Jan 28, 2005 -- Lecture 5

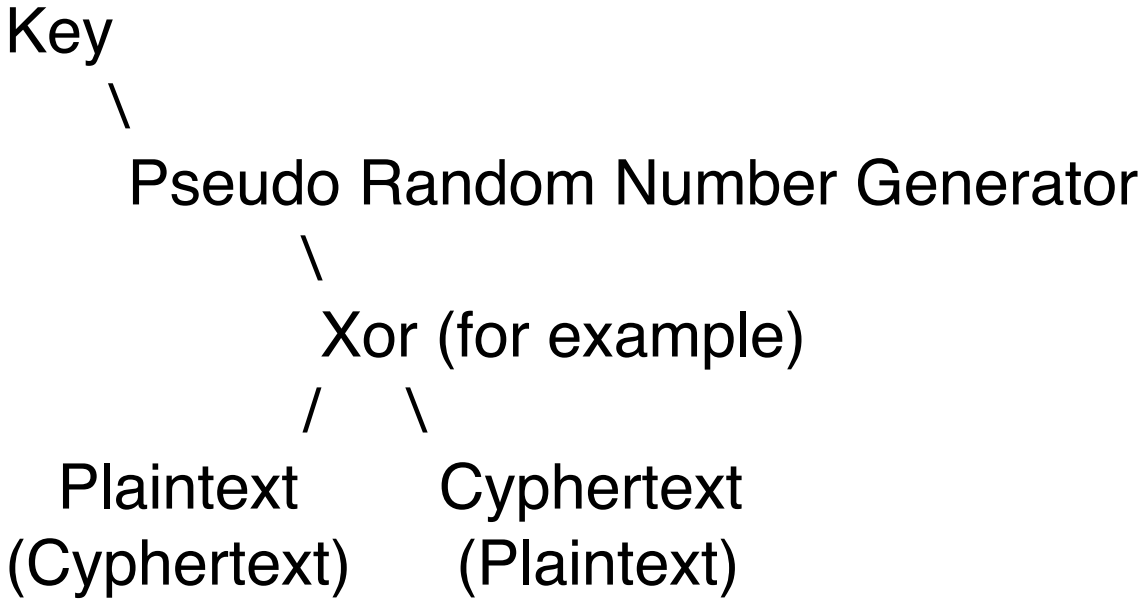
22C:169
Computer Security

Douglas W. Jones

Department of Computer Science

Stream Cyphers

An Idea For Symmetric Key Cyphers



Pseudo Random Number Generation

Not obviously possible --

Computers are deterministic

Properties:

S_1 = *seed* (used as cryptographic key)

$S_n = f(S_{n-1})$ (must be one to one)

$R_n = g(S_n)$ (may be many to one)

$S_n = S_{n+p}$ (p is a period of f)

Seed is mall and portable

Stream K is arbitrary length

Linear Congruential PRNG (RANDU)

$$S_n = (k\, S_{n-1}) \bmod (2^{31})$$

$$k = 65{,}539$$

This horrible PRNG lives on, despite the fact that it is awful, failing many obvious tests for randomness!

Linear Congruential PRNG

$$S_n = (k\ S_{n-1}) \bmod (2^{31}-1)$$

k = 16,807 or 48,271 or 69,621

Period of the Generator

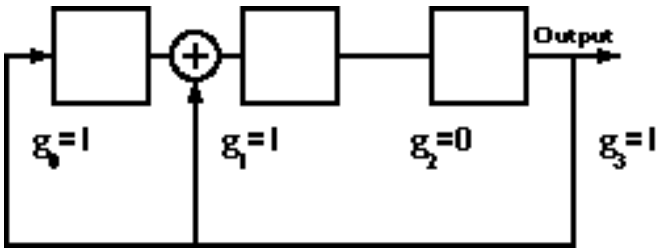
$$S_n = S_{n+2^{31}-2}$$

Source

Park and Miller, *CACM, 31*, 10 (Oct 1988)
Random number generators --
Good ones are hard to find

Linear Feedback Shift Registers

Shift register with XORed feedback:



Period, in bits, is 2^n-1 for an n bit register,
but only if the taps are in the right places!

Sn = **if** *odd*(Sn-1)
then (Sn-1 >> 1) $\oplus$ *mask*
else Sn-1 >> 1

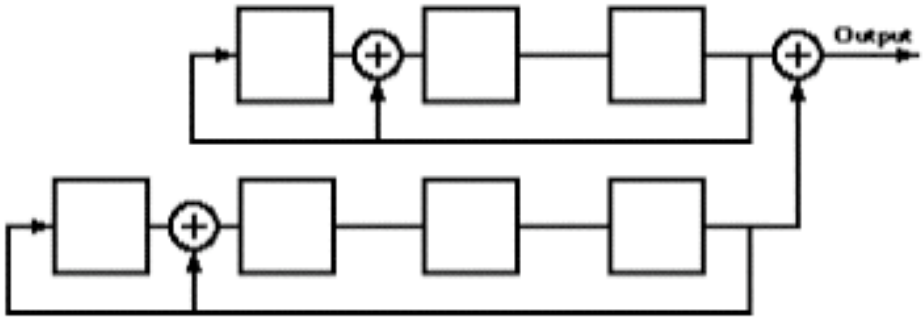
Rn = Sn mod 2

Combine weak PRNGs to make Strong?

$Seed_{COMBINED} = Seed_1$ II $Seed_2$?

Weak combining function and

$Period_1 \neq Period_2$



Entering Dangerous Territory!

Combine weak PRNGs to make Strong?

$Seed_{COMBINED} = Seed_1$ ll $Seed_2$?

Selective combining function and

$Period_1 = Period_2$

$S_n = \langle S_{1n}, S_{2j}, S_{3k} \rangle$

S_{n+1} = **if** even(S_{1n})

then $\langle S_{1n+1}, S_{2j+1}, S_{3k} \rangle$

else $\langle S_{1n+1}, S_{2j}, S_{3k+1} \rangle$

R_n = **if** *even*(S_{1n}) **then** S_{2j} **else** S_{3k}

Entering Dangerous Territory!

ISAAC

Indirect, shift, accumulate and count

Robert Jenkins, 1996

Seed: 256 integers, 32 bits each

Period: 2^{8295}

Must search 4.67×10^{1240} initial states for attack (square root of all possible).

Marina Pudovkina, 2001

A Known Plaintext Attack on the ISAAC ...

Seeding PRNGs

I. Use the text of the cryptographic key.
   keys must be small enough to carry

II. Seed with a genuine random number.
   must share number with remote user

III. Use small key to send big random key
   requires source of real randomness
   A Key Exchange Protocol

How to Generate Genuine Randomness:

I. Radioactive decay or cosmic rays
   inter-event intervals are exponential

II. Arrival times of eg: keypresses
   inter-event has no fixed distribution

III. Number of lines in system log file
   ad-hoc, system dependent.

Problem: *How many bits of randomness per second can we get from each source?*

How to combine randomness?

B_1 = random bits from source 1

B_2 = random bits from source 2

sources must be independent

B_1 || B_2 -- concatenation

risks loss of B_1 *if mod* 2^n

$B_1 \times B_2$ -- multiplication

does not produce prime results

Be Very Careful