ANSI X9.82 Discussions

September, 2002

1. Section 7.2.1 (Seeds and Reseeding), page 9, bullet 2: If we need only n/2 bits of security for an application, can we get away with n/2 bits of entropy?
2. Section 7.2.1 (Seeds and Reseeding), page 10, last bullet: Should this check only be done when the NRBG is co-located with the DRBG, or should it always be done?
3. Section 7.2.4 (Feedback): What additional statements should be made about feedback?
4. Section 7.2.6 (Error State): What additional requirements should be made regarding the error state?
5. Section 7.3 (General Implementation Requirements): What additional requirements are needed?
6. Section 9: Should we allow less than the maximum entropy as long as it is still an acceptable amount? For example, if a SHA-1 hash is used with full entropy, the strength = 160. If we allowed as few as 80 bits of entropy (strength = 80), this might still be acceptable.
7. Do the subsections for each generator seem appropriate?
8. Section 9.1.1.2 (Description): The notion of "r bits" was not in the original specification. Do we want it? If yes, should we use it in the other generators?
9. Section 9.1.1.2 (Description): XKEY in the original specification was changes to V so as not to be confused with generators that really use keys (e.g., the X9.17 generator). Any discussion?
10. Do we want to come up with cycle periods for each generator?
11. Do we want to indicate the maximum time between resseding each generator?
12. Discussion of the new SHA RNG specifications in 9.1.2 and 9.1.4.
13. Section 9.1.2.1 (Properties): Do we want to require that the seed = N (the output block size of the hash function)?
14. Section 9.1.3.1 (Properties): Note that XSEED is now identified as UserInput (which is what it is).
15. Section 9.1.3.1 (Properties), page 20: X9.71 (HMAC) specifies that the key length must be ≥ the output length (e.g., 160 ≤ *keylen*).The FIPS specifies that the key length must be ≥ one half of the output length (e.g., 80 ≤ *keylen*). How do we deal with this difference?
16. Section 9.1.3.1 (Properties), page 20: Does it seem reasonable to make the statement "Depending on consuming application requirements and risks, the key may be fixed or may be replaced when seeds are replaced"?

17. Section 9.1.3.2 (Description), page 22, Note: The following statement came out of x9.42; how much applies if seed comes directly from a non-deterministic generator? "*SEED* and *UserInput* shall be entered using appropriate controls where dual control with split knowledge is required".
18. Section 9.2.1.1 (Properties), page 25, bullet 2: There is a statement that three distinct keys shall be used for TDES. Should we relax this to two keys or even to a single key? Do we even want to retain the X9.17 generator?

19. Section 10 (Assurance) provides a diagram on the thinking with regard to assurance for deterministic generators.

20. Section 10.3.1 (Overview): Should we discuss state a requirement for RBGs to be in FIPS 140-2 cryptomodules? Should this be stated in Part 1?
21. Section 10.3.1 (Overview), para. 1: Do we need to identify various conditions when testing should be performed?
22. Section 10.3.3 (Software/Firmware Integrity Test): Is an EDC good enough?
23. Section 10.3.4 (Critical Functions Test): What other critical functions are there in these generators? Do we even need to specify this? This could include the algorithm building blocks, etc. and could be specified for each generator.
24. Section 10.3.6 (Manual Key Entry Test): Is an EDC good enough? What if the person entering the information modifies the entry so that the EDC works or enters the same modified entry twice? Could we require something better in some cases, e.g., include a hash of the entry or enter under dual control?

25: Section 10.3.7 (Continuous RBG Test): Is this test needed?