Computer Verification of the Miller-Rabin Primality Test

Markus Kaiser, Johannes Buchmann

Juli 2006

Computer Verification in Cryptography

Aim: Construction of formal/computer proofs in cryptography

Aspects:

– Cryptographic Protocol
– Functional Correctness
– Correct Implementation
– Proof of Security

Formal Proof System

- Isabelle/HOL
- Higher-Order Logic
- Interactive Proof Constructions
- Database

Algorithm

Input: $k \in \mathbf{N}$, $2 < k$ odd, $0 < x$, $\gcd(x,k) = 1$, $k - 1 = 2^z v$

Output: $b = 0$ (composite) or $b = 1$ (prime)

$$prim(x, k, v, z) = b$$

Computer Verification (Example)

Computer Lemma: $x, k \in \mathbf{Z}$, k prime, $\gcd(x,k) = 1$, $2 < k$, $x < k$, $0 < z$, $0 < v \Longrightarrow prim(x, k, v, z) = 1$;

Computer Proof → correct implemented algorithm, augmented database

Conclusion

complex, but useful approach for verification in cryptography;[1]

[1]The authors are with the Technische Universität Darmstadt, 64289 Darmstadt, Germany. This work was partially funded by the German Federal Ministry of Education and Technology (BMBF) in the framework of the Verisoft project under grant 01 IS C38. The responsibility for this poster lies with the authors.

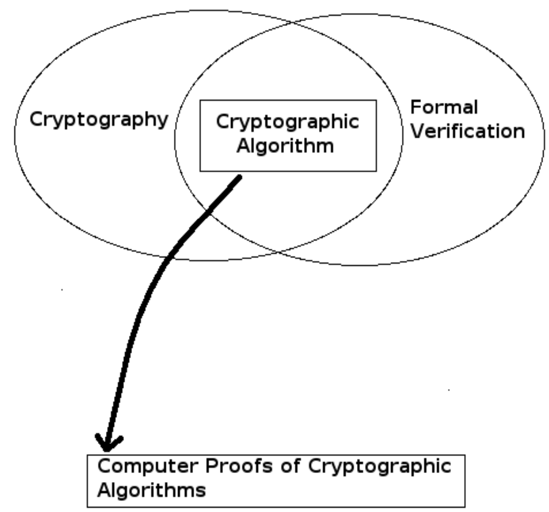

Figure 1: Cryptography and Formal Verification: correct proofs (minimum of errors), formally verified cryptographic client

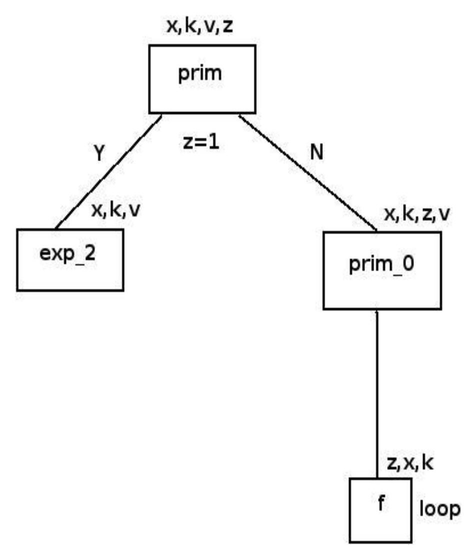

Figure 2: Illustration of Function $prim$ (Miller-Rabin Algorithm): $prim$ provides a case distinction ($z = 1$ or $z > 1$) what results in an application of exp_2 or $prim_0$ (with loop function f)