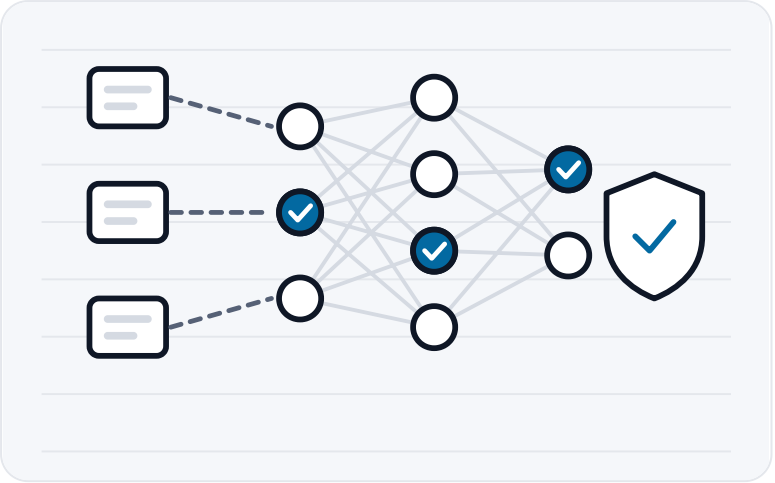
A-SIT Technology

E-Government Innovation Center

22 January 2025 Overview · Transparency · Privacy & Crypto · 21 min read

Authenticity & Verifiability in Federated Learning: Overview & Methods

Karl W. Koch



Authors	Karl W. Koch
Version	2024-12-20
Source	https://technology2.a-sit.at/en/publications/authenticity-verifiability-in-federated-learning-overview-methods/
Note	Machine translation.

Key findings

- The FL process has six trust points: the first three concern authenticity, the last three concern correctness.
- The authenticity of the model and the data source can be secured with digital signatures; group signatures protect the privacy of the signers.
- The correctness of training, weights and aggregation is mostly secured with zero-knowledge proofs (ZKPs); RiseFL reports the most efficient FL epochs.
- So far, no concept addresses all six trust points at once; the efficient combination of signatures and ZKPs is the next challenge.

- Blockchain-based solutions significantly increase the computational effort in the FL process and require additional entities such as blockchain nodes.

With the help of machine learning (ML) – or often referred to as AI – and a large amount of data from a wide variety of users, almost all applications/services can be improved; e.g. analysis of vital activities via smartwatches, recording of driving behavior, or improved cancer analyses via MRI images. On the one hand, such scenarios are becoming increasingly attractive because the corresponding ML technologies are now practically relevant, and on the other hand, one of the biggest challenges is the protection of privacy. Furthermore, users of an ML model may ask whether and to what extent an ML model can be trusted?

In order to train a global ML model based on data from many end-user devices and also to preserve privacy of users' training data, Google launched Federated Learning (FL) in 2016/17. For an FL framework that has practical relevance, in addition to preserving privacy, one must pay attention to trust throughout the entire cycle of an FL epoch. Therefore, to achieve a more reliable and trustworthy updated ML model, authenticity and correctness guarantees for training data and ML parameters/weights must be added in the FL process. Within the FL process flow, there are six important trust points (TPs). To achieve both – privacy and trustworthiness – FL must be upgraded again.

Therefore, this report shows methods for Trusted Federated Learning for the individual trust points and then discusses the entire FL process flow. Finally, a Conclusion & Further Work briefly summarizes gained insights and points out potentially-interesting further directions.

There exist already promising solutions for the individual TPs and, in some cases, concrete instantiations. The currently most common methods and concrete instantiations are based on the privacy-preserving cryptographic building block zero-knowledge proof (ZKP). The individual solution concepts offer different trade-offs; e.g., not all concepts address the privacy aspect when aggregating the returned ML parameters.

An overall concept of Trusted Federated Learning while preserving privacy, within the entire process flow, is the next scientific and engineering challenge. E.g., the efficient combination of digital signatures and ZKPs. And especially because most relevant work of "zkFL" – Trusted FL via ZKPs – has only been published between 2022 and 2024, it remains exciting to see how this new field will develop, and important for, e.g., any practical adopters to have their finger on the pulse of the time and the state of the art.

1. Introduction

With the help of machine learning (ML) - or often referred to as AI - and a large amount of data from a wide variety of users, almost all applications/services can be improved; e.g. analysis of vital activities via smartwatches or "wearables" in general [1], recording of driving behaviour [2] or improved cancer analyses via MRI images [3] . On the one hand, such scenarios are becoming increasingly attractive because the

corresponding ML technologies are now practically relevant, and on the other hand, preserving privacy is one of the biggest challenges. Furthermore, users of an ML model may ask whether and to what extent an ML model can be trusted?

Why Federated Learning (FL)? In order to train a global ML model based on data from many end-user devices while also preserving the privacy of users, Google launched FL in 2016/17 (McMahan et al. 2017). In FL, each user trains the corresponding ML model locally and "only" sends the updated ML model parameters - often called weights - back to the server. This process is called an FL epoch. Training such an ML model optimally may require several such FL epochs. In recent years, however, there have been practically relevant attacks on "standard FL" in which, e.g., the training data were reconstructed from the returned ML model weights.

To achieve a good trade-off between ML model accuracy and preservation of privacy, federated learning ("Federated Learning" / FL) with secure aggregation ("Secure Aggregation" / SecAgg) based on, e.g., secure multi-party computation ("Secure Multi-Party Computation" / MPC) is a good fit. See, e.g., (Bonawitz et al. 2017), (Bonawitz et al. 2022) or (Koch 2024) for an initial overview of privacy-preserving FL.

1.1. Motivation & Trust Points of Trusted Federated Learning

Trust in FL-trained ML models? Although the privacy of participants' training data is now preserved, the user of such an ML model may ask whether the training data used and the updated ML parameters can be trusted. In the context of virtual keyboards, for example, an app provider - such as Google's Gboard [4,5] - is likely to have a strong interest in the learned ML model delivering adequate results for predicting the next word or sentence, thereby increasing the likelihood that the app continues to be used. Or in the context of health diagnoses, where, e.g., a hospital or doctor has a strong interest in the learned ML model delivering adequate diagnoses from the MRI images shown. In such scenarios, the doctor usually gives the final diagnosis, but ML models are increasingly used as support [6,7,8] . A well-trained ML model thus has the potential to detect a possible cancer at an early stage and to immediately forward this to the respective doctor. E.g., Figure 1 shows a coarse-grained schematic of a cross-hospital FL scenario for constructing a more efficient and effective global ML model for analysing MRI images.

[4] play.google.com/store/apps/details?id=com.google.android.inputmethod.latin

[5] aclanthology.org/2023.acl-industry.60.pdf

[6] en.wikipedia.org/wiki/Computer-aided_diagnosis

[7] mdpi.com/2673-9909/4/3/59

[8] arxiv.org/abs/2311.08908

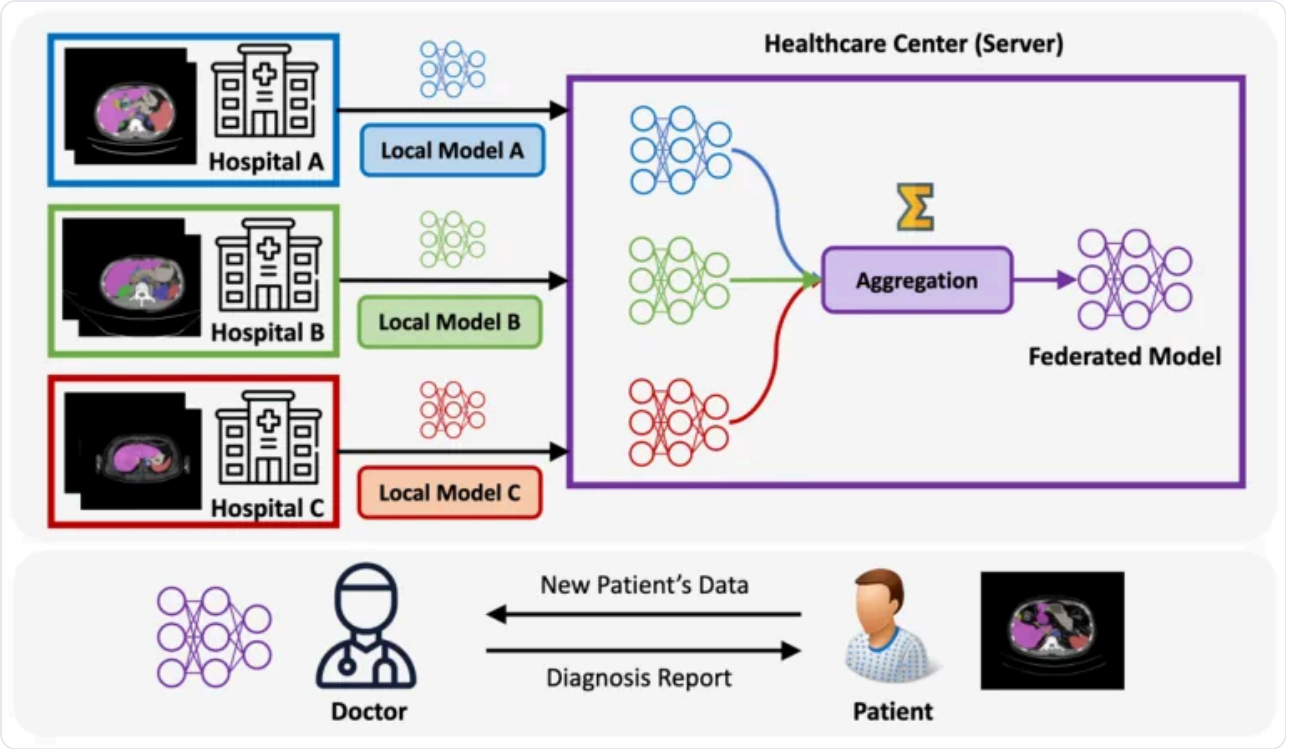


Figure 1: - Coarse-grained schematic overview of cross-hospital FL to construct a more efficient and effective global ML model. The 3 hospitals (A, B, C) could, e.g., be distributed internationally and each hold rare and significant datasets, which cannot be shared per se but can feed into the training of an ML model. This leads to improved diagnostic support for doctors and ultimately to better health for patients, as potential risks may be detected earlier.

6 Trust Points. It is therefore becoming ever more important that no "fake data" is used in FL or ML, but rather adequate training data, ML training functions and ML model parameters. With regard to trust in FL-trained ML models, 6 leading questions or trust points arise in the FL process. Three questions or trust points each concern the aspects of authenticity and correctness :

(0) (Authenticity-Server) Does the ML model originate from the server? And was it also not modified during transfer?

(1) (Authenticity-Client) Does the training data originate from a trustworthy source? And thus no "fake data" was used. E.g. certificates from a university, health data from a hospital or fitness activities from a verified or trusted smartwatch.

(2) (Authenticity-Client) Was authentic data used for training an ML model?

(3) (Correctness-Client) Were correct ML training functions executed?

(4) (Correctness-Client) Were the correct resulting ML parameters or weights returned to the server? Or used in the SecAgg process?

(5) (Correctness-Server) Was the correct ML model aggregation - with the correct resulting ML parameters or weights - carried out?

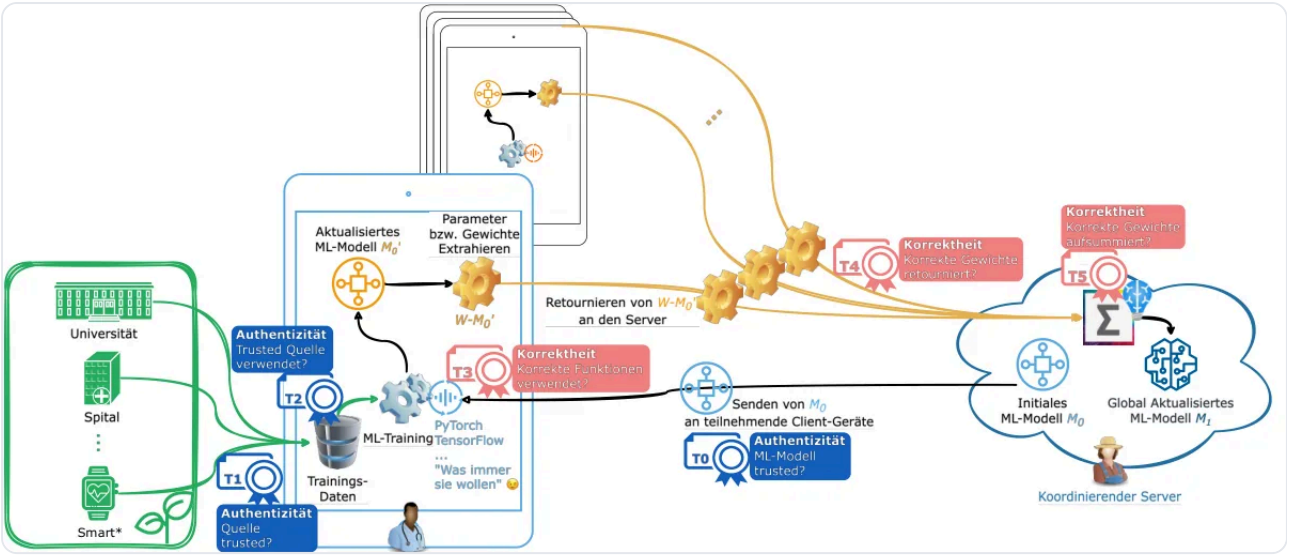


Figure 2: - Process flow of a federated learning epoch including the four mentioned trust points. Two trust points each address the aspect of authenticity and of correctness. The authenticity aspect denotes - in this scenario - whether the training data (T1) originate from a trustworthy or trusted source and (T2) were actually used. The correctness aspect denotes - in this scenario - whether (T3) the correct functions were executed during ML training and (T4) the correct resulting ML parameters or weights are returned to the server.

In order to achieve a more reliable or trustworthy updated ML model, authenticity and correctness guarantees for training data and ML weights must therefore be added to the FL process. To achieve both - protection of privacy and trustworthiness (“Trust”) - FL must be upgraded once more. For this purpose, privacy-preserving cryptographic building blocks can be used, such as zero-knowledge proofs (“Zero-Knowledge Proofs” / ZKPs) for correctness or group signatures for authenticity. The FL process flow can be examined more closely and relevant areas for authenticity and correctness highlighted. Figure 2 shows the FL process flow with the six mentioned trust points, for the “trust upgrade”.

Drawn with draw.io. The smartwatch icon is from Vecteezy.com, and the leaf icon from

Goals & Outlook. Although, e.g., the two mentioned cryptographic building blocks - considered in isolation - address the respective trust points relatively simply and straightforwardly, it is not straightforward how these building blocks or methods can be used for all six trust points at the same time - from the training data to the updated ML model. Or at least for more than one trust point, as a first step. Therefore, the following section (2) shows methods for Trusted Federated Learning for the individual trust points, and then discusses the aspect of the entire FL process flow - the (2.7) Σ Summa Summarum (“Putting it all Together”). Finally, a (3) Conclusion & Further Work briefly summarises the insights gained and points out potentially interesting further directions.

2. Methods for Trusted Federated Learning

This section shows methods for upgrading “standard FL” to trustworthy or “Trusted FL”. In an FL framework that has practical relevance, one must pay attention to trust throughout the entire cycle of an FL epoch. As described in the introduction, there are six important trust points (TPs) in Trusted FL. For each TP, a method - including a reference to at least one concrete protocol - is described for fulfilling the respective TP. Finally, the converging aspect of all TPs in the entire FL process flow is addressed.

2.1. (TP0-Server) Authenticity of the ML model used

Figure 3 shows the section of trust point 0 in the process flow of an FL epoch, including the trust upgrade with signatures (see Figure 13 for the overall overview of all trust upgrades).

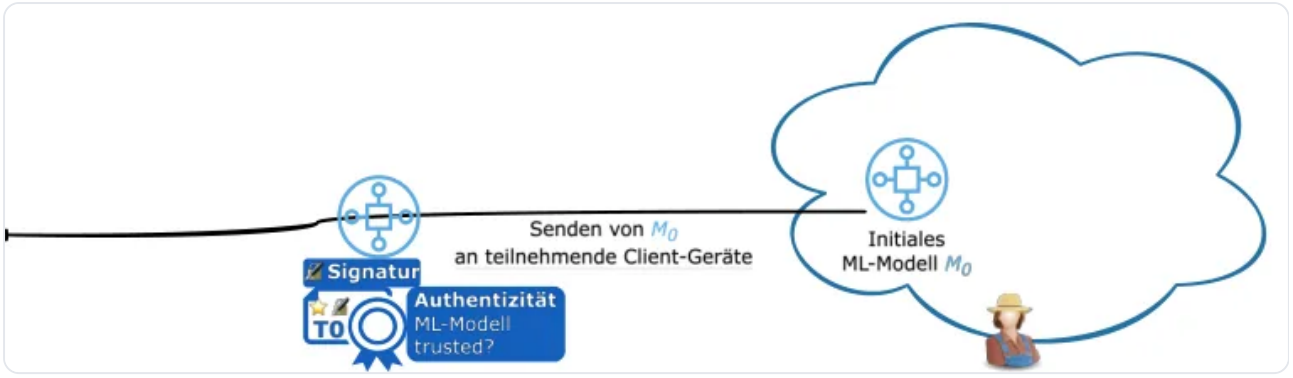


Figure 3: -(Authenticity) Focus on trust point 0 in the process flow of an FL epoch: does the training data originate from a trustworthy or trusted source? Including the trust upgrade with signatures .

At the beginning of the FL process flow, clients must verify whether the ML model used or sent to them can be trusted. As with documents or data in general, a digital signature is suitable here. That is, the server signs the ML model sent with its private signing key sk. Clients can then verify the signature with the associated public signing key pk. In addition to authenticity, the integrity of the ML model can also be checked here; that is, that the model was transferred unchanged.

Depending on the trust model in the respective FL scenario, a server could - in addition to the signature - upload the ML model used (semi-)publicly to an archive to which clients have access. Clients could then additionally check whether the ML model sent to them matches the version in the archive. If archive access is secured with an additional trust mechanism - such as a further password or passkey - the archive constitutes an extended security measure.

2.2. (TP1-Client) Authenticity of the data source

Figure 4 shows the section of trust point 1 in the process flow of an FL epoch, including the trust upgrade (see Figure 13 for the overall overview of all trust upgrades).

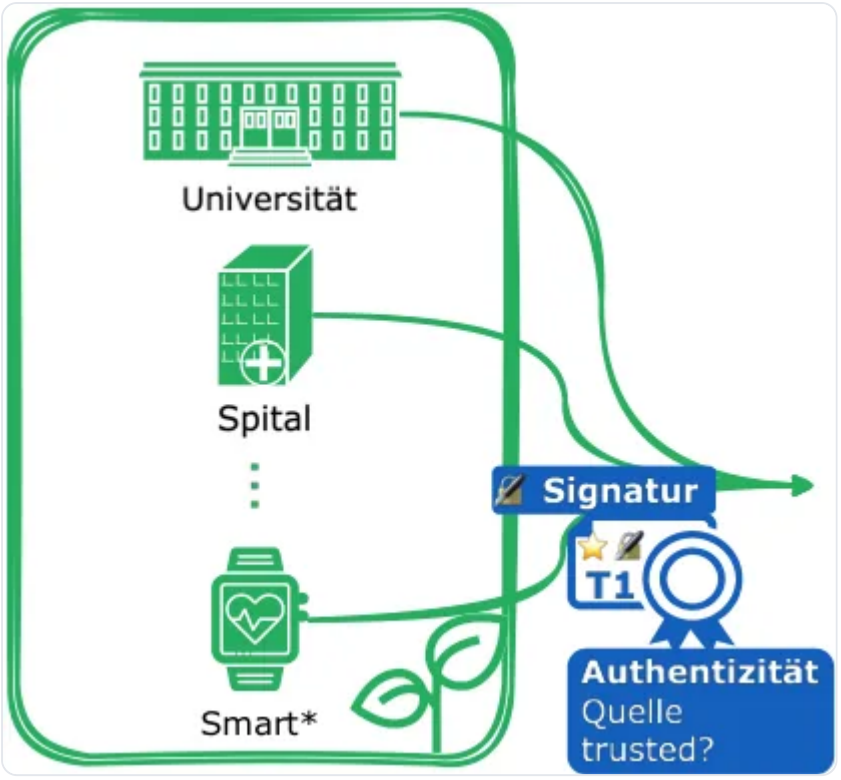


Figure 4: -(Authenticity) Focus on trust point 1 in the process flow of an FL epoch: does the training data originate from a trustworthy or trusted source? Including the trust upgrade with signatures .

Drawn with draw.io. The smartwatch icon is from Vecteezy.com, and the leaf icon from

One of the most common methods for guaranteeing the authenticity of the data source is digital signatures . With these, e.g., a hospital can sign a dataset - with the private signing key sk - and make it available to the respective client, which then uses the signed dataset for training the ML model. The signed dataset can then be verified by the client, the verifying entity (“verifier”), etc. - with the help of the public signing key pk - as to whether it was signed by a trusted source (e.g. the hospital in this case). A trusted source can upload its pk, e.g., to an accessible archive - such as a GitHub repository.

There may be scenarios in which it is desired, in case of doubt, to look up who exactly (doctor, department, etc.) signed. To preserve the privacy of the respective signing persons, departments, etc., the privacy-preserving cryptographic building block of digital group signatures is suitable. (Chaum and van Heyst 1991) were among the first to devise group signatures. Group signatures have the advantage that several ski can be derived within the respective organisation, and still only one pk is needed to verify the signatures. In addition, an authorised entity can be set up that can, in case of doubt, determine the identity behind the respective signature (like a judge). A further advantage of group signatures is that greater key agility is achieved. For example, a single ski can be revoked because, e.g., a doctor changes hospital, so that the other keys remain untouched. Otherwise, a new digital key pair would have to be generated and distributed anew to all doctors, departments, etc.

E.g. (Koch et al. 2022) used digital group signatures to verify the authenticity of the data source in a similar scenario (" KRAKEN"). In the " KRAKEN" scenario, the respective signed datasets are split among individual dedicated MPC computation nodes and made available for a later privacy-preserving analysis. The MPC computation nodes only carry out the actual computations if the signature has been verified as valid.

2.3. (TP2-Client) Authenticity of the data during ML training

Figure 5 shows the section of trust point 2 in the process flow of an FL epoch, including the trust upgrade (see Figure 13 for the overall overview of all trust upgrades).

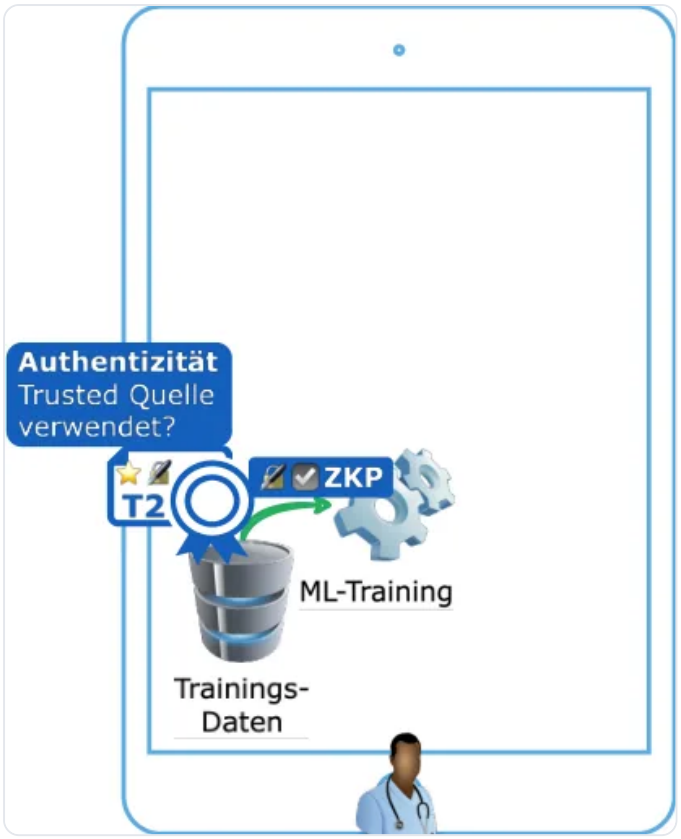


Figure 5: -(Authenticity) Focus on trust point 2 in the process flow of an FL epoch: was training data from a trustworthy or trusted source used? Including the trust upgrade with signatures and ZKPs .

To guarantee that authentic data are actually used when learning the ML model, the signature of the data - assuming that digital signatures are built upon - must be linked to the dataset used.

E.g. (Koch et al. 2022) use the privacy-preserving cryptographic building block " Pederson Commitment " to “attach” the signature of data to the secret-shared dataset. With the Pederson Commitment and the public pk, the dedicated MPC computation nodes can verify the signature on the “secret-shared” data.

In the area of FL, e.g., (Zhu et al. 2024) likewise build on such a hybrid scheme. They use Pederson Commitments and verifiable secret sharing (“Verifiable Secret Sharing” / VSS) based on Shamir and call their concept RiseFL. RiseFL guarantees that the resulting ML parameters or weights lie within a valid range of values. In addition, during the final aggregation the server gains no insight into the individual returned weights, but only the sum of all weights. Figure 6 shows the different phases of RiseFL.

Although RiseFL gives no guarantee that the training data used originate from a trusted source, RiseFL does offer a potential breeding ground for extending an FL framework in this direction. The verifiability of training data from a trusted source, given the returned weights or only the final sum of the weights, is a complex challenge for further research and construction work.

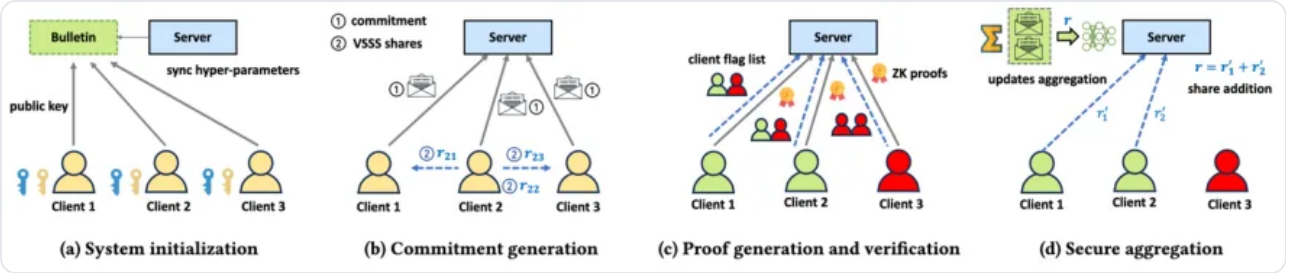


Figure 6: - The four phases of the FL concept RiseFL. (a) First, relevant meta-parameters are exchanged (e.g. number of participants and how many may be corrupt). (b) Clients commit to the returned ML parameters or weights. (c) The other clients verify the correctness of the commitment, and also whether the weights lie within a permissible range. (d) The server computes the final aggregation of the weights.

2.4. (TP3-Client) Correctness of the computing ML functions

Figure 7 shows the section of trust point 3 in the process flow of an FL epoch, including the trust upgrade (see Figure 13 for the overall overview of all trust upgrades).

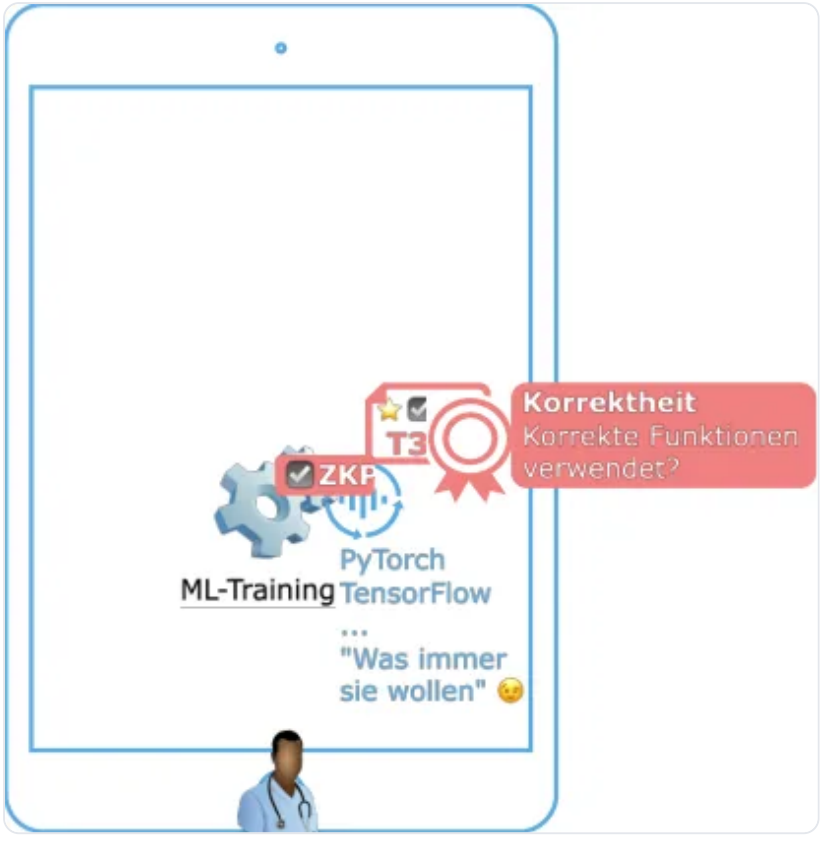


Figure 7: - (Correctness) Focus on trust point 3 in the process flow of an FL epoch: were correct ML training functions executed? Including the trust upgrade with ZKPs .

To guarantee that the correct ML functions were executed during ML training, the privacy-preserving cryptographic building block of zero-knowledge proofs (“Zero-Knowledge Proofs” / ZKPs) is suitable. E.g. (Xing et al. 2023) divide the challenge into several ZKPs for the different computation steps during ML training, and call their approach ZKP-FL. Or PZKP-FL for the more practical aspect. Figure 8 shows a coarse-grained schematic overview of ZKP-FL.

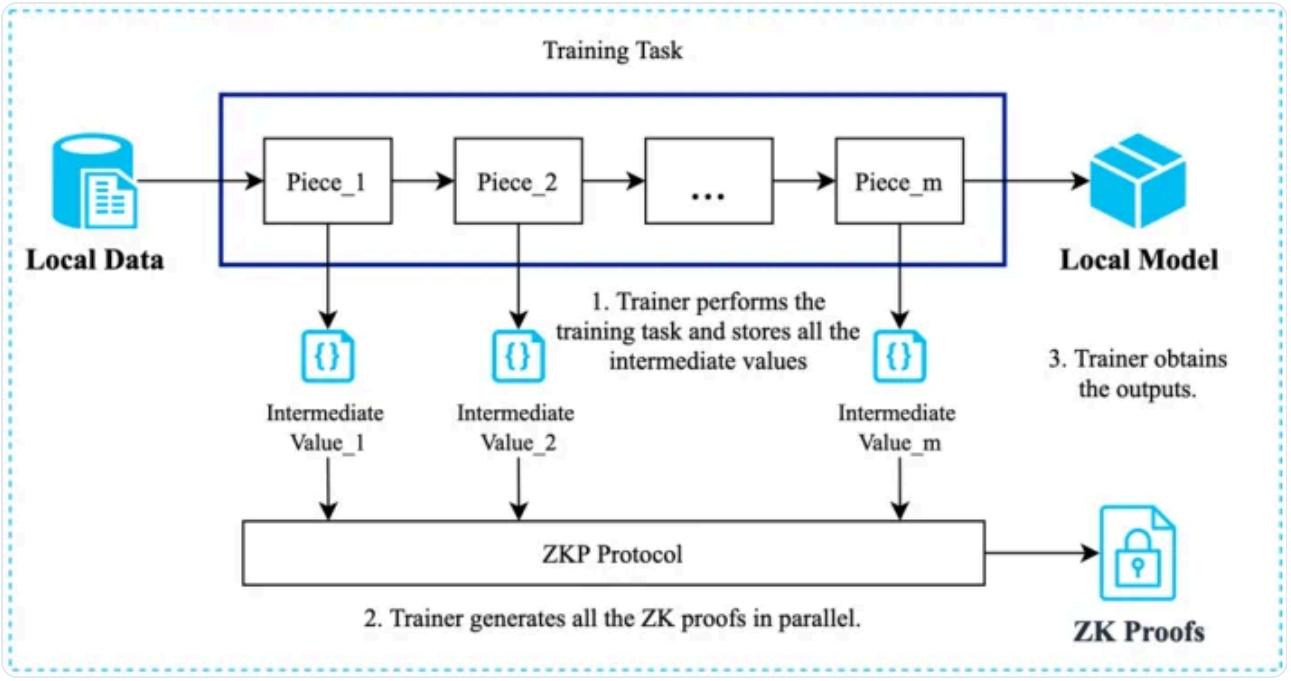


Figure 8: - Coarse-grained schematic overview of the verified ML training phase of ZKP-FL. Individual ZKPs are generated for the different computation steps during the training phase.

Source: (Xing et al. 2023).

2.5. (TP4-Client) Correctness of the returned weights

Figure 9 shows the section of trust point 4 in the process flow of an FL epoch, including the trust upgrade (see Figure 13 for the overall overview of all trust upgrades).

Figure 9 -(Correctness) Focus on trust point 4 in the process flow of an FL epoch: were the correct resulting ML parameters returned to the server? Or used in the SecAgg process. Including the trust upgrade with ZKPs .

As shown in 2.3 (TP2-Client) Authenticity of the data during ML training, RiseFL by (Zhu et al. 2024) gives guarantees that the returned ML parameters or weights lie within a valid range. RiseFL builds on Pederson Commitments and VSS. Furthermore, e.g., (Lycklama et al. 2023) (RoFL) via additive homomorphic commitments, (Chowdhury et al. 2022) (EIFFeL) via VSS and secret-shared non-interactive proofs, or (Bell et al. 2023) (ACORN) via Bulletproofs also devise - as an extension of the traditional privacy-preserving FL SecAgg or SecAgg+ - an FL system with guarantees on the returned weights. Of the concepts mentioned, RiseFL reports the most efficient FL epochs. Figure 10 gives an overview of the runtimes for different lengths of ML parameters or weights (d).

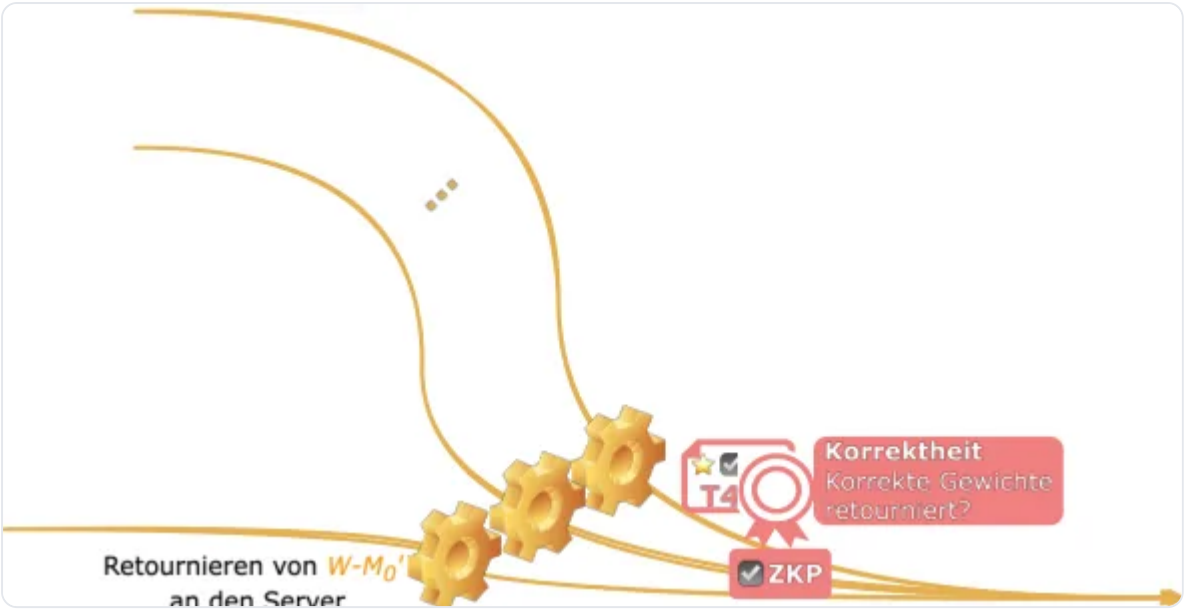


Figure 10: – Overview of the runtimes for different lengths of ML parameters or weights (d). Computation runtimes on the client and server are shown in seconds, as well as communication costs per client in MB.

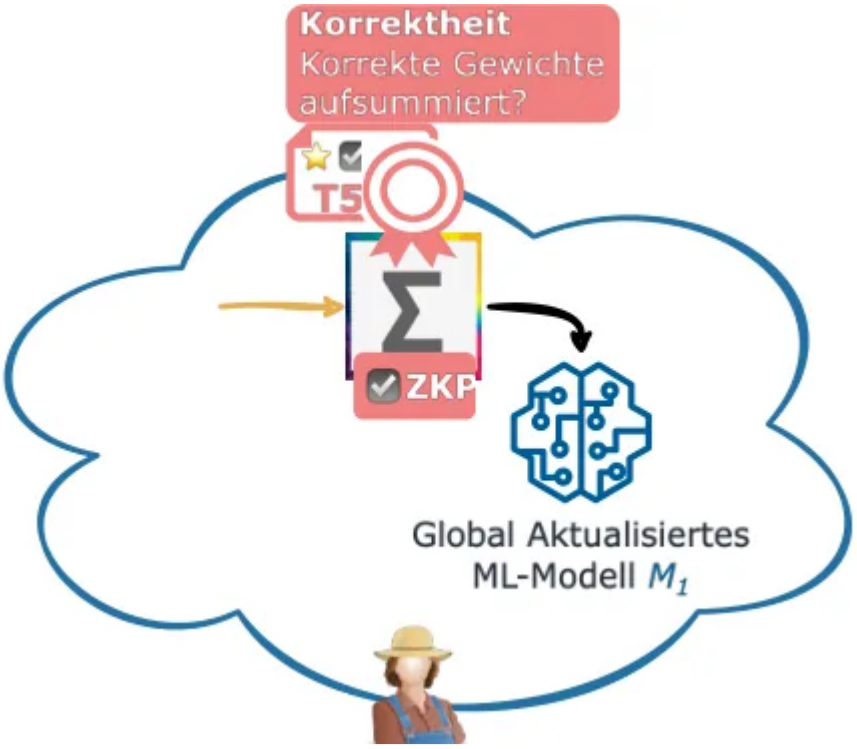
Linking to the direct result of the ML training is a complex challenge and the next step for this trust point. Trust point 3 (of the computing ML functions) represents a potential starting point here. That is, a link to the ZKPs that the correct functions were executed.

2.6. (TP5-Server) Correctness of the aggregating weights

Figure 11 shows the section of trust point 5 in the process flow of an FL epoch, including the trust upgrade (see Figure 13 for the overall overview of all trust upgrades).

#Param.	Approach	Client Computation (seconds)				Server Computation (seconds)				Comm. Cost per Client (MB)
		commit.	proof gen.	proof ver.	total	prep.	proof ver.	agg.	total	
d = 1K	EIFFeL	0.865	3.63	11.7	16.2	-	-	0.182	**0.182**	125
	RoFL	0.051	4.43	-	4.5	-	91.2	0.040	91.3	0.37
	ACORN	0.076	2.49	-	2.6	-	58.9	0.040	58.9	**0.004**
	RiseFL (ours)	0.054	1.48	0.08	**1.6**	1.17	75.6	0.071	76.8	0.44
d = 10K	EIFFeL	8.38	36.8	115	161	-	-	1.81	**1.81**	1250
	RoFL	0.51	46.4	-	46.9	-	860	0.41	860	3.66
	ACORN	0.75	24.5	-	25.3	-	522	0.41	523	**0.03**
	RiseFL (ours)	0.49	1.8	0.08	**2.3**	8.61	82.5	0.71	91.8	0.71
d = 100K	EIFFeL	84.7	382	1070	1536	-	-	18.8	**18.8**	12500
	RoFL	5.1	496	-	502	-	8559	4.1	8563	36.6
	ACORN	7.6	253	-	261	-	5087	4.1	5091	**0.3**
	RiseFL (ours)	4.8	4.5	0.08	**9.3**	73.3	139	7.2	219	3.5
d = 1M	EIFFeL	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM
	RoFL	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM
	ACORN	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM
	RiseFL (ours)	48.0	31.2	0.08	**79.3**	653	612	72.1	**1338**	**30.9**

Figure 11: -(Correctness) Focus on trust point 5 in the process flow of an FL epoch: were the correct resulting ML parameters or weights used by the server or in the SecAgg process. Including the trust upgrade with ZKPs .



To verify the correctness of the aggregation of the ML weights on the server side, the server must be able to prove that all the ML weights returned by the clients were actually used. Here too, the privacy-preserving cryptographic building block of the zero-knowledge proof ("Zero-Knowledge Proof" / ZKP) is suitable. With the help of such a ZKP, the server can create the corresponding proof - based on cryptography or mathematics.

E.g. (Wang et al. 2024) created such a concept. Wang et al. devise two variants with regard to the verification entities: (i) the clients and (ii) blockchain nodes, which, upon successful verification, store the hash value of the encrypted resulting ML model on a blockchain. Figure 12 shows a coarse-grained schematic overview of the first variant by Wang et al., in which the clients check the ZKPs.

Figure 12 - Coarse-grained schematic overview of the server verification approach for ML weight aggregation by (Wang et al. 2024). Wang et al. also devise a variant in which blockchain nodes - instead of the individual clients - verify the ZKP proof and then store a hash in a blockchain, which can be verified by the clients.

Source: (Wang et al. 2024).

2.7. Σ Summa Summarum ("Putting it all Together")

For each trust point there are already promising individual solution methods and, in part, concrete instantiations. However, there is - to date - no concept that addresses all trust points simultaneously in an entire FL process flow. This overall concept of holistic Trusted Federated Learning represents the next scientific and engineering challenge. Figure 13 shows the entire FL process flow including trust upgrades for each trust point.

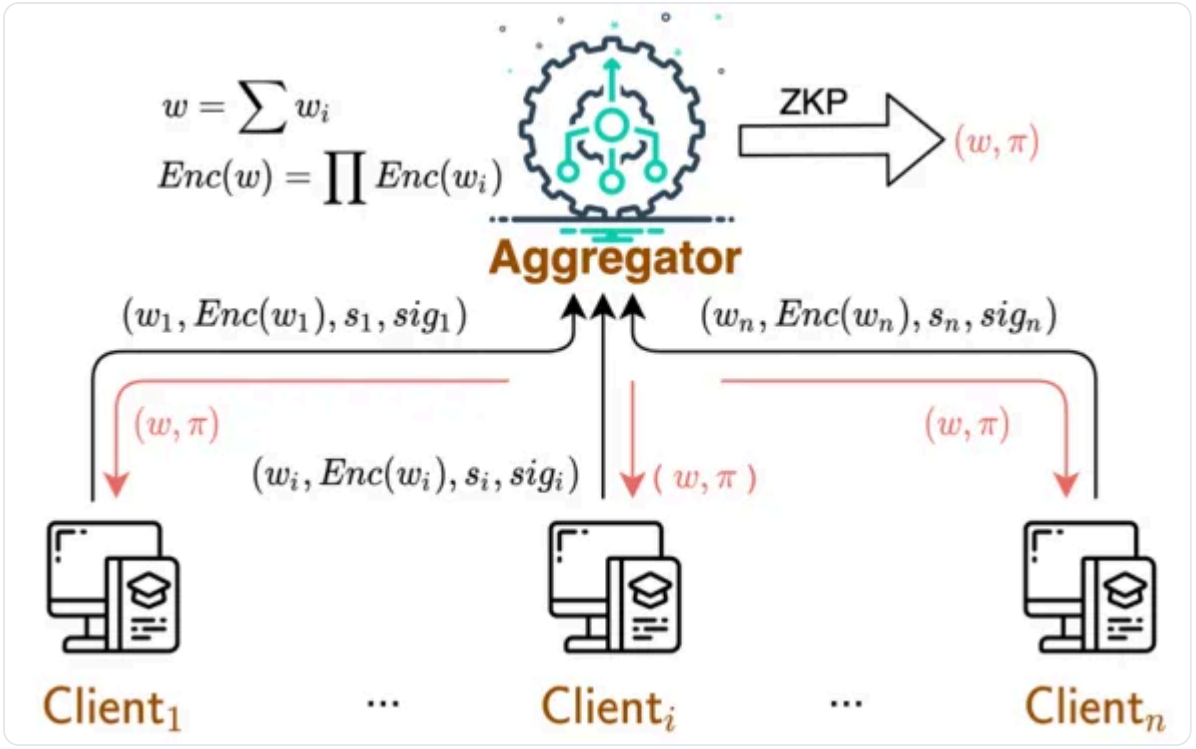


Figure 13: - Process flow of a federated learning epoch including the six mentioned trust points (0— 5) and the holistic trust upgrade for each trust point. Two technologies are primarily used here. For the authenticity aspect digital signatures . For the correctness aspect ZKPs .

Drawn with draw.io. The smartwatch icon is from Vecteezy.com, and the leaf icon from

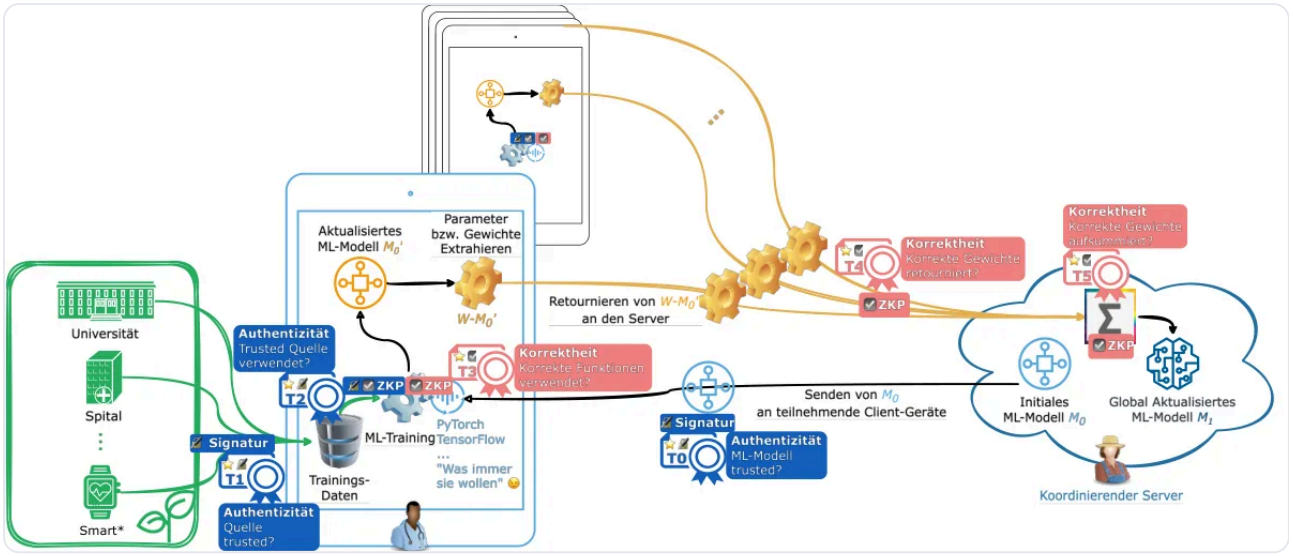
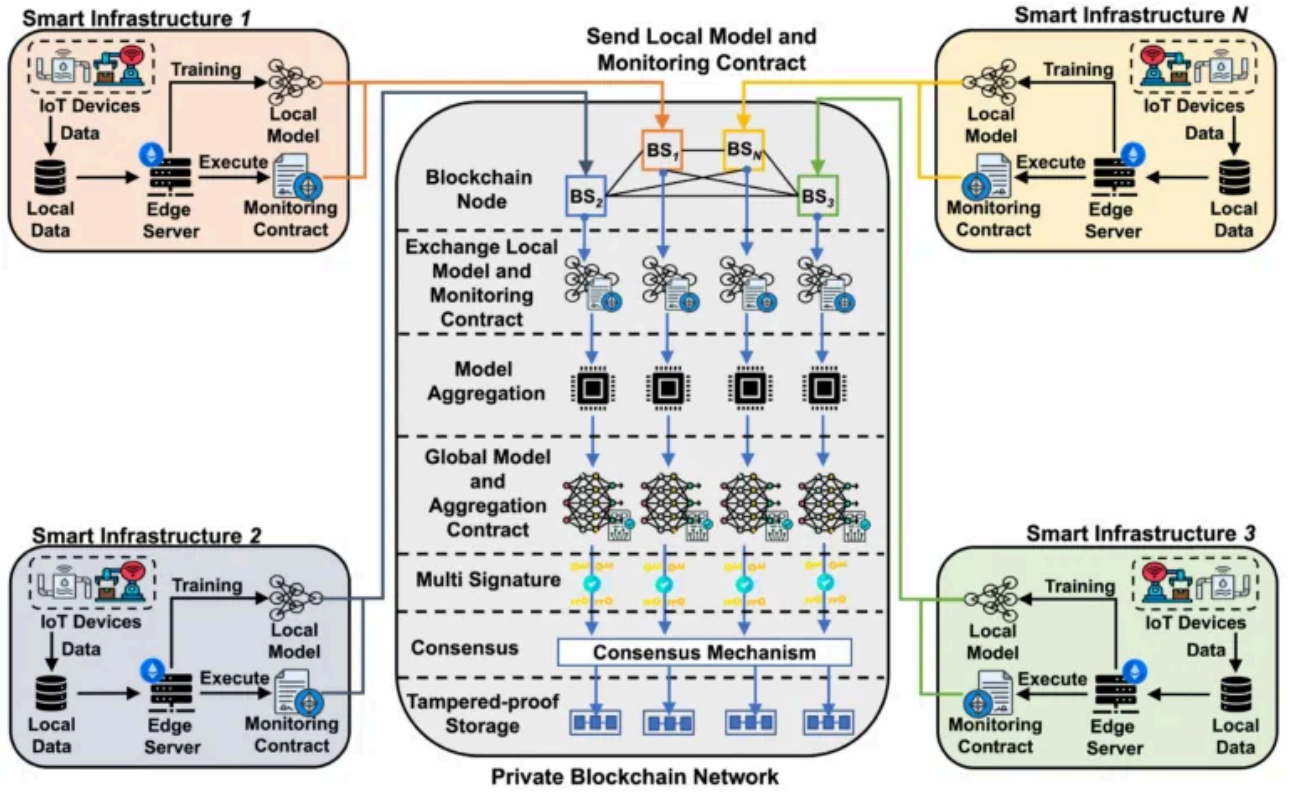


Figure 14: – The FL framework by (Kalapaaking et al. 2024) for realising correctness of ML training and ML model aggregation with the help of “smart contracts” and a blockchain architecture.



Source: (Kalapaaking et al. 2024).

3. Conclusion & Further Work

Federated Learning (FL) is by now a practically relevant concept for training an ML model locally on the respective devices. In an FL framework that has practical relevance, one must pay attention to trust throughout the entire cycle of an FL epoch. As described in the introduction, there are six important trust points (TPs) in Trusted FL. The first three TPs primarily address the aspect of authenticity (“Authenticity”), and the last three TPs primarily address the aspect of correctness (“Verifiability”). For each trust point there are already promising individual solution methods and, in part, concrete instantiations. The currently most common methods and concrete instantiations are based on the privacy-preserving cryptographic building block zero-knowledge proof (“Zero-Knowledge Proof” / ZKP). The various individual solution concepts offer different trade-offs; e.g., not all concepts address the aspect of privacy when aggregating the returned ML parameters or weights (see, e.g., (Koch 2024) in this regard).

To date, there is no concept that addresses all trust points simultaneously in an entire FL process flow. This overall concept of holistic Trusted Federated Learning represents the next scientific and engineering challenge. Especially the efficient combination of digital signatures and ZKPs. Figure 13 shows the entire FL process flow including trust upgrades for each trust point.

Other methods for Trusted FL. The area of “zkFL”, which realises Trusted FL via ZKPs, is relatively new. Most relevant works have only been published between 2022 and 2024. It remains exciting to see how this new field develops, and important for, e.g., any practical adopters to keep their finger on the pulse of the time and the state of the art.

Besides authenticity and correctness guarantees via ZKPs, there are also solutions for the various trust points via, e.g., blockchain. With the blockchain, data are stored in a decentralised and irrevocable way on a "chain of data". Furthermore, the blockchain also makes it possible for computations to be carried out on the "chain of data" - "on-chain" via "smart contracts".

E.g. (Dong et al. 2024) use blockchain-based "smart contracts" for a secure and verified aggregation of the ML weights. In such scenarios, however, the computational effort in the FL process flow also increases significantly. Furthermore, the blockchain requires additional entities - such as the blockchain nodes. In general, solution approaches via blockchain have their own trade-offs. Furthermore, e.g., (Kalapaaking et al. 2024) use "smart contracts" and a blockchain architecture to realise the correctness of (i) the ML training and (ii) the ML model aggregation in a decentralised FL scenario.

Figure 14 shows the (blockchain) architecture of the FL framework by Kalapaaking et al. in an FL scenario for smart infrastructures (such as IoT devices in "smart factories").

References

Bell, James, Adrià Gascón, Tancrède Lepoint, Baiyu Li, Sarah Meiklejohn, Mariana Raykova, and Cathie Yun. 2023. "{ACORN}: Input Validation for Secure Aggregation." Pp. 4805–22 in, *Usenix'23* .

Bonawitz, Kallista, Peter Kairouz, Brendan Mcmahan, and Daniel Ramage. 2022. "Federated Learning and Privacy." *Communications of the ACM* 65(4):90–97. doi: 10.1145/3500240.

Bonawitz, Keith, Vladimir Ivanov, Ben Kreuter, Antonio Marcedone, H. Brendan McMahan, Sarvar Patel, Daniel Ramage, Aaron Segal, and Karn Seth. 2017. "Practical Secure Aggregation for Privacy-Preserving Machine Learning." Pp. 1175–91 in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, *CCS '17* . New York, NY, USA: Association for Computing Machinery.

Chaum, David, and Eugène van Heyst. 1991. "Group Signatures." Pp. 257–65 in *Advances in Cryptology — EUROCRYPT '91*, *EC'91*, edited by D. W. Davies. Berlin, Heidelberg: Springer.

Chowdhury, Amrita Roy, Chuan Guo, Somesh Jha, and Laurens van der Maaten. 2022. "EIFFeL: Ensuring Integrity for Federated Learning."

Dong, Nanqing, Zhipeng Wang, Jiahao Sun, Michael Kampffmeyer, William Knottenbelt, and Eric Xing. 2024. "Defending Against Poisoning Attacks in Federated Learning With Blockchain." *IEEE Transactions on Artificial Intelligence* 5(7):3743–56. doi: 10.1109/TAI.2024.3376651.

Kalapaaking, Aditya Pribadi, Ibrahim Khalil, Xun Yi, Kwok-Yan Lam, Guang-Bin Huang, and Ning Wang. 2024. "Auditable and Verifiable Federated Learning Based on Blockchain-Enabled Decentralization." *IEEE Transactions on Neural Networks and Learning Systems* 1–14. doi: 10.1109/TNNLS.2024.3407670.

Koch, Karl. 2024. "MPC-Based Secure Aggregation in Federated Learning: Overview, Protocols, & Google's Gboard – A-SIT Technologie."

Koch, Karl, Stephan Krenn, Tilen Marc, Stefan More, and Sebastian Ramacher. 2022. "KRAKEN: A Privacy Preserving Data Market for Authentic Data." Pp. 15–20 in *Proceedings of the 1st International Workshop on Data Economy*, *DE '22* . New York, NY, USA: Association for Computing Machinery.

Lycklama, Hidde, Lukas Burkhalter, Alexander Viand, Nicolas Küchler, and Anwar Hithnawi. 2023. "RoFL: Robustness of Secure Federated Learning." Pp. 453–76 in *2023 IEEE Symposium on Security and Privacy (SP)*, *S&P'23* .

McMahan, Brendan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. 2017. "Communication-Efficient Learning of Deep Networks from Decentralized Data (FL Intro Paper)." Pp. 1273–82 in *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics* . PMLR.

Wang, Zhipeng, Nanqing Dong, Jiahao Sun, William Knottenbelt, and Yike Guo. 2024. "zkFL: Zero-Knowledge Proof-Based Gradient Aggregation for Federated Learning."

Xing, Zhibo, Zijian Zhang, Meng Li, Jiamou Liu, Liehuang Zhu, Giovanni Russello, and Muhammad Rizwan Asghar. 2023. "(PZKP-FL) Zero-Knowledge Proof-Based Practical Federated Learning on Blockchain."

Zhu, Yizheng, Yuncheng Wu, Zhaojing Luo, Beng Chin Ooi, and Xiaokui Xiao. 2024. "(RiseFL) Secure and Verifiable Data Collaboration with Low-Cost Zero-Knowledge Proofs." *Proceedings of the VLDB Endowment* 17(9):2321–34. doi: 10.14778/3665844.3665860.

---

1. mdpi.com/1424-8220/23/5/2821 ↩
2. link.springer.com/article/10.1007/s11227-023-05364-3 ↩
3. nypost.com/2024/07/20/health/ai-detects-cancer-with-17-more-accuracy-than-doctors-ucla-study ↩