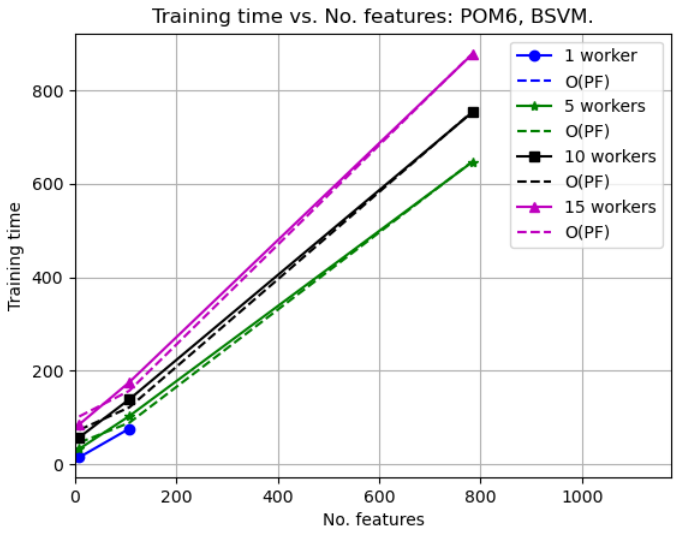
Figure 15. Typical examples of training time as a function of the No. of workers: POM6 in (a), POM4 in (b), POM5 in (c).

We observe in Figure 15 (a) that, in the case of POM6, the training time grows linearly with the number of workers because the protocols used need the cooperation between aggregator and the workers to compute the results, and more workers means more serial interactions between aggregator and workers, plus the needed communications. No quadratic dependence is observed with respect to the number of workers, so the test is positive in these cases. In the POM4 case (b), we observe that the training time is almost independent of the number of workers, since the encrypted data is first transmitted to the aggregator and the training interactions mainly take place between the aggregator and the cryptonode, irrespectively of the number of contributing users. In any case, no quadratic dependence is observed with respect to the number of workers, so the test is also positive.

In the POM5 case (c), we observe that the training time decreases with the number of workers. This is due to the fact that computations take mainly place in the workers, since they operate their local data with the received encrypted model, and more workers means less data in every one of them[4], and therefore the total computation time is reduced, since the needed operations are run in parallel. No quadratic growth is observed with respect to the number of workers, so the test is also positive.

Scalability w.r.t. No. features

We have represented the training time in datasets with different number of input features, as indicated in Figure 16:



(a)

[4] Remember that in all experiments, the same amount of data has been used, so if 2 workers are used, each worker has half the data, for 10 workers, each has 1/10 of the data and so on.

(b)

Figure 16. Examples of training time as a function of the No. of input features.

Again, we observe a linear relationship with both the number of training patterns (P) and the number of input features (F), i.e., the complexity is O(PF) in cases (a) and (b), and complexity O(PC) in case (c). Therefore, the behaviour is also linear with respect to the number of input features, the test being positive in all cases. Further details can be found in D6.2.

4 Use cases perspective

To complete the overview of the MUSKETEER platform in terms of impact and assessment, we organized two interviews with COMAU and Stellantis[5], Biotronics3D and Hygeia[6] to add the "use case perspectives" regarding the platform assessment. Considering that different assessments were already performed regarding use cases (mainly D7.5 "Use case execution and KPI evaluation in the Smart Manufacturing domain" and D7.6 "Use case execution and KPI evaluation in the Health domain"), this interview focused more on the high-level impact, exploitation, and future research areas. Besides, it provides useful dissemination material as expected by the description of the deliverable D8.6. It will be used for the MUSKETEER blog and for partners' dissemination.

[5] The interview was recorded on October 18, 2021. The speakers are Chiara Napione from Comau and Giacomo Fecondo from Stellantis. The interview was conducted by Antoine Garnier & Tobias Prasse from IDSA.

[6] The interview was recorded on October 15, 2021. The speakers are. The interview was conducted by Antoine Garnier & Nora Gras from IDSA.

4.1 Assessment of the MUSKETEER platform from the Smart Manufacturing use case perspective

4.1.1 Completion of the evaluation scenario for the smart manufacturing use case

Chiara Napione (COMAU)

With the use of the MUSKETEER platform, COMAU will have some positive impact on the quality of the welding process and the final product. In fact, **we are now able to learn from data coming from welding robots from different factories**, working in the field. This data is much richer than the one collected only in the COMAU plant before the robots are delivered to the customers. For instance, when a robot is completely new, it doesn't have a lot of problems: we are in a protected environment. On the contrary, when data comes from the factory field, it contains more information about actual failures and in turn we have more possibilities to prevent them.

Moreover, instead of monitoring only some samples of welded points, **customers like Stellantis, will be able to monitor all of them**. The model trained on a large quantity of data coming from different factories, will be very accurate and there will be a serious improvement. Moreover, it will be possible to classify not only a sample but all the welded points.

Giacomo Fecondo (Stellantis)

From my point of view, the most important result is to have understood that **the use of a Federated Learning architecture can improve the welding process**, thanks to the data collection based on collaboration between participants. It should be remembered that, in the case of factories, the introduction of Internet of Things (IoT) did not start from scratch, the so-called greenfield, was built on top of existing deployments, the so-called brownfield. So, with COMAU, one of Stellantis' robot providers we had the possibility with this project to positively evaluate the pros of this kind of data, architecture and new algorithms. Currently we have some non-destructive testing with the ultrasound as mentioned before based on sample analysis. But, as said before, now we can have a full comprehension of the process and the behaviour of the robot thanks to the algorithm. We have a deeper understanding of equipment operations at the welding process. This **will drive future improvements in maintenance**.

About the reduction of the number of person hours needed to configure the robots?

Chiara Napione

Less operators would eventually be involved in the setting process because the model will be able to detect the best set of welding parameters in order to maximize the welding quality.

Does it also mean a reduction of robot maintenance costs?

Chiara Napione

I think so because you can optimize the maintenance. **Analysing the data constantly, you can see when the process is changing and it is going to have a failure,** you can keep track of the slow degradation of the robot, look at it and anticipate the breakdowns. If you have an idea of the degradation of your equipment, you can organize your maintenance in advance and plan to install and repair things, for example.

Giacomo Fecondo

At Stellantis, we might improve our maintenance activities going adopting the **prognostic approach**. In other words, this is not only maintenance happening on a predefined schedule, but a more dynamic and case-by-case approach using AI and based on the behaviour of the robot.

4.1.2 Completion of MUSKETEER objectives

Machine Learning over a high variety of different privacy-preserving scenarios (O.1) How was it to use the different POMs?

Chiara Napione

I think that **you have to find a balance between the privacy you need and the algorithm that you use**, and how much time you can wait. That's because POM's encryption requirements are of course more effective, but their processing might be very long. It depends on how much training time is important for your use case. We chose POM3 for our use case, but we also tried POM1 (less secured) for some tests. The results are similar, but you must wait longer to get them in POM3. For the future a **scheme to simply understand performance of an algorithm over security (and time) might be useful**, because a lot of parameters come into play.

Giacomo Fecondo

I would like to add something. The introduction of the IoT and Industry 4.0 concepts are driving a shift in the context of data generated by the field, and the **convergence of the IT and OT**. Safety and reliability principles for instance, typical of oT environments, are now interweaving with cybersecurity principles from IT environments. POMs are very interesting for us if we carefully chose POMs reflecting our need (depending on the role of a participant, of the aggregate and the level of protection required for the data. In other words, **POMs are very useful here as they offer different privacy levels for adapted to different shopfloor activities**.

Definition of several Privacy Operation Modes (POMs) to provide compliance with the legal and confidentiality restrictions of most industrial scenarios (O1.1)?

Giacomo Fecondo

On the legal aspect, we have found that **privacy related issues** should increase, for example, considering the adoption of human-robot collaboration technologies. **With increasing volume of data, personal data might come into play** and potential identification of employees may become an important problem to face with. Again, the project gave us the possibility to have a look at this topic. For instance, **the rise of co-bots might trigger more personal data related issues** (more sophisticated, they could manage more information related to specific people).

Chiara Napione

Yes, I agree. COMAU sells a lot of different types of equipment. All the machines have access to a lot of data and this might trigger issues because with so many sources, **you get a 360 view on the shopfloor activity. This inevitably raises concerns about privacy.**

About creating predictive models without directly exposing them to the data consumers (O1.2), has it been achieved successful by the project in your opinion? Is the question of accountability important for instance?

Chiara Napione

Ok, so as end-users we feel secured, for two reasons. The first is that a partner in the project, Imperial College (IMP), worked on detecting malicious workers, so **the platform already provides a protection, if participants are malicious**. And the second thing is that the **accountability is also guaranteed by the aggregator of the platform**. For example, if COMAU has the role of aggregator, it offers to participate to a task (on the platform) and sells it as a service. If customers trust the aggregator, that could be enough to ensure trust in the whole system. It **is also in the interest of COMAU, to have honest participants to develop a good/accurate product** in the end.

Complete library of algorithms, having algorithms of different complexity levels (O1.4) Anything missing?

Chiara Napione

Algorithms requested were added. Some other pre-processing algorithms may be added.

Have you noticed any interest for the platform for, from external stakeholders? Especially from SMEs in your network? (O3.3)

Chiara Napione

We observed a big interest in COMAU when we talked about this project. We involved our colleagues and, in their opinion, the idea can be very useful. We didn't talk about this project to our customers or any externals SMEs as far as I'm aware of, but we've plans to use the results internally.

Giacomo Fecondo

We have discussed it for the engineering of prototypes. Today the use of algorithms such as those developed in MUSKETEER are used as offline process, and a further improvement could be to have them in real time. MUSKETEER brought us closer to that possibility. For Stellantis, the offering of such integrated (in the OT) services developed by COMAU and based on MUSKETEER technology could be interesting to evaluate to see in turn how to use them on our side.

Fast deployment, installation, and use (O4.4) How true?

Chiara Napione

The Client Connector from Engineering is not complicated, if you already have the Docker installed, the procedure is pretty standard. Otherwise, you have to install docker and it is a little longer.

Giacomo Fecondo

I agree. Beside minor issues of compliance with internal policies at the beginning, things were straightforward.

4.1.3 Completion of ICT 13 2018-2019 call objectives

Citizens' trust is improved as privacy-aware transparency and control features are increasingly streamlined across data platforms and Big Data applications. Do you feel Citizens' trust was reinforced by mechanisms like those developed in MUSKETEER?

Chiara Napione

I think what is very necessary, are large dissemination activities, to increase awareness, and the meaning of data privacy. There is a lot of work to do yet, because people in companies don't understand, they are still scared in some way, they do not completely trust in big data applications. But I also think that our MUSKETEER project is very focused on data privacy. We propose a secure platform, a privacy preserving platform – by design and the project has been created considering the privacy a primary feature.

Giacomo Fecondo

The adoption of these solutions depends on numerous factors including, of course knowledge, awareness as well as the obvious greater benefits compared to the current situation.

Better value-creation from personal and proprietary/industrial data, what about that?

Chiara Napione

Comparing the model trained on a single robot and the model trained on 2 different robots, we obtained that in the second case, there are significant improvements. **If Federated Learning is already effective on 2 robots, just imagine how it could be adding multiple robots from multiple production lines from multiple factories**. No matter what the factory is producing as you could ask the robot to perform a diagnostic cycle (specific movement of a robot) and learn through it.

And for the long-term perspective about Federated Learning especially?

Chiara Napione

Yes, I think that in the long-term, there will be companies providing platforms allowing data collection from other companies, and I think that makes "digital transformation" a real transformation. This will offer a really better value creation.

Giacomo Fecondo

Federated learning fits well with new factories architecture because we have the edge layer instead of a cloud hosting all the data. The distributed approach works well with edge computing concept.

Any figures that you could think of illustrating this point?

Chiara Napione

Not really, but a lot of companies would become part of the game if they would understand the benefit they could get from such platforms. **They would be eager to share data if they earn some concrete benefit**. Another important point is that there is a big difference between markets like the one in the e-commerce sector, where they are data driven and are used to exploit data, and markets **like automotive, where up to now, nobody has taken real advantage of data sharing**. In those sectors, there will be the highest impact because there is still a lot to do.

4.2 Assessment of the MUSKETEER platform from the Health domain use case perspective

4.2.1 Completion of the evaluation scenario for the smart health domain use case

Improve accuracy of AI algorithms by sharing knowledge from distinct organisations and data repositories, supporting cooperation keeping security and privacy of health data

Joao Correia (Biotronics3D)

Yes, we were able to train several AI models using the tools provided by the partners. We were able to **create models with good accuracy, around 90%** for some of them. We hope now with more data, we can continue to improve these AI models. We are currently running final tests, hopefully improving them again.

Petros Papachristou (Hygeia)

On our side, the direct benefit we can see, and this is the case in general for AI systems, they **relieve workers from repetitive tasks,** radiologists in our case, especially for studies that are classified as "normal". **They increase safety**, they add a safety net if radiologists miss something and detect things that were not detected in their routine since they have a lot of work already. When you use an AI system based on pre-classified data, it gives you the possible findings, stating for instance "this is normal but this area here, you have to see it again". It is probably fine, but it creates this so-called safety net. **It prevents mistakes**. **It is not replacing the radiologist but helping him through the image analysis process**. And additionally, it increases patient safety.

And about this aspect that such technology would enables the growth of the level of research in medical imaging AI tools supported by distributed data repositories

Joao Correia

There is definitely this positive outcome, that we will be able to **continue to work in the prostate cancer specific domain** and try to create new models and improving what we have already achieved. At the same time, **we could start looking to other diseases**, for instance, liver cancer and other cancers for which we can apply similar approaches by using medical imaging to **identify and eventually also classify the lesions existing in other organs of the human body**. The use of the MUSKETEER platform instantiation for healthcare enables us to keep working with Hygiea, and maybe also with the other partners to **continue the research in the area of medical imaging**. So, this is very important for us.

Does it improve also your commercial offer accordingly?

Joao Correia

Yes. At Biotronics 3D (B3D), we offer these medical imaging solutions that already integrate with some third-party AI applications. These are external models that we integrate in 3Dnet, our platform, to enable automatic features identification in medical imaging to support medical diagnosis. And we have lots of customers that are interested in AI and growing their application of AI in their specific cases. **We will be able not only to work with them and do some research but also start the certification of some AI modules** (needed for health sector, i.e. complying with the medical device regulations and future AI regulations) transforming them into products. This opens new opportunities for B3D.

Petros Papachristou

And for industrial productivity, when you have an AI system saying that the image is 99.99% normal, it is indeed, in general, normal. Because the work of radiologists implies a lot of screening, where in most cases there are no findings, such tools are of great help in a lot of cancer detection routines, say breast cancer or pelvis cancer, where you save a lot of work. **It eventually makes it important for both safety and productivity** in the end.

Any improvement that you were not expecting?

Petros Papachristou

Such technologies have opportunities in new areas where people are looking for solutions like breast cancer, as I said before.

4.2.2 Completion of MUSKETEER objectives

Machine Learning over a high variety of different privacy-preserving scenarios (O.1), how was the handling of the different POMs from a user point of view?

Joao Correia

We started by selecting two POMs at the beginning. As the project evolved, we decided to use one of the fastest POMs with the largest number of models available with unlimited complexity considering data shared with the platform is anonymised and that participants are known trusted parties that shall receive the final model. In this POM (POM1) only coefficients of the models are shared with the server, no personal data leave hospitals' premises. POM1 was the best for our use and the fastest.

And about the definition of several Privacy Operation Modes (POMs) to provide compliance with the legal and confidentiality restrictions of most industrial scenarios (O1.1)

Joao Correia

As said before, with only coefficients resulting from the training shared (instead of complete data), this is satisfactory regarding confidentiality restrictions. And moreover, the work done with KU Leven (KUL) on the Data Protection Impact Assessment (DPIA) was a very important

step. It enabled us to clear any potential issues related to data privacy. It provides an interesting basis for the future as a legal tool attached to the use of MUSKETEER instantiations. *(B3D is looking to keep using an instantiation of the MUSKETEER platform).*

Petros Papachristou

On Hygiea's side, first of all, even before the project, we receive the consent from our patients, any patients entering the hospitals, to use their data anonymously for scientific reasons. We have the foundation now, to work in the research area. And the good thing is that the project proved it can be done without exposing the data of our customers while improving algorithms quality through collaboration of different hospitals. **So, even in the minds of the management, or in the minds of the radiologist, it is clear now, that we could fulfil our legal constraints and build useful tools for the daily work at the same time**. It is good for the future projects or future products that we're going to install.

Complete library of algorithms, having algorithms of different complexity levels (O1.4) How useful? What is missing?

Joao Correia

We had many interactions with the partners in charge of this part. As we were testing the solution, they were still developing those libraries, a difficulty we encountered was to have all the algorithms available at the right time for us.

Enhancing data providers to share their data set thanks to the ability of creating predictive models without explicitly giving over their data set (O.3.1) Do you think the solution thought in 2018 is still valid today? New concepts/technologies that could complete and/or replace what you developed?

Joao Correia

From our side, I think this is very still very state of the art as far as we know. We identified other frameworks that enables federated learning, we didn't find anything more advanced than our MUSKETEER platform.

And about accountability, Mark from IBM mentioned it as a potential improvement (avoiding the platform to be complete black box for its users)?

Joao Correia

Well, the way we think the implementation **in health is that all partners that will be providing their updates to the training of AI models will be partners that know each other** and that agree to participate in these projects. So, in health care, we don't see this model of participating and providing data without knowing who the partners are and who is building

the model and so on. This will be even more so considering that in the end, the **AI models need to be certified, it is very important to know the provenance of that data**.

Allowing to measure the impact of every data owner on the accuracy of the predictive models (O3.2) How important is that for you?

Joao Correia

I think yes, it is important. With so many start-ups and companies trying to access data, measuring the impact could be interesting - to see how to compensate entities that provide the better part of the training. We have other projects, where data is compensated through a fee to hospitals whenever a model based on their data is built (in that specific case, the rewarding is only based the hospital participation, not the specific impact of its data).

About European SMEs involvement (O3.3) Did you witness anything? From externals in your network

Joao Correia

So, for instance, we have some partners developing AI tools that are very interested in the possibility to train models offering large infrastructure + access to data without legal infringement. It is very interesting. **Many AI developers are currently using mostly public data sets.** Platforms (as **MUSKETEER) offer ways to go beyond that limit** offering access to different types of populations for instance (some **data sets** are very specific about certain people, and countries).

About allowing interoperability with Big Data frameworks by providing portability mechanisms (O4.2)? Let's take the example of Gaia-X?

Joao Correia

It is indeed very important for us. In the health sector, we have DICOM (image) and HL7 (any other documents) standards. Interoperability of AI models, and pre-processing methods are very important for the completion of trainings. 3Dnet, our platform, is already a cloud solution relying on different cloud providers. In the last months, we've been looking to Gaia X, as they are offering solutions non-US based. Important as it might be required from European hospitals, we're interested in.

4.2.3 Completion of ICT 13 2018-2019 call objectives

Personal data protection is improved, and compliance with the General Data Protection Regulation (and other relevant legislation) is made easier for economic operators

Joao Correia

Yes. We feel comfortable with that aspect. **In the health sector, we already have very strict directives and regulations** in the way that the personal data is processed. So, I mean, for us, of course, there are new important requirements in the GDPR, but most of the challenges regarding data, privacy, and security were already under scrutiny/ tackled in the health sector. Additionally, of course, within the project, the DPIA was a very important exercise and tool for the future.

Christina Kotsiopoulou (HYGEIA Hospital – InteropEHRate)

On our side I don't think we're going to face any problems, because from the moment that the patient enters a hospital, we are compliant to the GDPR. Additionally, as said by Joao, the help of KUL with the DPIA made sure that we were completely fine according to the GDPR.

Citizens' trust is improved as privacy-aware transparency and control features are increasingly streamlined across data platforms and Big Data applications.

Joao Correia

Radiologists and medical staff are more confident with the processing of the data now (about the fact that the data of their patients are not shared or leave the hospital). **From the patient perspective, it is important they learn about this project**, and show that the industry is taking care of the privacy aspect and their data is used in a GDPR compliant way.

Christina Kotsiopoulou

It is important to **ensure that from the patient's point of view, their data is secured**. They know beforehand that a protocol is in place to protect their data, treated accordingly to GDPR.

Better value-creation from personal and proprietary/industrial data. On the long-term perspective?

Petros Papachristou

AI algorithms need a lot of data to be trained and MUSKETEER enables collaboration and access to these large troves of data (in order to improve and validate algorithms). It is already a great achievement. Another point, by-product of the project let's say, is that usually the data is not formatted correctly. But now, knowing the power of the technology, radiologists are "incentivized" so to say to report in a structured way (to benefit from large data sets, and eventually great new tools)

Joao Correia

In the UK there is a lack of radiologists, professionals able to interpret images and write reports. **It is important to have new tools to enable quicker identification of points of interest and allow radiologists to report about more patients**. **AI 'solutions are needed to improve the daily workflow supporting doctors to make quicker and better decisions**. This is **important for the patient outcome ensuring no one is left behind**. True in the EU.

Increase in the number of data provider organisations in the personal and industrial data platforms, do you see any improvement here thanks to MUSKETEER?

Petros Papachristou

On the contrary, as a group of hospitals, with Hygeia we're interested in exploiting it as the competitive advantage!

5 Conclusion

This deliverable presented a general evaluation and impact assessment of the MUSKETEER platform. After the introduction in Chapter 1, Chapter 2 recalled the different objectives assigned to the MUSKETEER project. Then, Chapter 3 presented the assessment of the MUSKETEER platform form an implementation and technical point of view. Main technical findings of the project include the importance of a sound software engineering, conceived from the beginning to provide integration, scalability and openness (of interfaces), the development of a trustworthy security architecture through: sound security architecture of the platform, use of encryption to protect data confidentiality, robustness to data perturbation attacks at both training and deployment time and the demonstration of scalability (depending on platform features used and context of use) but sufficient to show application in realistic use cases. Chapter 4 added the assessment point of view of the two use cases to complete this evaluation and impact assessment. Main business findings include the interest of the platform as an enabler allowing to train models that would be difficult to train otherwise, with evidence of a clear business advantage, even if we are only in the early stages of discovery and adoption of this technology with a lot more to gain. Finally, the project showed some directions for future work including maintaining data provenance/accountability, certification, robustness to novel/emerging forms of attacks. Chapter 5 is concluding the deliverable.

6 References

[Du_2002] Du, Wenliang & Zhan, Zhijun. (2002). A Practical Approach to Solve Secure Multi-Party Computation Problems. Proceedings NSPW'02: New Security Paradigms Workshop.

[Ghorbani2019] Ghorbani, A., Zou, J.: Data shapley: Equitable valuation of data for m chine learning. In: Proc.36th International Conference on Machine Learning, PMLR, vol. 97, p. 2242–2251 (2019)

[Goldreich_1987] Goldreich, Oded & Micali, S & Wigderson, Avi. (1987). How to play ANY mental game. 218-229.

[Gonzalez_2017] Francisco-Javier González-Serrano, Angel Navia-Vázquez and Adrián Amor-Martín. Training Support Vector Machines with privacy-protected data. Pattern Recognition. Vol. 72, pp. 93-107, 2017.

[Mohassel_2011] P. Mohassel. 2011. Efficient and Secure Delegation of Linear Algebra. IACR Cryptology ePrint Archive 2011/605.

[Muñoz-González et al. 2019] Muñoz-González, Luis, Kenneth T. Co, and Emil C. Lupu. "Byzantine-Robust Federated Machine Learning through Adaptive Model Averaging." arXiv preprint arXiv:1909.05125, 2019.

[Shapley1953] Shapley., L.S.: A Value for n-person Games. In: Annals of Mathematical Studies: contributions to the Theory of Games, vol. 28, p. 307–317. Princeton University Press (1953)

[Song_2019] T. Song, Y. Tong and S. Wei, "Profit Allocation for Federated Learning,"
2019 IEEE International Conference on Big Data (Big Data), Los Angeles, CA, USA, 2019

[Wang_2011] Wang C, Ren K, Wang J. Secure and practical outsourcing of linear programming in cloud computing. 2011 Proceedings IEEE INFOCOM, Shanghai, China, 2011; 820–828. DOI: 10.1109/INFCOM.2011.5935305.

[Wang_2015] Yulong Wang and Yi Li. An efficient and tunable matrix-disguisin method toward privacy-preserving computation. Security Comm. Networks. 8:3099–3110. 2015.

[Yao_1986] Andrew C. Yao. How to generate and exchange secrets. In Proc. 27th IEEE Symp. on Foundations of Comp. Science, pages 162–167, Toronto, 1986.

[Zhu_2015] Youwen Zhu, Tsuyoshi Takagi. Efficient scalar product protocol and its privacy-preserving application. Int. J. Electronic Security and Digital Forensics, Vol. 7, No. 1, 2015.