República Federativa do Brasil
Ministério da Economia
Instituto Nacional da Propriedade Industrial

(11) PI 0909294-3 B1

(22) Data do Depósito: 12/03/2009

(45) Data de Concessão: 10/09/2019

(54) Título: PROCESSO DE TORNAR SEGURA UMA EXECUÇÃO DE UM PROGRAMA PRINCIPAL

(51) Int.Cl.: G06F 21/00.

(30) Prioridade Unionista: 14/03/2008 FR 0801406.

(73) Titular(es): MORPHO.

(72) Inventor(es): DAVID DECROIX; LOUIS-PHILIPPE GONCALVES; GUILLAUME ROUDIERE.

(86) Pedido PCT: PCT FR2009000254 de 12/03/2009

(87) Publicação PCT: WO 2009/115712 de 24/09/2009

(85) Data do Início da Fase Nacional: 13/09/2010

(57) Resumo: PROCESSO DE TORNAR SEGURA UMA EXECUÇÃO DE UM PROGRAMA PRINCIPAL Processo de tornar segura uma execução de um programa principal que emprega funções imbricadas, compreendendo as etapas: - executar um programa gestor de segurança arranjado para atualizar uma lista de funções em curso, - assinalar um início de execução de cada função ao programa gestor de segurança e atualizar a lista de funções em curso, - sinalizar um final de execução de cada função ao programa gestor de segurança, - verificar após cada sinalização de final de execução de uma função que esta função é efetivamente a última a ter começado e caso contrário efetuar uma ação de proteção.

“PROCESSO DE TORNAR SEGURA UMA EXECUÇÃO DE UM PROGRAMA PRINCIPAL”

[0001] A presente invenção se refere a um processo de tornar segura a execução de um programa por um microprocessador, e notadamente um microprocessador de um cartão suporte de dados como um cartão de banco ou um cartão eletrônico de identificação.

ANTECEDENTES DA INVENÇÃO

[0002] Um tal cartão comporta um circuito integrado que compreende geralmente um microprocessador, uma memória volátil e uma memória não volátil. A memória não volátil contém dados pessoais tais como a identidade da pessoa para a qual o cartão foi emitido, uma senha de identificação, informações bancárias ou outras... e pelo menos um programa principal que permite a troca de informações entre o cartão e um leitor de um terminal ao qual ele estará ligado como um distribuidor automático de notas ou um terminal de pagamento. Este programa principal emprega várias funções imbricadas que permitem notadamente comunicar ao terminal alguns dos dados pessoais, recuperar um código apreendido pelo usuário sobre o terminal, operar cálculos sobre este código, por exemplo, para fins criptográficos ou de autenticação, comunicar ao terminal o resultado da autenticação... Estas funções são imbricadas no sentido que uma primeira função em execução chama uma segunda que em execução chamando uma terceira etc. ; o retorno da execução da segunda função e depois da primeira função se efetuando sucessivamente, após o fim da execução da terceira função e da segunda função respectivamente. Para permitir o retorno à função que chama, o microprocessador gera uma bateria de chamada de funções que lhe permite armazenar um endereço de retorno a cada vez que uma função é chamada de tal maneira que ao final da execução da função chamada, o microprocessador recupera o endereço memorizado na bateria de chamada de funções e aí posiciona seu ponteiro de execução para prosseguir a execução da função que chama.

[0003] Nestes cartões, uma das funções críticas em matéria de segurança consiste em decrescer um contador após cada falha de uma operação de

autenticação que consiste em verificar que o código apreendido no terminal é idêntico à senha de identificação memorizada no circuito integrado do cartão. Este número, no caso de um cartão de banco, é fixado geralmente em três enquanto o código propriamente comporta quatro números. O contador assim decrescido é remetido ao seu valor inicial após uma verificação bem-sucedida do código. Em contrapartida, após três falhas de verificação do código, o circuito integrado é bloqueado. Existe um modo de fraude que consiste em perturbar o funcionamento do cartão no final da operação de autenticação para impedir a execução da função que assegura o decréscimo do contador. Compreende-se que, se esta função não for efetuada, o fraudador pode tentar todas as combinações possíveis de quatro números até encontrar a boa.

[0004] Outro modo de fraude consiste em perturbar o funcionamento do microprocessador para levar a comportamentos inesperados do microprocessador que permite eventualmente recuperar dados que devem normalmente permanecer secretos. Por exemplo, poderia ser possível, após uma função ter sido chamada por uma função com chamada, de voltar à execução da função com chamada antes do fim da execução da função chamada.

OBJETO DA INVENÇÃO

[0005] Pela invenção, propõe-se um meio que permite proteger a execução de um programa que acarreta a chamada de funções imbricadas.

SUMÁRIO DA INVENÇÃO

[0006] Para esse efeito, prevê-se, de acordo com a invenção, um processo de tornar segura uma execução de um programa principal empregando funções imbricadas, compreendendo as etapas:

- executar um programa gestor de segurança arranjado para atualizar uma lista de funções em curso,

- assinalar ao programa gestor de segurança um início de execução de cada função e atualizar a lista de funções em curso,

- assinalar um final de execução de cada função ao programa gestor de segurança,

- verificar após cada sinalização de final de execução de uma função que esta função é efetivamente a última a ter começado,

- no caso de falha da verificação, realizar uma ação de proteção selecionada em função de um nível de segurança relacionado a cada função.

[0007] Assim, a lista mantida pelo programa gestor de segurança permite detectar um funcionamento anormal do programa principal e mais particularmente de chamada das funções.

[0008] De preferência, a ação de proteção consiste, por exemplo, na emissão de uma alerta ou uma interrupção do programa principal.

[0009] De acordo com um modo de realização particular, a lista de funções é uma tabela que comporta uma sequência de campos destinados a receber um identificador de funções, o processo compreendendo as etapas:

- inicializar os campos,

- à sinalização do início de execução de uma função, inscrever o identificador da função no primeiro campo livre.

[0010] Este modo de realização é particularmente simples e eficaz e seu emprego provoca apenas um emprego limitado dos recursos de cálculo e de memória.

[0011] Vantajosamente, o processo compreende a etapa, à sinalização do final de execução de uma função chamada por uma função com chamada, memorizar a função chamada e fazer verificar pela função com chamada com o gestor de segurança que a última função executada é a função chamada.

[0012] Outras características e vantagens da invenção surgirão na leitura da descrição que segue de um modo de emprego particular não limitativo da invenção.

BREVE DESCRIÇÃO DOS DESENHOS

[0013] Será feito referência aos desenhos anexados, dentre os quais:

- a figura 1 é uma vista esquemática de um cartão de dados utilizável para a aplicação do processo da invenção,

- a figura 2 é uma representação que ilustra esquematicamente o desenrolar de um programa principal no quadro do processo de acordo com a

invenção,

- a figura 3 é uma representação que ilustra esquematicamente o desenrolar de um programa gestor de segurança no quadro do processo de acordo com a invenção.

DESCRIÇÃO DETALHADA DA INVENÇÃO

[0014] Em referência às figuras, a invenção é aqui descrita em aplicação a um cartão de dados 1 que comporta um circuito integrado, geralmente designado em 2, compreendendo um microprocessador 3 ligado a uma memória volátil 4, por exemplo de tipo RAM (Memória de Acesso Aleatório), e a uma memória não volátil 5, por exemplo de tipo EEPROM (Memória de Leitura Programável Eletricamente Apagável) ou ROM (Memória Somente de Leitura).

[0015] A memória não volátil 5 contém um programa principal, simbolizado em 6 na figura 2, empregando funções F1, F2, F3 imbricadas. A função F1 é arranjada para, durante sua execução, chamar a função F2 que é arranjada para, durante sua execução, chamar a função F3. Ao final da execução da função F3, a função F2 prossegue e no final da execução da função F2, a função F1 prossegue. A natureza das funções F1, F2 e F3 importa pouco, estas funções que asseguram, por exemplo, as operações de recuperação de um valor de um código apreendido em um terminal ao qual liga o cartão ao circuito integrado, operações de cálculos, verificação deste código, codificação, de aumento ou decréscimo de um contador, inscrição de uma memória em uma outra...

[0016] Cada função F1, F2 e F3 possui um identificador ID-F1, ID-F2 e ID-F3 que designa de maneira única as referidas funções. O identificador das funções é construído de maneira a conter sob forma codificada informações próprias para função e notadamente um nível de segurança.

[0017] A memória não volátil 5 contém, além disso, um programa gestor de segurança, simbolizado por 7 na figura 3, arranjado para estabelecer e atualizar uma lista de funções em execução ou bateria de chamada de funções. Esta lista se apresenta sob forma de tabela 8 que comporta uma sequência de campos 9 destinados a receber a identificação das funções em execução. O programa gestor

de segurança 7 utiliza, de maneira conhecida em si (e consequentemente não descrito detalhadamente aqui), um índice para apontar sobre o primeiro campo livre da bateria. O programa gestor de segurança tem, além disso, acesso a uma tabela que coloca em relação ações de proteção a se efetuar em função dos níveis de segurança codificados no identificador das funções.

[0018] O processo da invenção será agora descrito.

[0019] Em funcionamento, quando o microprocessador 3 do cartão 1 executa o programa principal 6, ele executa simultaneamente o programa gestor de segurança 7.

[0020] Ao arranque, os campos 9 da tabela 8 são inicializados inscrevendo um valor padrão, aqui NOFCT, e o índice aponta no primeiro campo que contém este valor por padrão (igualmente chamado campo livre).

[0021] Quando o programa principal executa a função F1, a função F1 assinala o início de sua execução ao programa gestor de segurança 7 através de uma interface FONCTION-BEGIN (tal interface é conhecida em si). O programa gestor de segurança 7 inscreve então o identificador ID-F1 da função 1 no primeiro campo 9 livre da tabela 8 e faz apontar seu índice sobre o campo 9 seguinte que se torna o primeiro campo livre.

[0022] Quando a função F1 chama a função F2, a função F2 assinala o início da sua execução ao programa gestor de segurança por meio da interface FONCTION-BEGIN. O programa gestor de segurança 7 inscreve então o identificador da função F2 (aqui ID-F2) no campo 9 apontado e depois faz apontar o índice imediatamente 9 seguinte.

[0023] Quando a função F2 chama a função F3, a função F3 assinala o início de sua execução para o programa gestor de segurança 7 que inscreve no campo apontado pelo índice o identificador ID-F3 da função F3 e aponta o índice sobre o campo 9 seguinte.

[0024] Quando a execução da função F3 termina, a função F3 assinala o fim de sua execução ao programa gestor de segurança 7 através de uma interface FONCTION-END (tal interface é conhecida em si). O programa gestor de segurança

7 verifica enquanto a função F3 que assinala o final de sua execução é efetivamente a última a ter começado sua execução. Se for este o caso, a execução da função F2 prossegue até seu final, a função F2 assinalando então para o programa gestor de segurança 7 o final de sua execução através da interface FONCTION-END. O programa gestor de segurança 7 verifica enquanto a função F2 é efetivamente a última a ter começado sua execução. Se for este o caso, a execução da função F1 retoma até seu final.

[0025] Na hipótese da função que assinala para o programa gestor de segurança 7 o final de sua execução não ser o último a ter começado esta última, o programa gestor de segurança 7 efetua uma ação de proteção. Esta ação de proteção é aqui uma alerta emitida para a destinação do programa principal 6 que interrompe sua execução. A ação de proteção pode igualmente ser traduzida em um bloqueio do cartão, o desbloqueio podendo ser realizado, por exemplo, apenas pelo organismo emissor do cartão após verificação da identificação do detentor desta última. A ação de proteção pode ser uma contramedida de segurança. Por contramedida de segurança, entende-se um meio que permite notadamente evitar uma recuperação de informações por um defraudador, por exemplo:

- fazendo variar a frequência de relógio,
- inserindo de maneira aleatória instruções para modificar a duração de execução do código, as emissões eletromagnéticas ou o consumo energético durante a execução do código...

[0026] No início da execução da função, o programa gestor de segurança determina a partir do identificador desta função qual(is) ação(ões) de proteção ou de contra medida(s) de segurança(s) ativar. Em variante, quando o final de execução de uma função intervém enquanto esta função não é a última a ter começado, o programa gestor de segurança pode ser arranjado para deduzir do identificador desta função a ação de proteção a executar.

[0027] O programa gestor de segurança pode igualmente ser arranjado para ativar pelo menos uma contra medida de segurança no início da execução de uma função. O identificador da função é, então de preferência, determinado para indicar

ao programa gestor de segurança a necessidade da ativação de tal medida sem necessariamente constituir uma indicação da natureza desta contra medida (esta contra medida pode ser a mesma para todas as funções que necessitam desta ativação).

[0028] Em variante ainda, o processo pode compreender a etapa de memorizar a última função executada quando esta assinala o final de sua execução. A função que chamou a última função executada pode então interrogar o programa gestor de segurança para lhe pedir controlar a coerência entre a execução nominal do programa e a execução real. No exemplo descrito acima, quando a função F3 assinalou o final de sua execução, a função F2 pode pedir ao programa gestor de segurança verificar que é efetivamente a função F3 que acaba de terminar.

[0029] Naturalmente, a invenção não se limita ao modo de realização descrito e pode-se trazer aqui variantes de realização sem sair do quadro da invenção tal como definida pelas reivindicações.

[0030] Em particular, de acordo com uma variante, pode-se prever uma tabela de chamadas possíveis de funções por cada função do programa principal. A título de exemplo, se a função F2 chama a função F3, o programa gestor de segurança controla que a função F2 tem efetivamente o direito de chamar a função F3. É igualmente possível afetar as funções em grupos e elaborar um identificador de funções que unem estas aos grupos aos quais elas pertencem para permitir ao programa gestor de segurança verificar que as funções que são chamadas mutuamente pertencem efetivamente ao mesmo grupo.

REIVINDICAÇÕES

1. Processo de tornar segura uma execução de um programa principal que implementa funções imbricadas, caracterizado pelo fato de que compreendendo as etapas de:

- executar um programa gestor de segurança disposto para atualizar uma lista de identificadores de função em curso,

- informar ao programa gestor de segurança um início de execução de cada função do programa principal e atualizar a lista de identificadores de função em curso,

- informar ao programa gestor de segurança um final de execução de cada função,

- após ser informado de cada final de execução de uma função, verificar que a função é efetivamente a função a ter começado mais recentemente; e

- no caso de falha da verificação, realizar uma ação de proteção que é selecionada em função de um nível de segurança associado a cada função;

em que a lista de identificadores de função é uma tabela compreendendo uma sequência de campos para identificadores de função respectivos, o processo compreendendo as etapas de:

- inicializar os campos e, sendo informado do início de execução de uma função, escrever o identificador da função no primeiro campo disponível; e em que o identificador de cada função contém um nível de segurança associado à função, e o programa gestor de segurança tem acesso a uma tabela associando ações de proteção a serem realizadas em função dos níveis de segurança.

2. Processo de acordo com a reivindicação 1, caracterizado pelo fato de que a ação de proteção consiste em emitir um alerta.

3. Processo de acordo com a reivindicação 1, caracterizado pelo fato de que a ação de proteção consiste em interromper o programa principal.

4. Processo de acordo com a reivindicação 1, caracterizado pelo fato de que o programa gestor de segurança tem acesso a uma tabela de chamadas possíveis de funções de cada função do programa principal.

5. Processo de acordo com a reivindicação 1, caracterizado pelo fato de que ao ser informado do final de execução de uma função chamada por uma função que chama, o método consiste em armazenar a função que é chamada e fazer com que a função que chama o gestor de segurança verifique que a função executada mais recentemente é a função chamada.

6. Processo de acordo com a reivindicação 1, caracterizado pelo fato de que, no início de execução de uma função, o programa gestor de segurança ativa pelo menos uma contramedida de segurança.

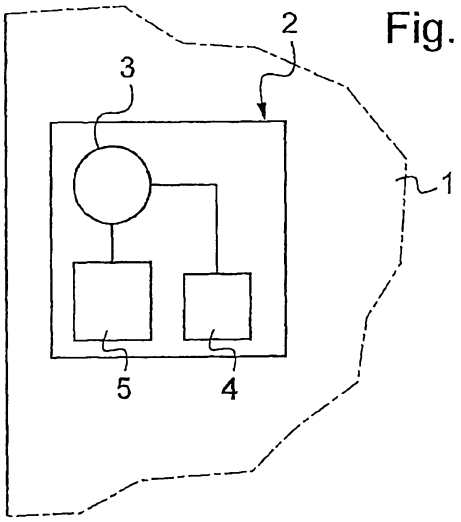
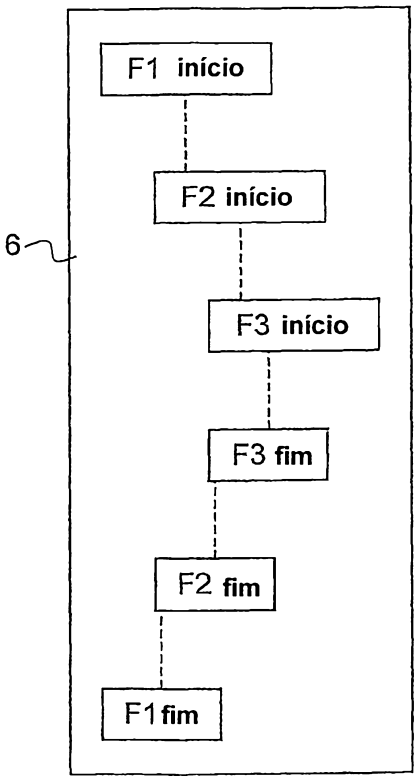


Fig.1

Fig.2



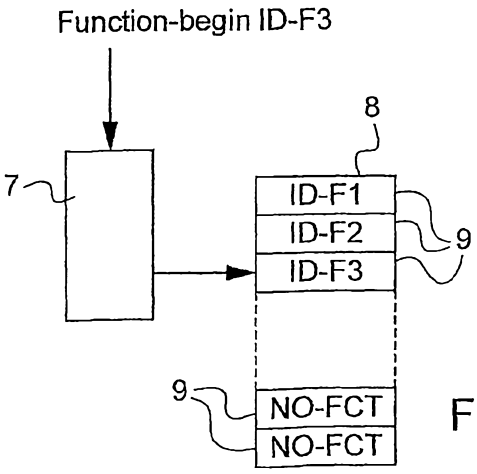


Fig.3