Hash Functions and Pseudorandomness

September 30, 2005
Don B. Johnson, Entrust CygnaCom, djohnson@cygnacom.com

Up to now, the security attributes that hash functions have been designed to meet are the explicit criteria of the output block being non-invertible and collision-free, that is, it is a property of the output block considered as a whole. However, when one examines a hash function output block, it "looks" like a random string of bits. A little reflection reveals that if the output block really was non-random, then this could easily have large implications on achieving the explicit design criteria. Therefore, somewhat unofficially, hash functions are sometimes used for their (apparent) pseudorandomness properties, which is a property of bitstrings. Examples of this usage are in key derivation functions and pseudorandom number generators but also include hash MAC constructions. However, in some sense, using a hash for its supposed pseudorandomness property is putting the cart before the horse. Some people have even posited that such usage should not be done, as no official claims have been made for this supposed pseudorandomness property.

This paper recommends that the property of pseudorandomness for the output block be part of the explicit design criteria. This will put the methods that use a hash function for its pseudorandomness property on a firmer foundation.

A proposed pseudorandomness criterion for further discussion: Sufficient output created from incremented input passes statistical test suites for random data.

(Note that recursive output (that is, creating new output from old output) is not recommended to create test data, as it may eventually not appear random, due to the possibility of collisions and short cycles.)

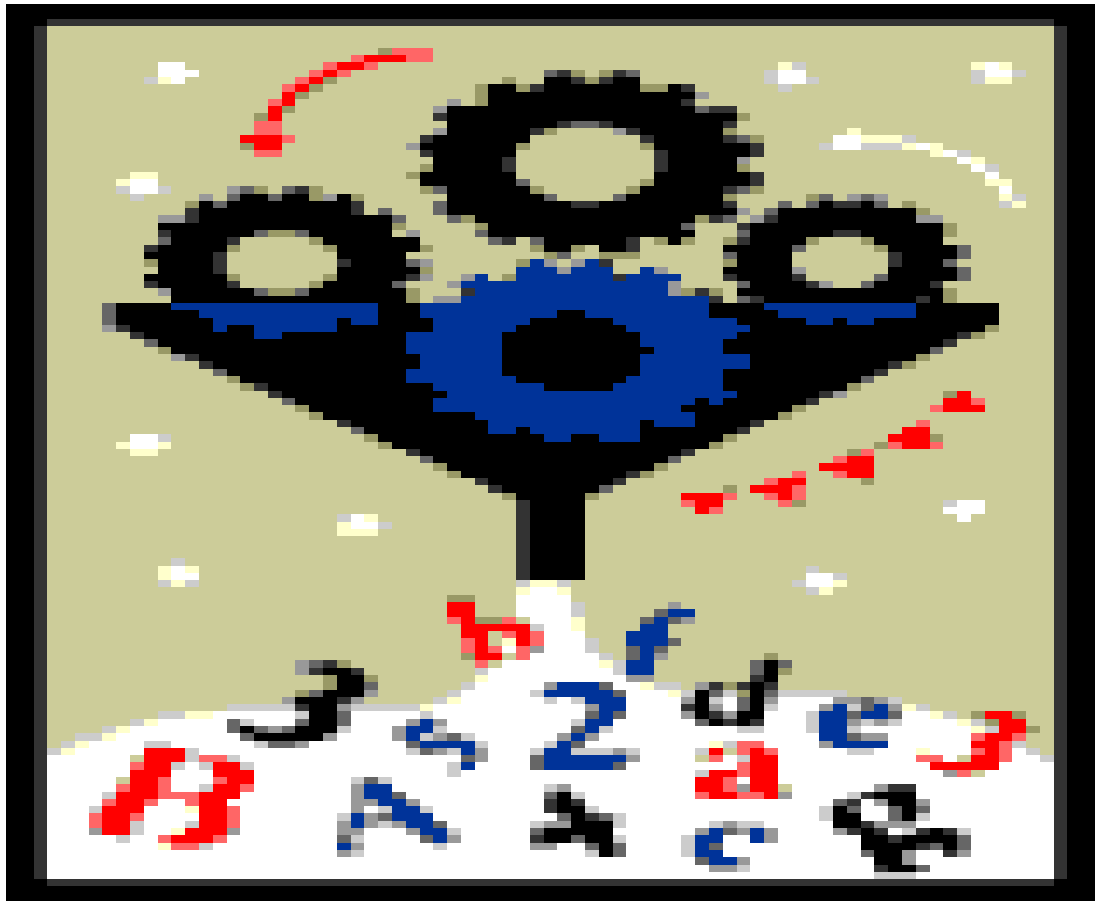
CYGNACOM
SOLUTIONS

Hash Functions & Pseudorandomness

Don B. Johnson
Entrust CygnaCom

CYGNACOM SOLUTIONS

Hash Function Model



CYGNACOM SOLUTIONS

Hash Design Criteria

- Hash function outputs are designed to be non-invertible and collision-resistant, which are **properties of the entire output block**.
- Many observe that the output “looks” random, leading to use of hash functions in key derivation functions, random number generation and hash MACs.

Putting The Cart Before the Horse?

- Using a hash function for its supposed pseudorandomness properties is suspect, unless it is an **explicit** criterion.
- The current hash function design criteria are properties of output blocks, while **randomness is a property of bits**.
- **Recommendation: Add the property of pseudorandomness to the explicit hash function design criteria.**

CYGNACOM SOLUTIONS

A Proposed Randomness Criteria

A proposed pseudorandomness criterion to stimulate discussion:

Sufficient output created from incremented input to the hash passes statistical test suites for random data.

Note: Recursive output (creating new output from old output) is **not** recommended to create test data, as it may not appear random, due to collisions or short cycles.