Analysis of VAES3 (FF2)

Summary

Draft SP 800-38G specifies three AES modes of operation for format-preserving encryption (FPE). One of the three modes, VAES3, was submitted to NIST by Joachim Vance of VeriFone Systems, Inc.; the submission document is posted at http://csrc.nist.gov/groups/ST/toolkit/BCM/documents/proposedmodes/ffx/ffx-ad-VAES3.pdf. In Draft SP 800-38G, VAES3 is specified under the name FF2. This note describes a theoretical chosen-plaintext attack that shows the security strength of FF2 is less than 128 bits.

Description of VAES3

VAES3 encrypts the plaintext and tweak, P and T, under a master key, K, in two stages: 1) a 128-bit subkey, SK, is generated from an encoding of the tweak by invoking the AES block cipher under the master key; and 2) the subkey is used as the key for a Feistel-based encryption, FEIST, of the plaintext. Symbolically,

$$SK = \mathrm{AES}_K(\text{encoding of } T)$$
$$\mathrm{VAES3}_K(P,T) = \mathrm{FEIST}_{SK}(P).$$

The Feistel-based encryption includes ten invocations of the AES block cipher as the pseudorandom round function, but this fact and the other details of the Feistel encryption do not affect the analysis below. The encoding of T includes the lengths of the plaintext and the tweak.

Chosen-Plaintext Attack

VAES3 invokes its internal Feistel encryption with a different subkey for every tweak. Therefore, many subkeys may be available as targets for an attack; moreover, given the right kind of information, the computational cost of a brute-force search is reduced.

In particular, suppose an attacker knew a set of ciphertexts $C_i = \mathrm{VAES3}_K(P,T_i)$, where the set is indexed by i; in other words, the "chosen plaintexts" are inputs to VAES3 in which distinct tweaks are paired to a single common plaintext, P. The actual values of P and T_i may be arbitrarily chosen.

The attacker computes $D_j = \mathrm{FEIST}_{Kj}(P)$ for some set of distinct subkeys, K_j, which also may be arbitrarily chosen. If $C_i = D_j$ for any pair of indices (i, j), then K_j is a candidate subkey for the attacker to investigate further. If K_j turns out to be a false positive, then the attacker searches for another candidate. However, if confirmed, the knowledge that $K_j = \mathrm{AES}_K(\text{encoding of } T_i)$ constitutes a significant breach of VAES3: for any other plaintext with the same length and with the same tweak T_i, the

attacker could encrypt the plaintext, or decrypt its ciphertext, without discovering the master key.

In order to investigate a candidate K_j, the attacker must learn/know $\text{VAES3}_K(Q,T_i)$ for some number of additional of plaintexts Q with the same length (and format) as P. In particular, if $\text{VAES3}_K(Q,T_i) \neq \text{FEIST}_{Kj}(Q)$ for any Q, then K_j is a false positive. The number of successful trials that are necessary to confirm K_j as a true match depends on the length and format of P: if P contains b bits of information, then approximately $\lceil 128/b \rceil$ successful trials are required.

Estimate of the Complexity of the Attack

The probability that the attacker would find a true match depends on 1) the number of chosen inputs available and 2) the number of Feistel encryptions that the attacker can generate of the common plaintext with different subkeys. In particular, if there are 2^u and 2^v of them, respectively, for positive integers u and v, then at least one match would be expected when $u + v \geq 128$. Consequently, the security level of VAES3 should be regarded as no higher than 128-u.

Although not really practical, this level of computation is at least conceivable for an important intended use of FPE, the encryption of the middle-six digits of credit card numbers, tweaked by the outer ten digits. Thus, a single plaintext could in principle have 10^{10} different tweaks, so the size of the chosen input set could approach 2^{34}. In this case, the analysis would have a good chance of revealing a subkey if the attacker could compute Feistel encryptions for up to 2^{94} candidate subkeys. Moreover, multiple tweaks could potentially be compromised by the same set of chosen inputs.

Note that the computational costs of the attack are independent of the length of the master key. In other words, the use of a 192-bit or 256-bit master key offers no additional security over 128-bit keys.