Format-Preserving Encryption vs. Tokenization

Tokenization and FPE both address data protection but from an IT perspective, they have differences!

Lock Mechanisms

Tokenization uses an algorithm to generate the random data values, commonly called ‘tokens.’ The tokenization process replaces original data values with tokens, retaining the same format as the original data.

Encryption uses an algorithm and an encryption key to scramble the original data into an unreadable form. FPE refers to encrypting data in such a way that the output is in the same format as the original data.

Business or operational concerns

Tokenization		FPE
YES - the original data is replaced with a token, which retains format of the original data	**Avoid re-coding applications and re-structuring databases**	**YES** - the original data is encrypted and formatted in such a way that the format of the original data is retained
YES - hackers cannot reverse engineer tokenized data (or vice versa) as random data was used to generate the tokens	**Unbreakable?**	**NO** - the systematic encoding process is reversible with the right encryption key
YES - tokenization ‘replaces’ the original data with a token, therefore original data no longer exists	**Successfully removed sensitive data?**	**NO** - actual data is still there, it’s just scrambled, so technically, it’s not removed
YES - tokenization reduces compliance scope, as compliance auditing affects systems hosting sensitive data	**Helps reduce overall compliance burden?**	**NO** - although FPE meets compliance regulations, the systems still need to be audited, thus the burden of proof is still there
YES - tokens are generated by a centralized server which doesn’t require rotation or management of encryption keys	**Remove or reduce the burden of key management?**	**NO** - organizations need to rotate encryption keys on an annual basis, which adds to the operational management burden
YES - tokens are still usable in their protected state so users don’t need to have access to unprotected data	**Protects data even when user/admin credentials are leaked?**	**YES** - as long as access to the key manager is not compromised, data is protected
YES - the ANSI X9.119-2 standard governs the secure generation of tokens. Not all solutions implement the ANSI standard	**Standards-based approach?**	**YES** - the AES-FFx standard governs FPE. Note that currently only AES-FF1 is considered secure as FF2 and FF3 have known vulnerabilities
YES - stateless tokenization is ideal since the token server doesn’t replicate tokens across its nodes and doesn’t store any sensitive data ever.	**Can be implemented in a stateless fashion?**	**YES** - stateless key management allows for any key to be derived at any point in time and alleviates key replication issues. Stateless keys cannot be destroyed.

SecurDPS by comforte delivers both Tokenization & FPE. SecurDPS Enterprise combines the comforte AG patented, stateless tokenization algorithm and the proven, highly scalable and fault-tolerant architecture.

comforte

www.comforte.com