Duality Comments on EDPB Anonymisation Guidelines

To: European Data Protection Board

Re: Public consultation on Guidelines 02/2026 on Anonymisation (version 1.0, adopted 7 July 2026)

From: Duality Technologies, Inc.

Date: Oct 28, 2026

Introduction

Duality Technologies thanks the EDPB for the opportunity to comment on the draft Guidelines 02/2026 on Anonymisation. Duality develops and offers to the market a platform encompassing the main Privacy Enhancing Technologies (PETs), including Fully Homomorphic Encryption (FHE), Trusted Execution Environments (TEE), Federated Learning (FL) and Federated Analytics (FA), and Differential Privacy. Duality has deep expertise in practical implementations of solutions based on PETs in general and on FHE in particular. FHE allows organisations to analyse sensitive data and run models on it without decrypting it at any point during processing. Duality is also a lead contributor to the OpenFHE open-source FHE library. Duality's co-founders helped establish HomomorphicEncryption.org, the industry-academia-government consortium that produced the community's first security and API standards for FHE, and Duality's researchers continue to contribute to that work and to the ongoing ISO/IEC effort to standardise fully homomorphic encryption, including the security guidelines for implementing FHE that support it. Our comments draw on this experience.

We welcome the Guidelines' risk-based approach, under which anonymity is assessed from the perspective of each relevant entity, in line with *EDPS v SRB* (C-413/23 P), and the clear framework of three criteria. Our comments address one gap. The Guidelines consider encryption only as a way of restricting access to data (para 91). However, consideration must be given to situations where data can be *processed* while it remains encrypted. This matters because, under para 15, an entity processing data on behalf of a controller is to be assessed from the controller's perspective, regardless of whether it can access intelligible data.

In summary:

1. **FHE should be distinguished from conventional encryption in para 91.** With FHE, the entity performing the computation on FHE-encrypted data does not need, and never receives, the decryption key in order to perform the computation and therefore has no access to intelligible inputs, intermediate values or results.
2. **A processor computing on tokenised (identifiers replaced by tokens), FHE-encrypted data has no means reasonably likely to identify individuals.** Where the controller replaces identifiers with tokens and encrypts the data with FHE, retaining exclusive control of the token mapping and the decryption key, the processor's position is factually identical to that of an independent recipient of the same data. The perspective rule in para 15 should not override that factual assessment.
3. **Such data passes the Guidelines' own three criteria from the processor's perspective.**
4. **We propose conditions for this conclusion, together with amendments to paras 15 and 91 and a worked example.**

Comment 1: Distinguish FHE from conventional encryption (para 91)

Para 91 rightly notes that encryption is intended to be reversible and that it should be assessed whether entities with access to encrypted data could decrypt it, including through cryptanalysis. We agree. However, para 91 reflects conventional encryption, which protects data at rest and in transit. Under such traditional encryption schemes, indeed, to process such encrypted data, the processing entity must decrypt it, and therefore has access to intelligible data.

FHE differs in a way that is material to the Guidelines' analysis. Computation is performed directly on ciphertexts, the results are themselves generated while in encrypted form, and these encrypted results can only be opened by the controller - not the processor. This is because only the holder of the secret key can decrypt the results. The entity performing the computation does not hold the secret key, and at no point has access to intelligible inputs, intermediate values or outputs. It receives only ciphertexts and public evaluation keys, which allow computation but not decryption.

Reversibility. FHE is reversible only by the key holder. From the perspective of an entity without the key, reversal would require either obtaining the key from the controller or breaking the scheme. Under para 20, decryption controlled by another party is relevant only where it can be applied through means reasonably likely to be used.

Cryptanalysis. The FHE schemes in practical use (BGV, BFV, CKKS and TFHE/CGGI) are based on the Learning With Errors (LWE) problem and its ring variants. Parameters are selected under the community HomomorphicEncryption.org Security Standard to reach at least 128-bit security against known attacks. Lattice problems of this family also underpin NIST's post-quantum key-encapsulation standard (ML-KEM, FIPS 203), so these schemes are considered resistant to known quantum attacks. This is relevant to the requirement in para 29(e) to consider reasonably foreseeable technological developments. FHE is being standardised as ISO/IEC 28033 (in draft).

We therefore ask the EDPB to distinguish, in para 91, between encryption that must be removed before processing and encryption under which processing takes place.

Comment 2: Processors computing on tokenised, FHE-encrypted data (para 15)

The scenario. A controller replaces direct identifiers with opaque tokens, keeping the mapping. By “opaque tokens” we mean tokens that, to the processor, are indistinguishable from random values and cannot be reversed, recomputed or attributed to an individual without additional information held exclusively by the controller. They need not be generated by a truly random process (see Comment 4, condition 1, below). The controller encrypts all other attributes with FHE and keeps the secret key. It sends the processor the tokens, the ciphertexts and the public evaluation keys. The processor performs the computation on the controller's behalf and returns encrypted results, which only the controller can decrypt.

The processor's means. The processor cannot identify anyone through means reasonably likely to be used. The tokens are opaque to the processor, and the mapping is held only by the controller. Every attribute is encrypted, and the processor holds no key that allows decryption. Its factual capability is identical to that of an entity receiving the same ciphertexts and acting independently, which para 15 and Example 4 would assess from its own perspective.

Why para 15 should not override that assessment. We understand the rationale for para 15: preventing controllers from avoiding GDPR protections by outsourcing processing. That rationale does not apply here, for four reasons.

1. **The controller's obligations are unaffected.** Tokenisation, encryption, decryption and any use of the results remain the controller's processing of personal data, subject to all GDPR requirements, including a legal basis (para 38), transparency (para 40) and informing data subjects about recipients (as confirmed in *EDPS v SRB* and Example 2). Nothing is outsourced in a way that escapes GDPR. The only question is whether the processor, which cannot read the data, is itself processing personal data.
2. **Identifiability is a question of means, not of contract.** Recital 26 and the CJEU's case law turn on the means reasonably likely to be used. Two entities holding identical ciphertexts with identical capabilities should not reach different conclusions solely because one acts under the controller's instructions. In *EDPS v SRB*, the Court assessed identifiability by reference to the means of the recipient (Deloitte), which the SRB had itself engaged, rather than to that relationship. Deloitte acted as an independent valuer rather than as a processor, but the Court's reasoning rests on the recipient's means, and it therefore extends logically to cases where the recipient acts on the controller's behalf.
3. **No added protection for data subjects.** Applying processor obligations to an entity that cannot access intelligible data adds little protection, since it cannot misuse, disclose or lose anything intelligible. The protection lies in the encryption and in the controller's exclusive key control.
4. **Incentives.** Treating FHE-based processing the same as processing in the clear removes a legal incentive to adopt it, contrary to data protection by design (Article 25) and the explicit mention of encryption in Article 32.

The chain of means (paras 20–21). The only party able to decrypt is the controller. A decision by the controller to share its key or token mapping with the processor would be a new processing operation, within the controller's own control and fully subject to the GDPR. It is not a means reasonably likely to be used by the processor. Any third party that obtains the ciphertexts from the processor faces the same barrier, so onward transfer (para 21) does not change the conclusion.

Tokens repeated across computations. The conclusion does not depend on tokens being single-use. A controller may legitimately use the same token for an individual across several computations, for example to follow outcomes over time or to join tables. The processor can then recognise that records in different computations carry the same token, but every attribute linked in this way remains encrypted. Linking ciphertexts to other ciphertexts adds nothing intelligible: the processor learns no attribute of the person behind the token and has no means of attributing the token to an individual. Nor can it use the token to treat that person differently, as the third party does in Example 6, because it cannot evaluate the encrypted values and the results return, encrypted, to the controller. What matters is that the tokens are not disclosed together with intelligible attributes to any entity from which the processor could obtain data with means reasonably likely to be used. Otherwise the tokens could serve as the common identifier for linkage described in para 62.

Comment 3: The three criteria, from the processor's perspective

Applying the Guidelines' own framework to the processor in the scenario above:

Criterion	Assessment
No Record Isolation (paras 54–58)	The processor sees only opaque tokens and ciphertexts. FHE schemes are semantically secure (IND-CPA) and post-quantum secure: ciphertexts reveal nothing about the values they encrypt. Even where records are structurally unique, para 99 recognises that a unique record that is not intelligible does not allow singling out.
No Linkage (paras 59–65)	Opaque tokens match no dataset available to the processor, provided they are not disclosed elsewhere together with intelligible data. Where the same tokens recur across computations, the processor can link its own encrypted records, but under para 102 the linked information is re-tested: it still consists only of ciphertexts, so it again passes all three criteria. FHE encryption is randomised: the same value encrypted twice produces different ciphertexts, so attribute values cannot be matched across datasets.
No Inference (paras 66–82)	Semantic security guarantees that the processor can learn nothing about the plaintext from the ciphertexts beyond what is explicitly revealed, such as data size. This is the property the Guidelines' own test in footnotes 43–44 looks for: the processor's view is computationally indistinguishable whether or not any individual's record is included.

Residual information. The processor may see metadata such as the number of records, the size of the data, timing, the type of computation requested and, where tokens are reused, how often and in which computations a token appears. In typical deployments this does not allow identification, but it should be assessed as part of the analysis.

Output privacy is separate. This conclusion concerns the processor only. Once the controller decrypts the results, whether those results are anonymous, for example if published, must be assessed separately under the three criteria. FHE protects data during processing; it does not determine what the decrypted results reveal.

Comment 4: Conditions for the conclusion

We do not suggest that any use of FHE automatically makes data anonymous for a processor. We propose that the Guidelines recognise this outcome where the following conditions are met and documented (para 41):

1. **Tokenisation.** Direct identifiers are replaced by opaque tokens. The tokens need not be truly random. They may be generated randomly, by a keyed cryptographic function (such as a keyed hash, or encryption under a key held by the controller) or by a lookup table, provided that, to the processor, they are indistinguishable from random values and cannot be reversed or recomputed without information held exclusively by the controller (cf. para 50). Tokens derived by an unkeyed function, such as a plain hash of an identifier, do not meet this condition, because anyone could recompute them from candidate identifiers. The mapping, and any key used to generate the tokens, is held exclusively by the controller. The same tokens may be reused across computations, provided they are not disclosed together with intelligible attributes to any entity from which the processor could obtain data.
2. **Encryption of all other attributes** with a standardised FHE scheme, using parameters that meet a recognised security standard (at least 128-bit security).
3. **Exclusive key control.** The secret key is generated and held only by the controller. The processor receives only public and evaluation keys. This separation is enforced technically and organisationally.
4. **No return of decrypted results to the processor.** Encrypted results are returned and can be decrypted only by the controller. This matters because seeing decrypted results could, in some cases, increase the risk that the processor might work out the secret key.
5. **Metadata assessed.** Record counts, data sizes, timing, the computation performed and, where tokens are reused, the pattern in which they recur have been considered and do not allow identification.
6. **Controller obligations unaffected.** The controller remains responsible for all of its own processing, including its legal basis, transparency, informing data subjects about recipients, security and any DPIA.

Conditions 1 and 3 align with the EDPB's Guidelines 01/2025 on Pseudonymisation: the processor falls within the pseudonymisation domain, while the token mapping (and any tokenisation key) and the decryption key, as pseudonymisation secrets, are held exclusively outside it by the controller.

Variants in which the decryption key is split among several parties (threshold or multi-key FHE) raise additional questions about collusion between key holders, which the analysis in paras 20 and 34 would need to address. We would welcome guidance on these as well.

Proposed amendments

1. Para 91: add at the end:

> *Encryption schemes that allow computation on encrypted data, such as fully homomorphic encryption, differ in this respect: the entity performing the computation does not need, and need never have, access to the decryption key or to intelligible data. Where such schemes are implemented with state-of-the-art parameters and the decryption key remains under the exclusive control of another entity, decryption by the computing entity, whether by obtaining the key or by cryptanalysis, will generally not be a means reasonably likely to be used by that entity.*

2. New paragraph after para 15:

> *The rule in the previous paragraph does not prevent an assessment of the means reasonably likely to be used by the processing entity itself where that entity demonstrably has no access to intelligible data. This may be the case where (a) the controller has replaced direct identifiers with tokens that cannot be reversed, recomputed or attributed to an individual without information held exclusively by the controller, whether those tokens are used once or across several processing operations, (b) the remaining data is encrypted using a scheme that permits computation on encrypted data, and (c) the decryption keys remain under the exclusive control of the controller. In such cases, the information may be anonymous from the perspective of the processing entity. This is without prejudice to the controller's own obligations, for which the information remains personal data.*

3. New example after Example 3:

> ***Example 3a:** A hospital wishes to use a cloud analytics provider to compute statistics on patient records. Before transmission, the hospital replaces patient identifiers with tokens, keeping the mapping, and encrypts all other attributes using fully homomorphic encryption, keeping the secret key. It uses the same token for each patient in every monthly analysis, so that trends can be followed over time. The provider performs the computation on the encrypted data on the hospital's behalf and returns encrypted results, which only the hospital can decrypt. The information remains personal data for the hospital, which remains subject to all of its obligations, including informing patients about recipients. The provider, however, holds only tokens and ciphertexts, and has no access to the token mapping or the decryption key. Although it can recognise that records in successive analyses carry the same token, every attribute linked in this way remains encrypted. Assessed by reference to its own means, it cannot single out, link or draw inferences about any patient, and the information is anonymous from its perspective. This differs from Example 3, where the agency received intelligible excerpts that the retailer could identify and that the agency processed in the clear.*

4. Glossary: add:

> ***Fully homomorphic encryption (FHE):** An encryption scheme that allows computations to be performed directly on encrypted data, producing encrypted results that can be decrypted only with the secret key, without the computing entity ever having access to the data in intelligible form.*

> ***Opaque token:*** *A value that replaces a direct identifier and that, to any entity other than the controller, is indistinguishable from a random value and cannot be reversed, recomputed or attributed to an individual without additional information held exclusively by the controller. Opaque tokens may be generated randomly, by keyed cryptographic functions or by lookup tables.*

Closing

We support the EDPB's aim of ensuring that data is safely anonymised where possible and proportionate. Recognising the specific properties of FHE would give controllers legal certainty when adopting a technology that substantially reduces the exposure of personal data during processing, without weakening their obligations or the protection of data subjects. We would be glad to provide further technical detail or to contribute to any follow-up work.

Yours sincerely,

Rina Shainski, Co-founder, Chairwoman

Duality Technologies, Inc.

rina@dualitytech.com