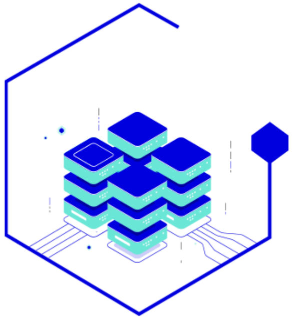
SOTERIA, user-friendly secured
personal data management platform



Towards a Decentralized ML-enabled Data Vault

Luis S. Luévano García, PhD
Postdoctoral Researcher, WIDE team, Inria Rennes
Davide Frey, PhD
Senior Researcher, WIDE team, Inria Rennes

Summary

01

Platform Description

SOTERIA Objective

- To combine a **high-level identification tool** with a **decentralized secured data storage** tool

- To enable **all citizens to fully protect and control their personal data** with awareness on potential privacy risks

Versions:

- Centralized and decentralized approaches

SOTERIA Centralized version

High-level identification

- Allow the creation of a digital identity for a centralized authentication

Personal data protection

- Give citizens the control over their personal data.

Co-creation approach

- Develop a platform meeting European citizens' needs and expectations to maximize its acceptability

In-cloud data wallet for MPC/ML

Highlights

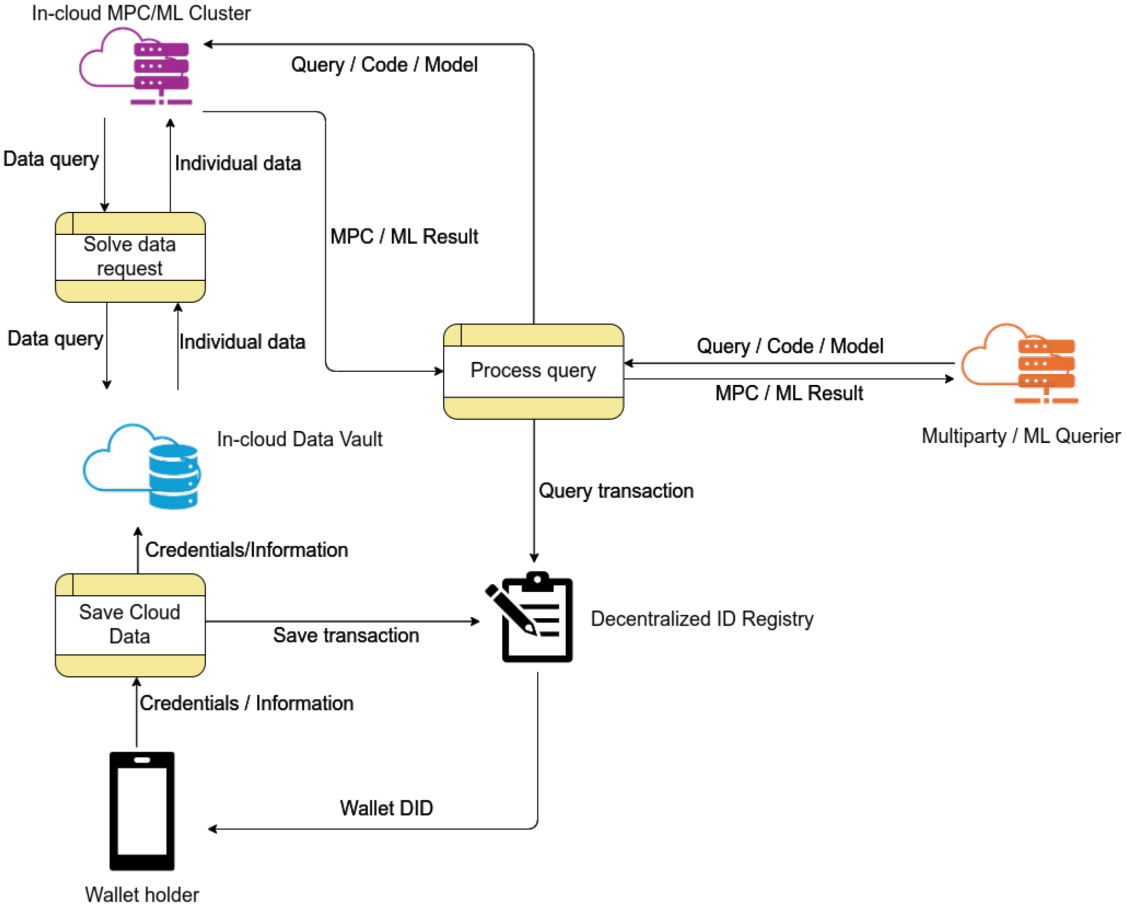
- Retain **pseudoanonymity** and **unlinkability**
- Empower **user control** over their data
- Minimize of the **data shared** with service providers
- Personal data protected by advanced **cryptography** and **privacy** techniques

Centralized Design for MPC/ML tasks

Entities and tasks

- **Wallet holder**
  - Sends IDs and information to the In-Cloud Data Vault
- **In-Cloud Data Vault**
  - Receives and stores information from Wallet Holder
  - Responds requests from MPC / ML Cluster
- **In-Cloud Multiparty Computation / Machine Learning Cluster**
  - Receives MPC / ML queries and performs computations
  - Requests data from In-Cloud Data Vault
- **Multiparty Computation / Machine Learning Querier**
  - Requests a Multiparty Computation or Machine Learning service
- **Decentralized ID registry**
  - Holds registry to communicate, lookup, and register queries for Wallet holders.

High-level centralized data flow



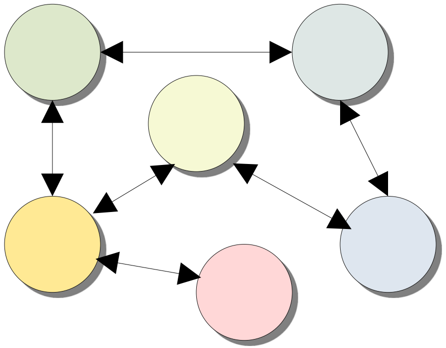
02

Decentralized version

Motivations for decentralization

No central entity coordinating computations and data

- **Private data stays** in each individual's devices
- **Less** reliance on **external services**
- **No centralized target** for attackers
- **Less** reliance on **expensive** computing infrastructure



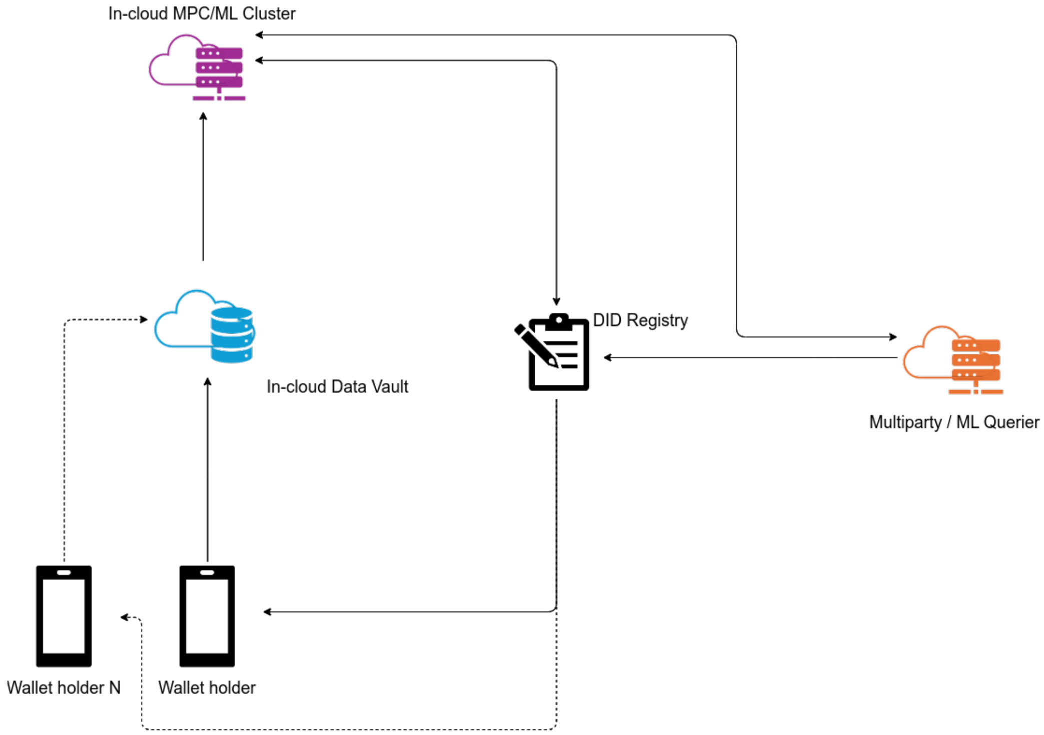
Security & Privacy considerations

- **Communication protocol**
  - > Encryption and authentication
- **Neighbor selection and Topology**
  - > Logical neighbors and effective neighbors
- **TEE availability and alternatives**
  - > Secure hardware availability and encryption
  - > Protection from side-channel attacks
- **Aggregation / Learning algorithm**
  - > Secure aggregation
  - > Noise-based algorithms
- **Model parameter updates**
  - > Privacy, Accuracy, Efficiency
- **Metrics**
  - > Privacy, Accuracy, Efficiency
- **Training paradigms and models**
  - > Lightweight models, Knowledge Distillation, Quantization, Privacy-aware regularization
  - > Limiting computation layers

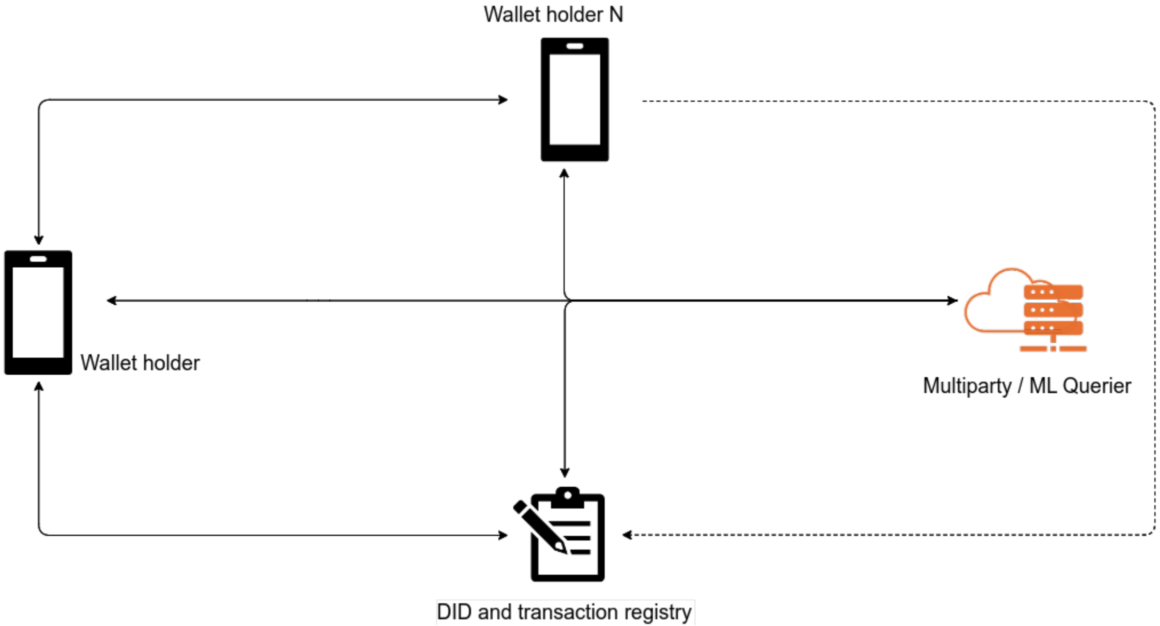
03

Decentralized Data Vault for MPC/ML

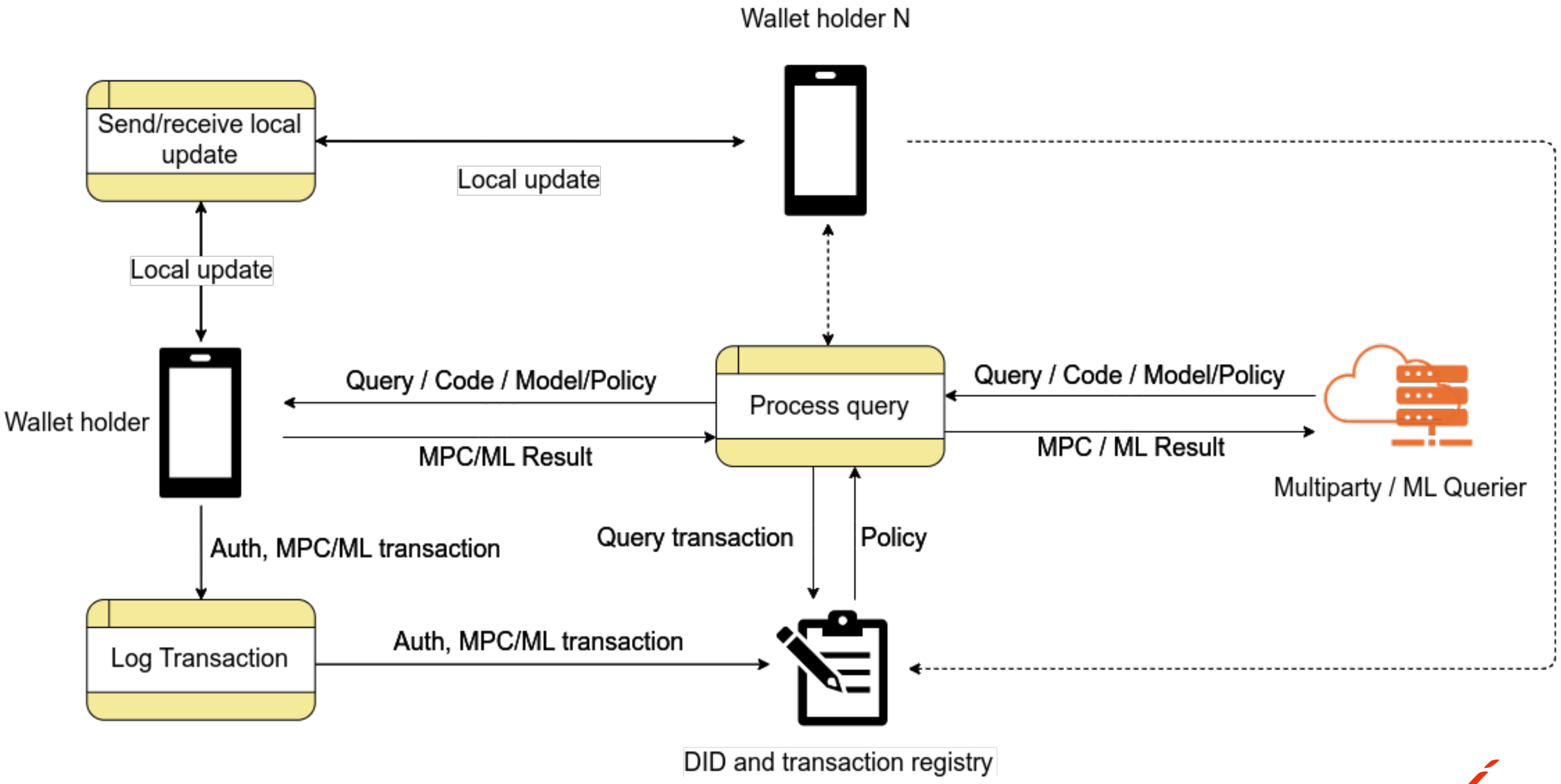
Centralized scenario

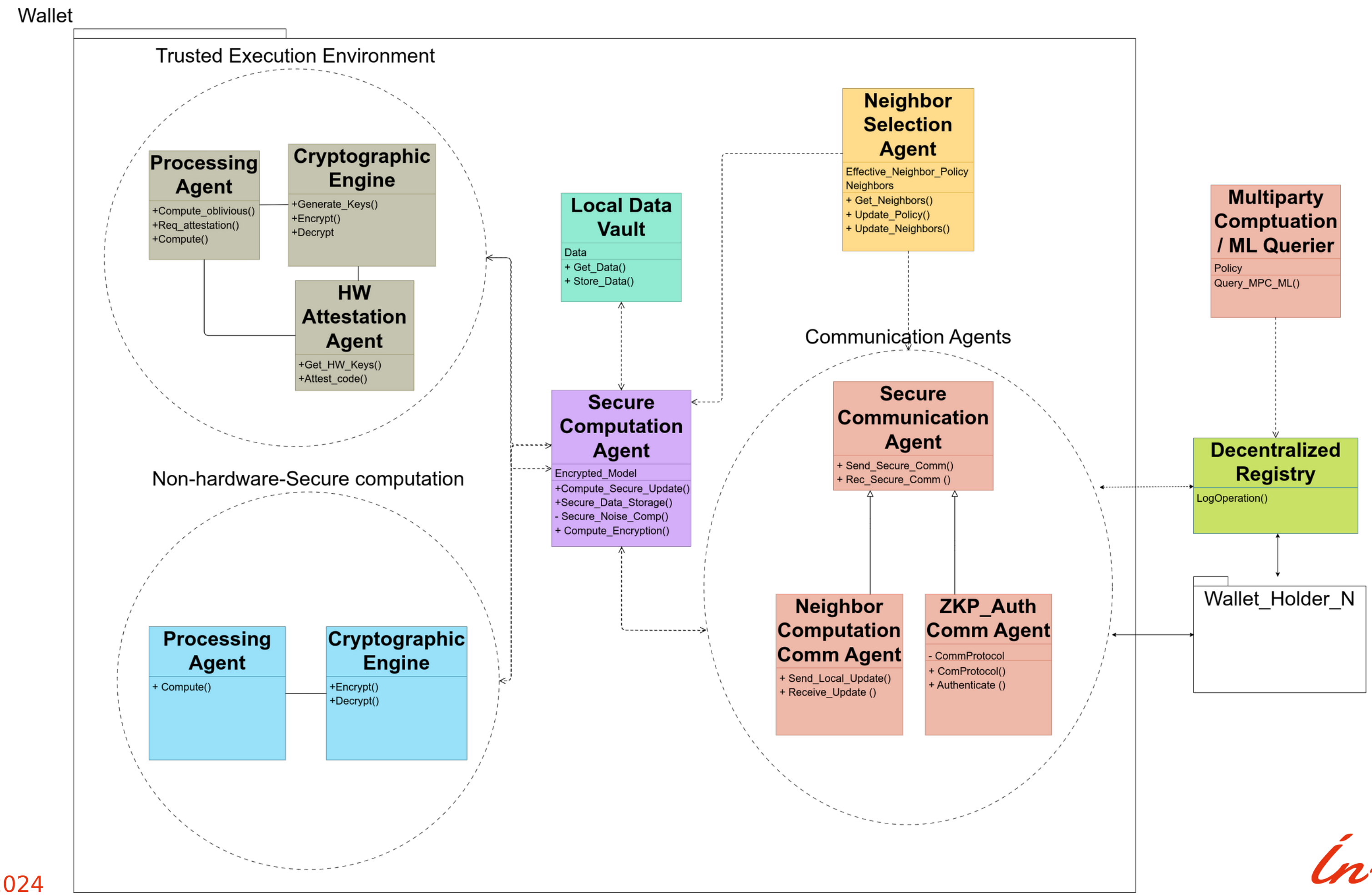


Decentralized scenario



Decentralized data flow




Wallet
Trusted Execution Environment
Processing Agent
+Compute_oblivious()
+Req_attestation()
+Compute()
Cryptographic Engine
+Generate_Keys()
+Encrypt()
+Decrypt
HW Attestation Agent
+Get_HW_Keys()
+Attest_code()
Local Data Vault
Data
+ Get_Data()
+ Store_Data()
Neighbor Selection Agent
Effective_Neighbor_Policy
Neighbors
+ Get_Neighbors()
+ Update_Policy()
+ Update_Neighbors()
Multiparty Comptuation / ML Querier
Policy
Query_MPC_ML()
Communication Agents
Secure Computation Agent
Encrypted_Model
+Compute_Secure_Update()
+Secure_Data_Storage()
- Secure_Noise_Comp()
+ Compute_Encryption()
Secure Communication Agent
+ Send_Secure_Comm()
+ Rec_Secure_Comm ()
Decentralized Registry
LogOperation()
Non-hardware-Secure computation
Processing Agent
+ Compute()
Cryptographic Engine
+Encrypt()
+Decrypt()
Neighbor Computation Comm Agent
+ Send_Local_Update()
+ Receive_Update ()
ZKP_Auth Comm Agent
- CommProtocol
+ ComProtocol()
+ Authenticate ()
Wallet_Holder_N

Security & Privacy considerations (detailed)

Possible approaches

- **Communication protocol**
  - Pseudoanonymization, DHT, ZKP, SSI
- Neighbor selection and Topology
  - Epidemic protocols, Dynamic topologies
- TEE availability and alternatives
  - Intel SGX & TXT, ARM TZ, AMD, Apple iOS Secure Enclave
  - Encrypted layer computation, HME
- Aggregation / Learning algorithm
  - Secure Aggregation
  - Noise-based algorithms
- Model parameter updates
  - Fine-tuning, hierarchical aggregation, random walks, convergence
- Metrics
  - Accuracy: Differential privacy
  - Efficiency: FLOPs, Latency, No. Parameters
  - Privacy: Differential privacy, attack precision
- Training paradigms and models
  - Lightweight Neural Networks
  - Quantization, Privacy-aware regularization, Adversarial training, Knowledge Distillation
  - Limiting computation layers

Communication protocol

Authentication

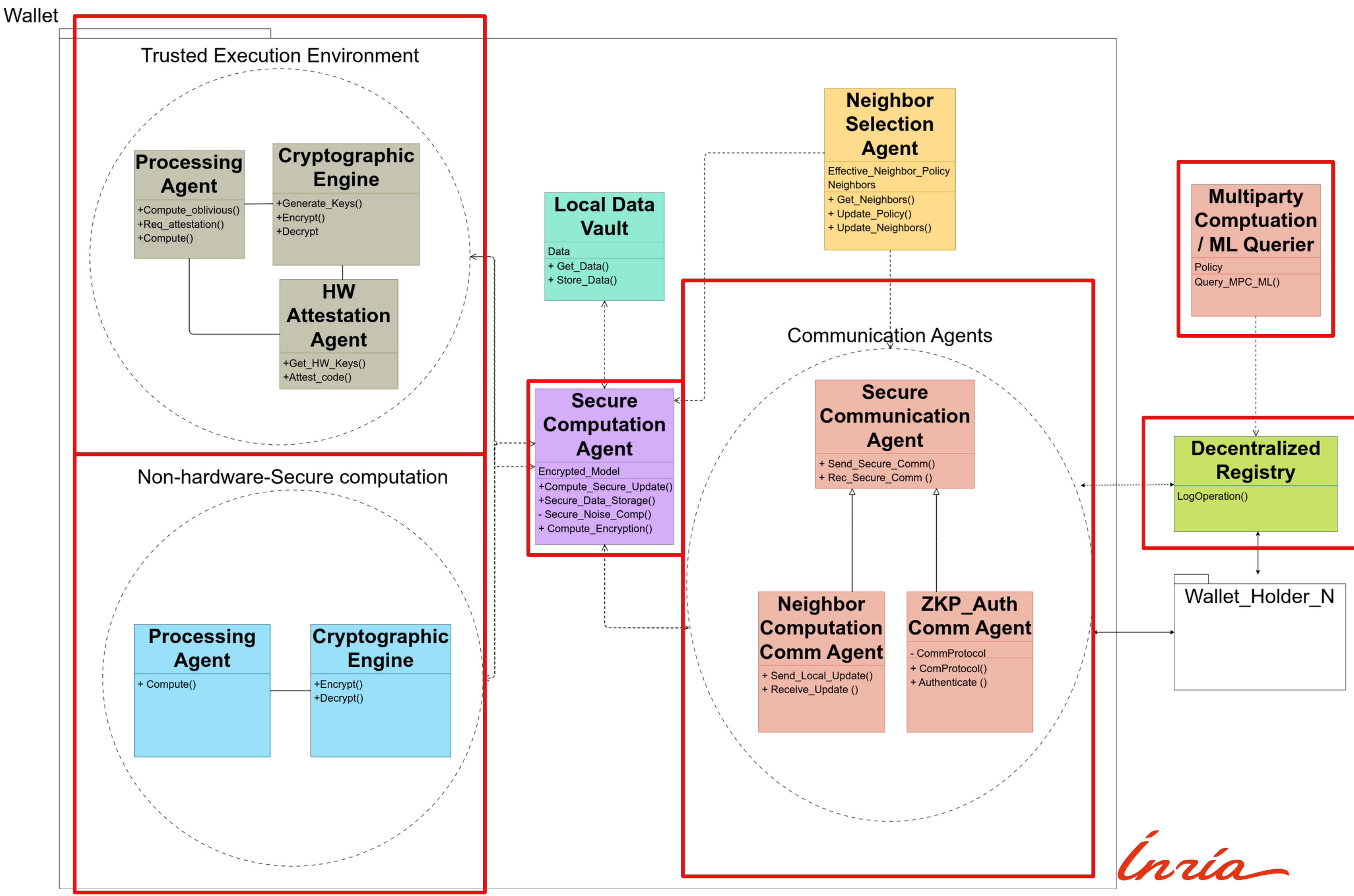
- TLS Encrypted communications
- Pseudoanonymity by:
  - Distributed ID and registry
  - Zero Knowledge Proof
  - Self-Sovereign Identity with Access Control Lists[1]

Distributed objects

- Access Control Lists
- Distributed Hash Tables
- May require concensus depending on the object
- Enables scenarios for :
  - E-voting
  - Key-management systems
  - Money transfers

[1]D. Frey, M. Gestin, and M. Raynal. The Synchronization Power (Consensus Number) of Access-Control Objects: the Case of AllowList and DenyList. In 37th International Symposium on Distributed Computing (DISC 2023). Leibniz International Proceedings in Informatics (LIPIcs), Volume 281, pp. 21:1-21:23, Schloss Dagstuhl – Leibniz-Zentrum für Informatik (2023) https://doi.org/10.4230/LIPIcs.DISC.2023.21

- **Communication protocol**



Security & Privacy considerations (detailed)

Possible approaches

- Communication protocol
  - Pseudoanonymization, DHT, ZKP, SSI
- **Neighbor selection and Topology**
  - Epidemic protocols, Dynamic topologies
- TEE availability and alternatives
  - Intel SGX & TXT, ARM TZ, AMD, Apple iOS Secure Enclave
  - Encrypted layer computation, HME
- Aggregation / Learning algorithm
  - Secure Aggregation
  - Noise-based algorithms
- Model parameter updates
  - Fine-tuning, hierarchical aggregation, random walks, convergence
- Metrics
  - Accuracy: Differential privacy
  - Efficiency: FLOPs, Latency, No. Parameters
  - Privacy: Differential privacy, attack precision
- Training paradigms and models
  - Lightweight Neural Networks
  - Quantization, Privacy-aware regularization, Adversarial training, Knowledge Distillation
  - Limiting computation layers

Neighbor selection and topology

Neighbor selection

- Gossip/epidemic protocol-based communication[1]
- Dynamic view changes
- Balancing number of effective neighbors
- Policy for enforcing individual privacy

Topology

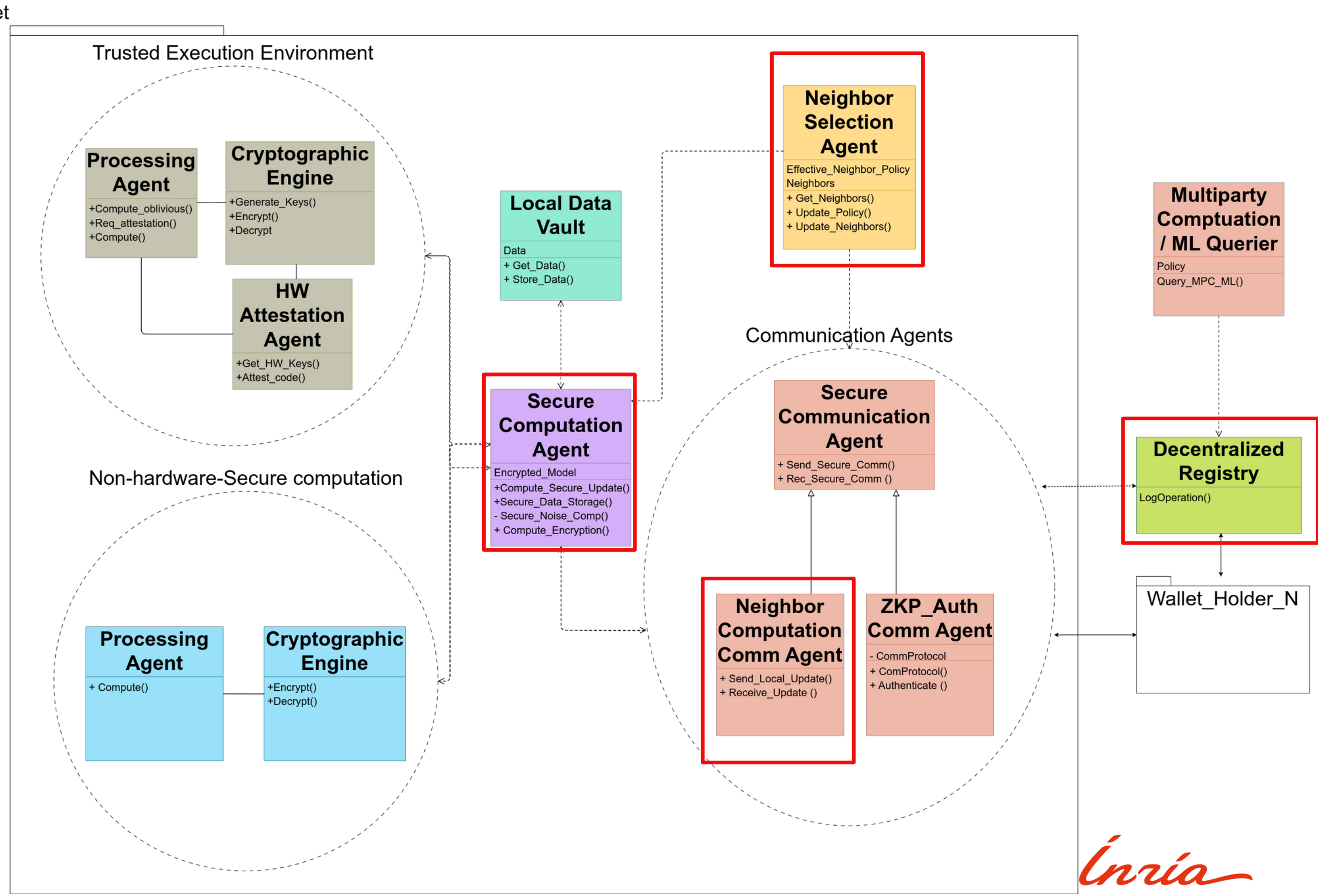
- Time Varying Exponential
- Dynamic addition of members

- Impact on:
  - Accuracy and convergence[2]
  - Privacy per number of connections
  - Communication latency

(1) C. Georgiou et al. 2008. On the complexity of asynchronous gossip. In Proceedings of the twenty-seventh ACM symposium on Principles of distributed computing (pp. 135-144).

(2) T. Vogels et al. 2022. Beyond spectral gap: The role of the topology in decentralized learning. Advances in Neural Information Processing Systems, 35, 15039-15050.

- **Neighbor selection and Topology**



Security & Privacy considerations (detailed)

Possible approaches

- Communication protocol
  - Pseudoanonymization, DHT, ZKP, SSI
- Neighbor selection and Topology
  - Epidemic protocols, Dynamic topologies
- **TEE availability and alternatives**
  - Intel SGX & TXT, ARM TZ, AMD, Apple iOS Secure Enclave
  - Encrypted layer computation, HME
- Aggregation / Learning algorithm
  - Secure Aggregation
  - Noise-based algorithms
- Model parameter updates
  - Fine-tuning, hierarchical aggregation, random walks, convergence
- Metrics
  - Accuracy: Differential privacy
  - Efficiency: FLOPs, Latency, No. Parameters
  - Privacy: Differential privacy, attack precision
- Training paradigms and models
  - Lightweight Neural Networks
  - Quantization, Privacy-aware regularization, Adversarial training, Knowledge Distillation
  - Limiting computation layers

Trusted Execution Environments and alternatives

Implementations

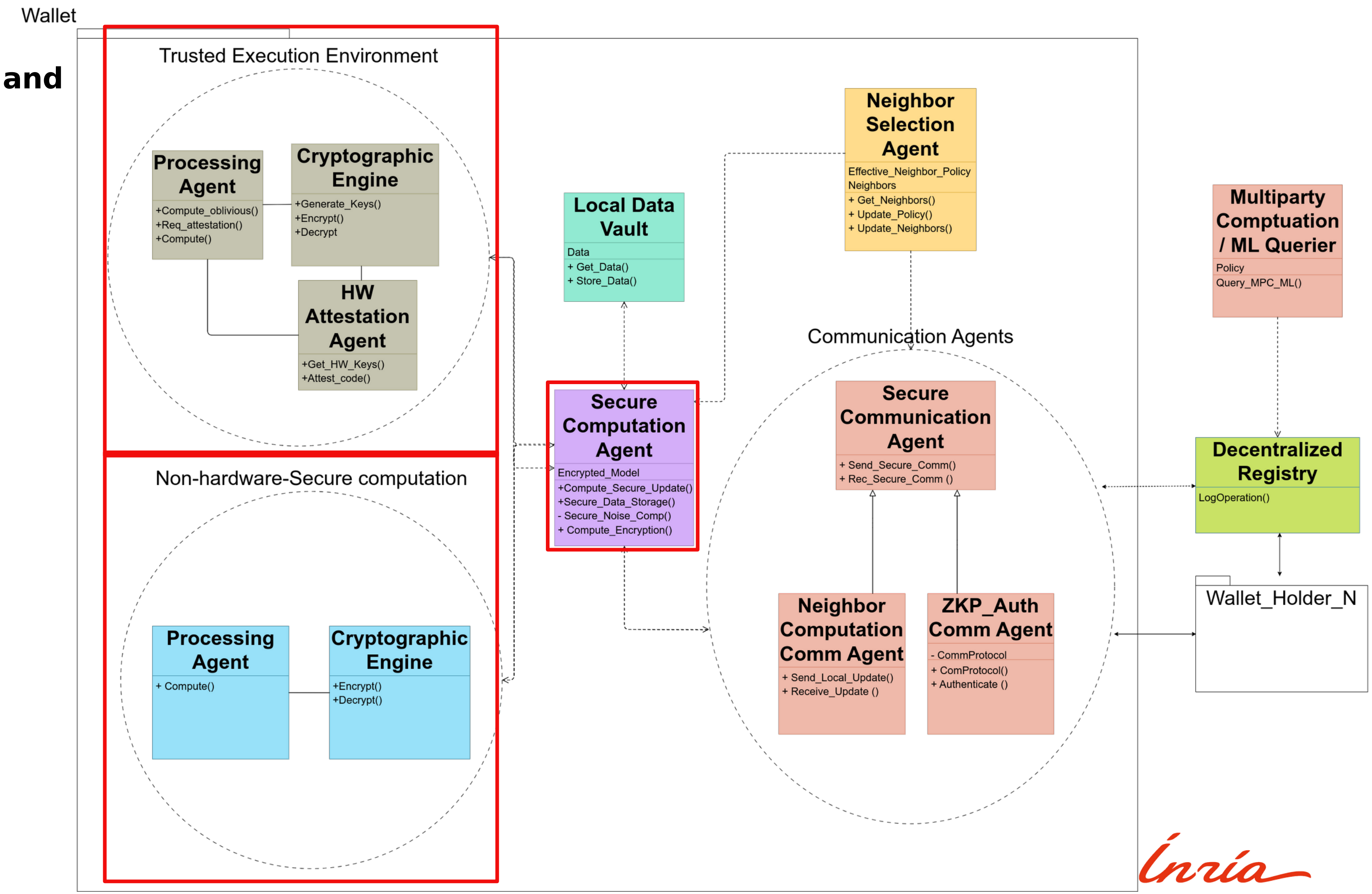
- Intel SGX & TXT, ARM TZ, Apple iOS SE, AMD
- Applications
  - Local and remote attestation[1]
  - Isolated computation
  - Cryptographic services
  - Control Flow Attestation[3]

Considerations

- Availability & platform restrictions
- Encryption-based alternatives on TEE unavailability[4]
- Continued research on vulnerabilities
  - Mitigate side-channels[2]
- Trade-offs :
  - Computation overhead
  - Limited computing resources

(1) Intel. 2023. Attestation & Provisioning Services Intel Software Guard Extensions. hhttps://www.intel.com/content/www/us/en/developer/tools/software-guard-extensions/attestation-services.html
(2) K. Fumiyuki et al.. 2023. OLIVE: Oblivious Federated Learning on Trusted ExecutionEnvironment against the risk of sparsification. arXiv:2202.07165 [cs.LG]
(3) M. Morbitzer et al. 2022. GuaranTEE: Introducing Control-Flow Attestation forTrusted Execution Environments. arXiv:2202.07380 [cs.CR]
(4) K. Cheng et al. 2023. Manto: A Practical and Secure Inference Service of Convolutional Neural Networks for IoT. IEEE Internet of Things Journal. PP. 1-1. 10.1109/JIOT.2023.3251982.

- **TEE availability and alternatives**



Security & Privacy considerations (detailed)

Possible approaches

- Communication protocol
  - Pseudoanonymization, DHT, ZKP, SSI
- Neighbor selection and Topology
  - Epidemic protocols, Dynamic topologies
- TEE availability and alternatives
  - Intel SGX & TXT, ARM TZ, AMD, Apple iOS Secure Enclave
  - Encrypted layer computation, HME
- **Aggregation / Learning algorithm**
  - Secure Aggregation
  - Noise-based algorithms
- Model parameter updates
  - Fine-tuning, hierarchical aggregation, random walks, convergence
- Metrics
  - Accuracy: Differential privacy
  - Efficiency: FLOPs, Latency, No. Parameters
  - Privacy: Differential privacy, attack precision
- Training paradigms and models
  - Lightweight Neural Networks
  - Quantization, Privacy-aware regularization, Adversarial training, Knowledge Distillation
  - Limiting computation layers

Aggregation / Learning algorithm

Masking with Lossless noise

- No accuracy loss
- Global masking[1]
  - Centralized scenarios.
  - Requires cooperation by all nodes
- Local masking
  - Additional communications
  - Must trust neighbors

Noise injection[2]

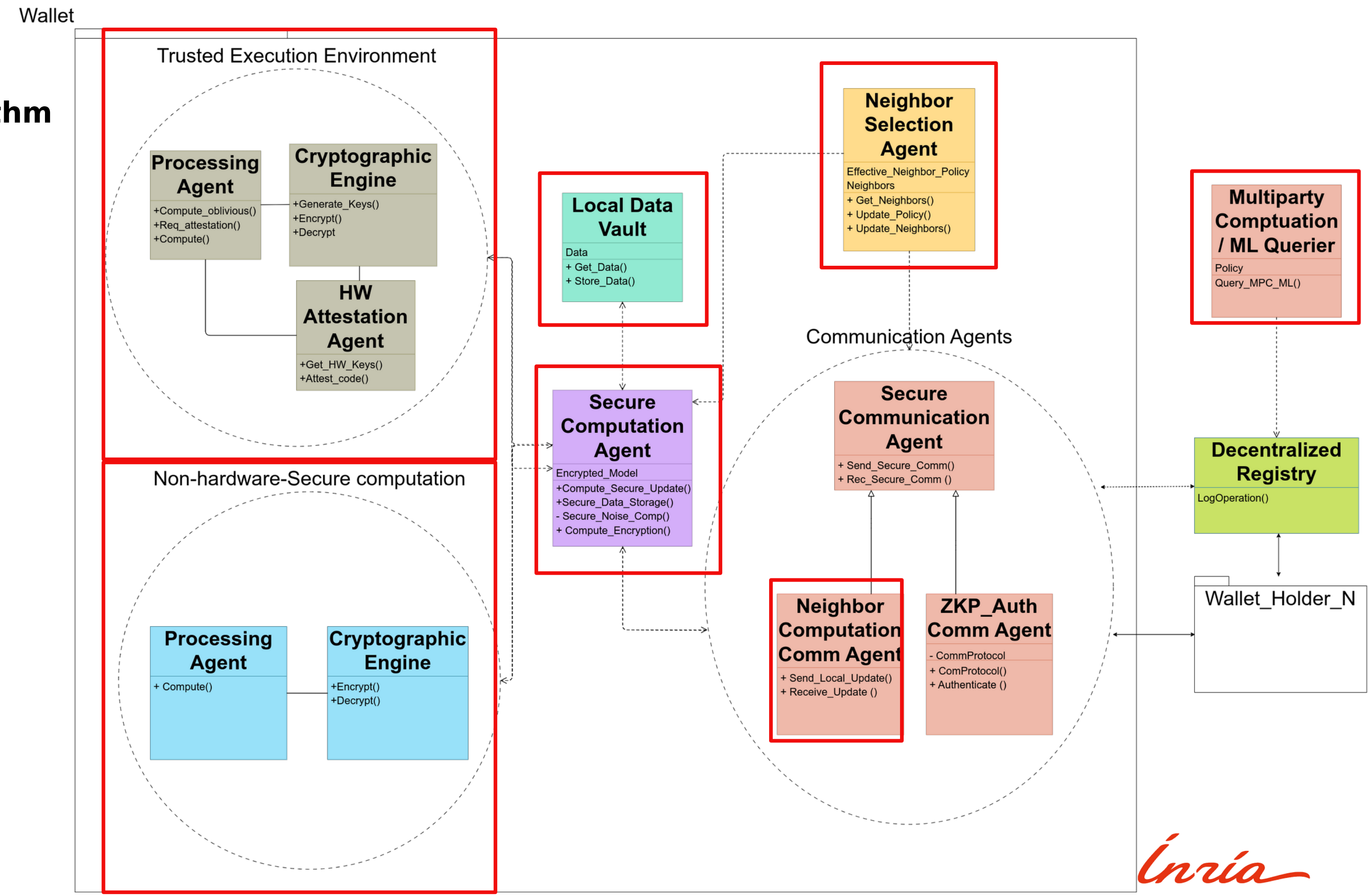
- SGD can manage noisy models
- Differential Privacy[3] as a "gold standard"
- Trade-offs :
  - Lower accuracy
  - Longer training times

Secure aggregation

- Filtering updates from malicious clients[4]

[1] Bonawitz, K.; Ivanov, V.; Kreuter, B.; Marcedone, A.; McMahan, H. B.; Patel, S.; Ramage, D.; Segal, A.; Seth, K. Practical Secure Aggregation for Privacy-Preserving Machine Learning. In ACM SIGSAC, 2017.
[2] Cyffers, E.; Even, M.; Bellet, A.; Massoulié, L. Muffliato: Peer-to-Peer Privacy Amplification for Decentralized Optimization and Averaging. Advances in Neural Information Processing Systems 2022, 35, 15889–15902.
[3] Dwork, C.; Smith, A.; Steinke, T.; Ullman, J. Exposed! A Survey of Attacks on Private Data. Annu. Rev. Stat. Appl. 2017, 4 (1), 61–84.
[4] Tramer, F., & Boneh, D. (2018). Slalom: Fast, verifiable and private execution of neural networks in trusted hardware. arXiv preprint arXiv:1806.03287.

- **Aggregation / Learning algorithm**



Security & Privacy considerations (detailed)

Possible approaches

- Communication protocol
  - > Pseudoanonymization, DHT, ZKP, SSI
- Neighbor selection and Topology
  - > Epidemic protocols, Dynamic topologies
- TEE availability and alternatives
  - > Intel SGX & TXT, ARM TZ, AMD, Apple iOS Secure Enclave
  - > Encrypted layer computation, HME
- Aggregation / Learning algorithm
  - > Secure Aggregation
  - > Noise-based algorithms
- **Model parameter updates**
  - > Fine-tuning, hierarchical aggregation, random walks, convergence
- Metrics
  - > Accuracy: Differential privacy
  - > Efficiency: FLOPs, Latency, No. Parameters
  - > Privacy: Differential privacy, attack precision
- Training paradigms and models
  - > Lightweight Neural Networks
  - > Quantization, Privacy-aware regularization, Adversarial training, Knowledge Distillation
  - > Limiting computation layers

Inria

Model parameter updates

Fine-tuning updates

- Less risk of leakage in training
- Slower convergence
- Protects Querier intelectual property
- Ability to use GPUs in certain layers

Update strategy

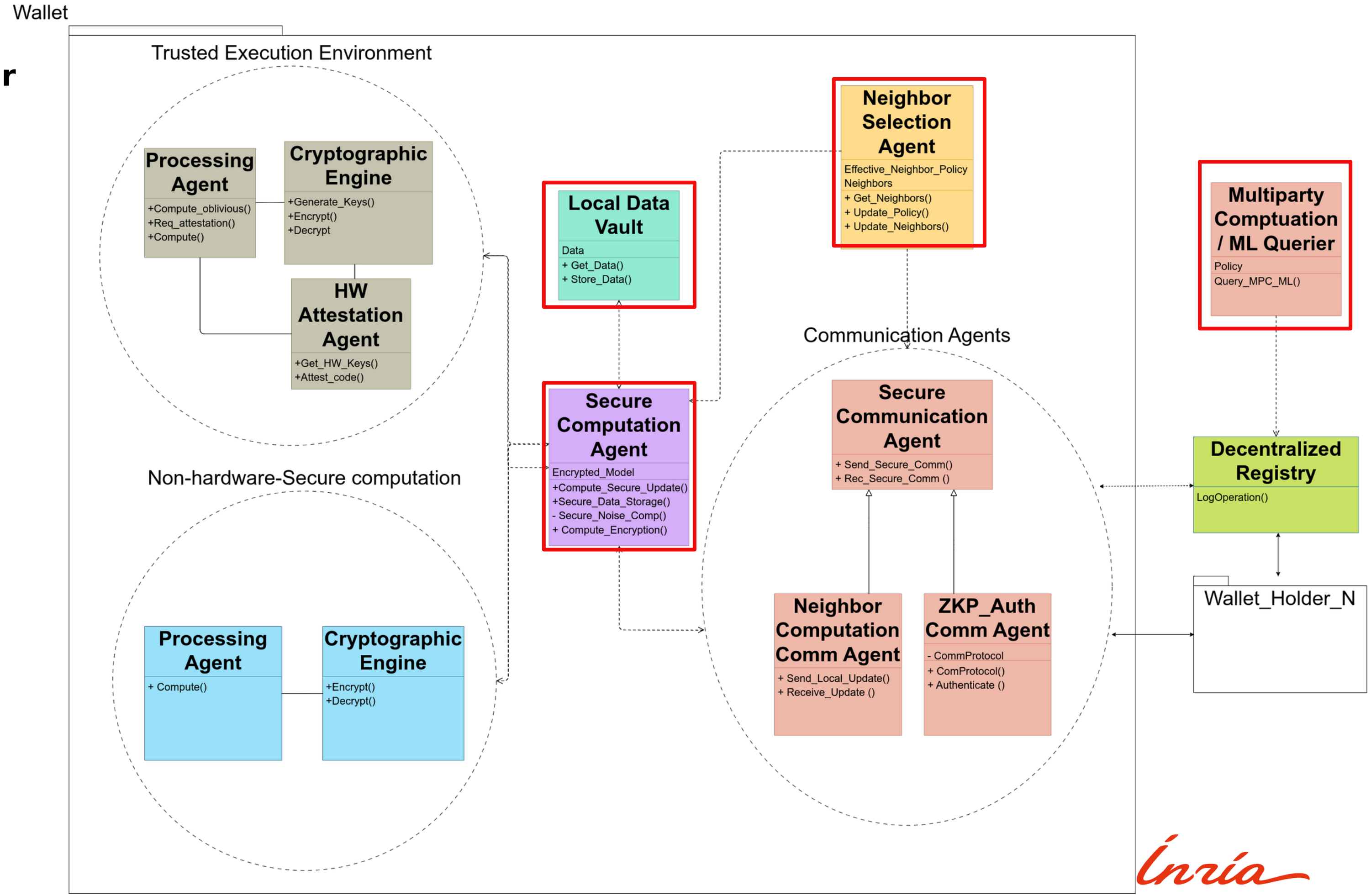
- Affected by neighbor selection and topology
- Random walk-based, gossip-based[1]
- Affects convergence of the model

Hierarchical aggregation

- Group-based strategy desgining group leaders and bottom-up aggregation

[1]Cyffers, E., Bellet, A., & Upadhyay, J. (2024). Differentially Private Decentralized Learning with Random Walks. arXiv preprint arXiv:2402.07471.

- **Model parameter updates**



Security & Privacy considerations (detailed)

Possible approaches

- Communication protocol
  - > Pseudoanonymization, DHT, ZKP, SSI
- Neighbor selection and Topology
  - > Epidemic protocols, Dynamic topologies
- TEE availability and alternatives
  - > Intel SGX & TXT, ARM TZ, AMD, Apple iOS Secure Enclave
  - > Encrypted layer computation, HME
- Aggregation / Learning algorithm
  - > Secure Aggregation
  - > Noise-based algorithms
- Model parameter updates
  - > Fine-tuning, hierarchical aggregation, random walks, convergence
- **Metrics**
  - > Accuracy: Differential privacy
  - > Efficiency: FLOPs, Latency, No. Parameters
  - > Privacy: Differential privacy, attack precision
- Training paradigms and models
  - > Lightweight Neural Networks
  - > Quantization, Privacy-aware regularization, Adversarial training, Knowledge Distillation
  - > Limiting computation layers

Metrics

Performance metrics

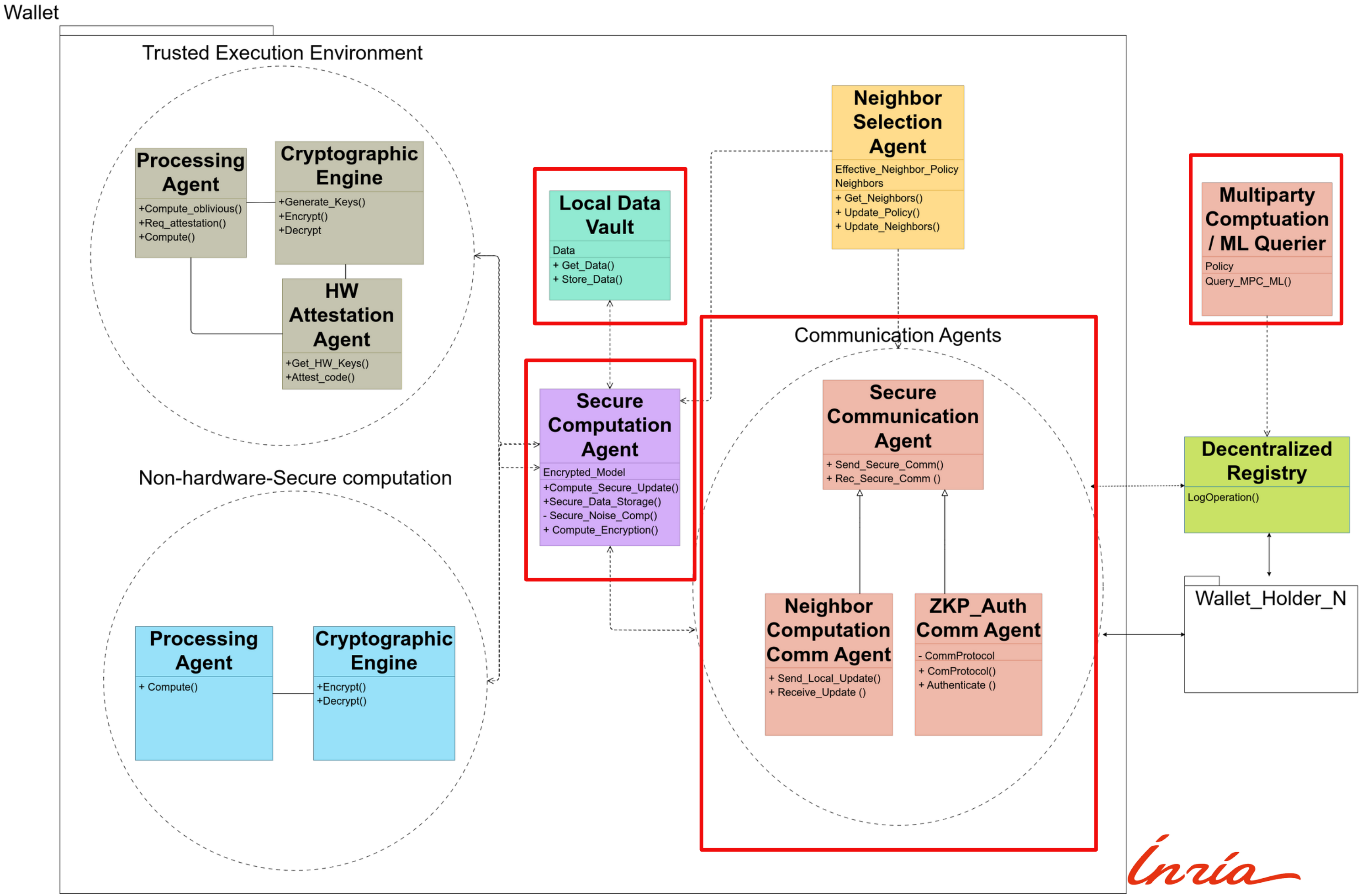
- Introduce privacy to the performance metrics
- In terms of
  - > Accuracy (precision, recall, etc.)
  - > Efficiency (FLOPs, latency, # params.)
  - > Privacy (Differential privacy[1], attack perf.)
  - > Network communication latency
  - > Model convergence (# rounds)

Considerations

- Extensive evaluations needed to achieve best balance for the system and regulation compliance
- Measuring privacy mostly depends on attack performance

[1] I. Mironov, "Rényi Differential Privacy," 2017 IEEE 30th Computer Security Foundations Symposium (CSF), Santa Barbara, CA, USA, 2017, pp. 263-275, doi: 10.1109/CSF.2017.11. keywords: {Privacy;Standards;Tools;Databases;Additives;Computer security;Google;differential privacy;renyi divergence},

- **Metrics**


Wallet
Trusted Execution Environment
Processing Agent
+Compute_oblivious()
+Req_attestation()
+Compute()
Cryptographic Engine
+Generate_Keys()
+Encrypt()
+Decrypt
HW Attestation Agent
+Get_HW_Keys()
+Attest_code()
Non-hardware-Secure computation
Processing Agent
+ Compute()
Cryptographic Engine
+Encrypt()
+Decrypt()
Local Data Vault
Data
+ Get_Data()
+ Store_Data()
Secure Computation Agent
Encrypted_Model
+Compute_Secure_Update()
+Secure_Data_Storage()
- Secure_Noise_Comp()
+ Compute_Encryption()
Neighbor Selection Agent
Effective_Neighbor_Policy
Neighbors
+ Get_Neighbors()
+ Update_Policy()
+ Update_Neighbors()
Communication Agents
Secure Communication Agent
+ Send_Secure_Comm()
+ Rec_Secure_Comm ()
Neighbor Computation Comm Agent
+ Send_Local_Update()
+ Receive_Update ()
ZKP_Auth Comm Agent
- CommProtocol
+ ComProtocol()
+ Authenticate ()
Multiparty Comptuation / ML Querier
Policy
Query_MPC_ML()
Decentralized Registry
LogOperation()
Wallet_Holder_N

Security & Privacy considerations (detailed)

Possible approaches

- Communication protocol
  - Pseudoanonymization, DHT, ZKP, SSI
- Neighbor selection and Topology
  - Epidemic protocols, Dynamic topologies
- TEE availability and alternatives
  - Intel SGX & TXT, ARM TZ, AMD, Apple iOS Secure Enclave
  - Encrypted layer computation, HME
- Aggregation / Learning algorithm
  - Secure Aggregation
  - Noise-based algorithms
- Model parameter updates
  - Fine-tuning, hierarchical aggregation, random walks, convergence
- Metrics
  - Accuracy: Differential privacy
  - Efficiency: FLOPs, Latency, No. Parameters
  - Privacy: Differential privacy, attack precision
- **Training paradigms and models**
  - Lightweight Neural Networks
  - Quantization, Privacy-aware regularization, Adversarial training, Knowledge Distillation
  - Limiting computation layers

Training Paradigms and Models

Efficient models

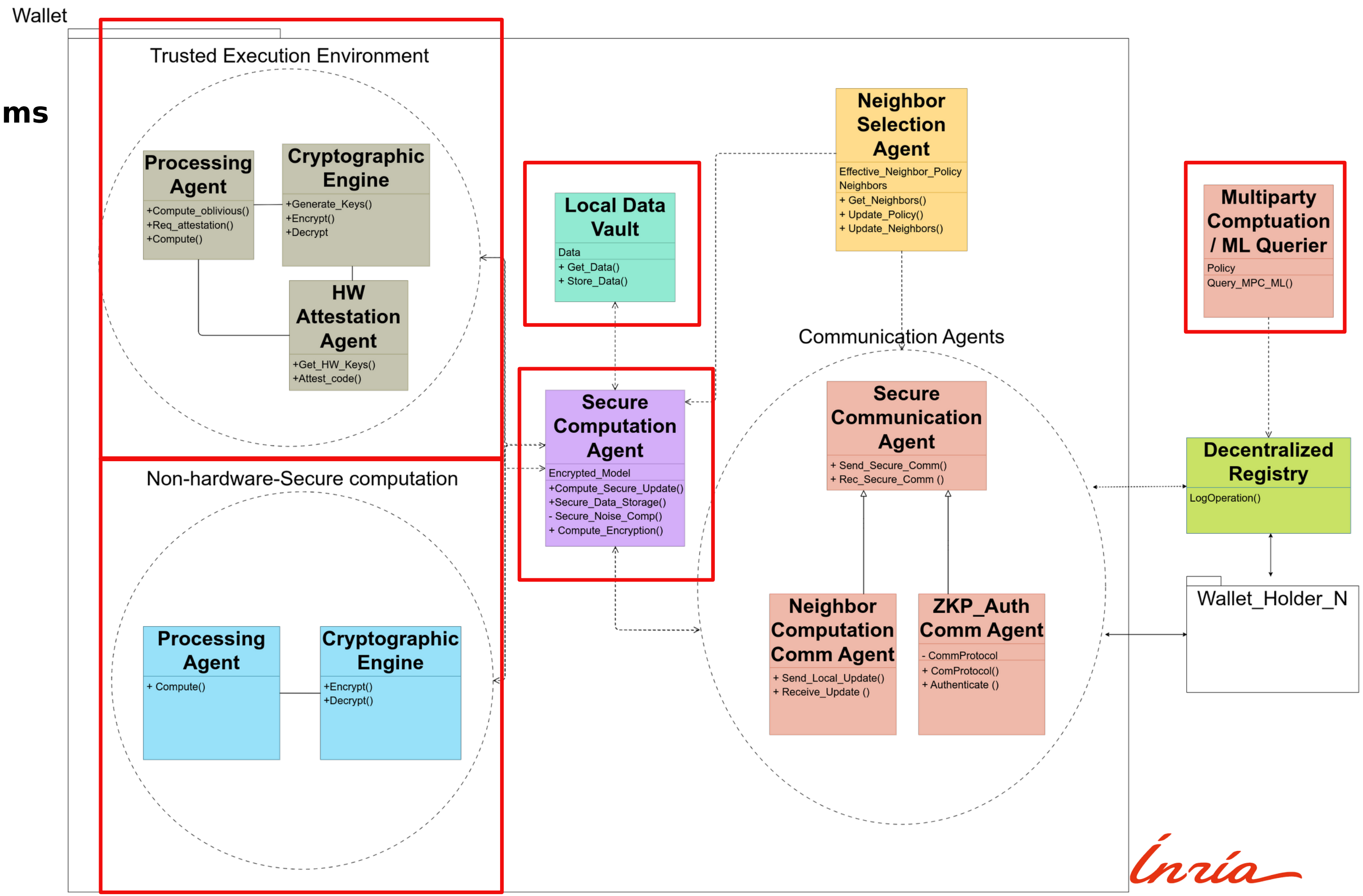
- Less parameters, smaller models, less leakage, less communication overhead
- Approaches
  - Quantization (2 to 8 bits)[1]
  - Lightweight DNNs[3]/Transformers
  - Privacy-aware training regularization[2]
  - Knowledge Distillation[1]

Considerations

- Highly efficient and accurate
- Fine-tuning with fewer layers feasible
- Can be highly biased
  - Adversarial training for bias mitigation
- Trade-offs :
  - Balance accuracy, efficiency and privacy

(1) Y. Choi et al. Data-Free Network Quantization With Adversarial Knowledge Distillation. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, 2020, pp. 710-711
(2) Y.Kaya et al. (2020). On the effectiveness of regularization against membership inference attacks. arXiv preprint arXiv:2006.05336.
(3) A.Howard et al. Searching for MobileNetV3. Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), 2019, pp. 1314-1324

- **Training paradigms and models**



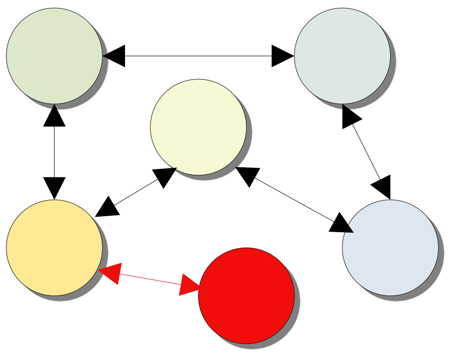
04

Discussion

Challenges

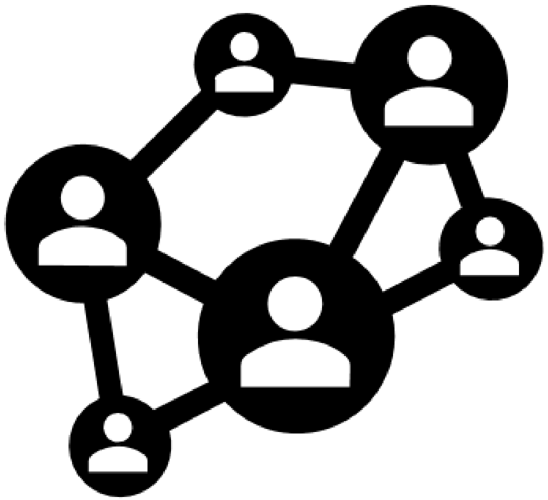
Computation overhead, attacks, and performance

- Machine Learning attacks
  - Inversion/Reconstruction, membership inference, etc.
- Communication overheads over large-scale systems
  - Training paradigms, topology and neighbor selection, etc.
- Convergence and stability
- Balancing accuracy, efficiency, and privacy
- Active malicious neighbors

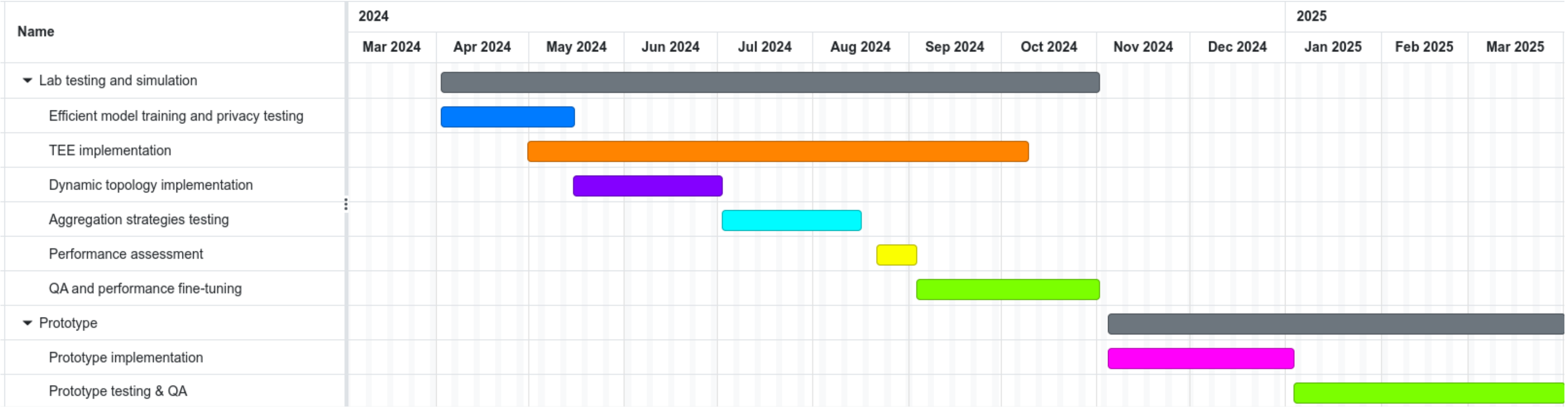


Applications for decentralized MPC/ML

- **Fraud detection**
- **Healthcare Data Analysis**
- **Social networking**
- **Distributed biometrics authentication**
- **Privacy-Preserving Personalized Advertising**



Proposed timeline (1-year)

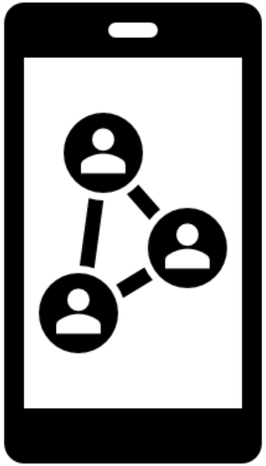


05

Conclusion

Conclusion

- Potential on **compelling** applications
- **Empowers** users' control of their local data
- **Mitigate risks** when computing with local data
- Increase in **complexity**
- Analyze and **prevent data leakage**
- **Performance** considerations
- Compliance with **GDPR** regulations



Thank you !

Follow us at www.inria.fr

luis-santiago.luevano-garcia@inria.fr
davide.frey@inria.fr