Cryptology ePrint Archive: Report 2013/124

Tamper Resilient Cryptography Without Self-Destruct

Ivan Damgaard and Sebastian Faust and Pratyay Mukherjee and Daniele Venturi

Abstract: We initiate a general study of schemes resilient to both tampering and leakage attacks. Tampering attacks are powerful cryptanalytic attacks where an adversary can change the secret state and observes the effect of such changes at the output. Our contributions are outlined below:

(1) We propose a general construction showing that any cryptographic primitive where the secret key can be chosen as a uniformly random string can be made secure against bounded tampering and leakage. This holds in a restricted model where the tampering functions must be chosen from a set of bounded size after the public parameters have been sampled. Our result covers pseudorandom functions, and many encryption and signature schemes.

(2) We show that standard ID and signature schemes constructed from a large class of Sigma-protocols (including the Okamoto scheme, for instance) are secure even if the adversary can arbitrarily tamper with the prover's state a bounded number of times and/or obtain some bounded amount of leakage. Interestingly, for the Okamoto scheme we can allow also independent tampering with the public parameters.

(3) We show a bounded tamper and leakage resilient CCA secure public key cryptosystem based on the DDH assumption. We first define a weaker CPA-like security notion that we can instantiate based on DDH, and then we give a general compiler that yields CCA-security with tamper and leakage resilience. This requires a public tamper-proof common reference string.

(4) Finally, we explain how to boost bounded tampering and leakage resilience (as in 2. and 3. above) to continuous tampering and leakage resilience, in the so-called floppy model where each user has a personal floppy (containing leak- and tamper-free information) which can be used to refresh the secret key (note that if the key is not updated, continuous tamper resilience is known to be impossible). For the case of ID schemes, we also show that if the underlying protocol is secure in the bounded retrieval model, then our compiler remains secure, even if the adversary can tamper with the computation performed by the device.

In some earlier work, the implementation of the tamper resilient primitive was assumed to be aware of the possibility of tampering, in that it would switch to a special mode and, e.g., self-destruct if tampering was detected. None of our results require this assumption.

Category / Keywords: Foundations / tamper resilience, general compiler, provable security

Date: received 28 Feb 2013, last revised 2 Mar 2013

Contact author: sfaust at cs au dk

Available formats: PDF | BibTeX Citation

Version: 20130305:124838 (All versions of this report)

Discussion forum: Show discussion | Start new discussion

[Cryptology ePrint archive]