

What is claimed is:

1. A method of providing secure code upgrades in a system comprising:

storing original code in an on-chip memory;

storing new code in an off-chip memory;

storing a new function table in the off-chip memory;

using the new function table to selectively execute code stored in the on-chip memory and the off-chip memory.

2. The method of claim 1 comprising storing an original function table in the on-chip memory for use before a new function table is stored in off-chip memory.

3. The method of claim 2 comprising determining at boot whether to use the new function table or the old function table.

4. The method of claim 3 wherein the determining comprises determining whether new code is stored in the off-chip memory.

5. The method of claim 1 comprising verifying new code to be loaded into the off-chip memory.

6. The method of claim 5 comprising only loading verified code into the off-chip memory.

7. The method of claim 1 comprising verifying a new function table to be loaded into the off-chip memory.

8. The method of claim 7 comprising only loading a verified function table into the off-chip memory.

9. The method of claim 1 comprising caching code from the off-chip memory into an on-chip cache.

10. The method of claim 1 comprising encrypting code before loading it into the off-chip memory.

11. The method of claim 1 wherein the off-chip memory comprises flash memory.

12. The method of claim 1 wherein the system comprises a TPM.

13. The method of claim 1 comprising authenticating code and function tables to be stored in the off-chip memory.

14. A processing system comprising:

an external data memory; and

a processing chip comprising:

at least one processor;

at least one data memory; and

an instruction module configured to select code routines for execution by the at least one processor from either the at least one data memory or the external data memory.

15. The system of claim 14 wherein the instruction module comprises an instruction multiplexer adapted to provide the code routines from either the at least one data memory or the external data memory to the at least one processor.

16. The system of claim 14 wherein the instruction module comprises an instruction cache adapted to cache code routines and a function table retrieved from the external data memory.

17. The system of claim 14 wherein the external data memory comprises a flash memory.

18. The system of claim 17 comprising a flash controller configured to control access to the flash memory.

19. The system of claim 14 wherein original execution code is stored in the at least one data memory.

20. The system of claim 14 wherein an original function table is stored in the at least one data memory.

21. The system of claim 14 wherein new execution code is stored in the external data memory.

22. The system of claim 14 wherein a new function table is stored in the external data memory.

23. The system of claim 14 wherein the instruction module selects code for execution from both the at least one data memory and the external data memory when new execution code is loaded into the external data memory.

24. The system of claim 23 wherein the instruction module uses a new function table stored in the external data memory to select code for execution from the at least one data memory or the external data memory.

25. A method of updating a key used for secure code loads comprising:

storing a public key of a first private-public key pair in at least one data memory associated with a cryptographic processing system;

receiving, by the cryptographic processing system, a public key of a second private-public key pair signed with a private key of the first private-public key pair;

verifying, by the cryptographic processing system, the signed public key using the public key of the first private-public key pair;

storing the public key of the second private-public key pair in the at least one data memory;

receiving, by the cryptographic processing system, secure code signed with a private key of the second private-public key pair;

verifying, by the cryptographic processing system, the signed secure code using the public key of the second private-public key pair; and

storing the secure code in the at least one data memory.

26. The method of claim 25 comprising determining, by the cryptographic processing system, whether to use the public key of the first private-public key pair or the public key of the second private-public key pair based on a presence of the public key of the second private-public key pair in the data memory.

27. The method of claim 25 wherein the secure code comprises code and a function table.

28. The method of claim 25 comprising encrypting the public key of the second private-public key pair before it is stored in the at least one data memory.

29. The method of claim 25 comprising encrypting the secure code before it is stored in the at least one data memory.

30. The method of claim 25 wherein a portion of the at least one data memory comprises a flash memory located external to the cryptographic processing system.

31. The method of claim 25 wherein the public key of the second private-public key pair comprises a portion of a secure code descriptor.

32. The method of claim 25 wherein the private key of the first private-public key pair and the private key of the second private-public key pair are securely maintained within a FIPS Level 3 hardware security module.