

MÉTODO PARA PROTEGER UN TERMINAL DE PAGO

CAMPO DE LA INVENCIÓN

La presente invención se refiere a un método para proteger un terminal de pago, en particular mediante tarjeta bancaria. La presente invención también se refiere a un terminal de pago seguro, especialmente mediante tarjeta bancaria.

ESTADO DE LA TÉCNICA

Los terminales de pago mediante tarjeta bancaria están dedicados al pago (como un terminal de punto de venta con una clave de autenticación por código confidencial integrado o separado) o a la distribución de productos (como un cajero automático o una distribución de autoservicio gasolina).

Los terminales de pago con tarjeta bancaria deben cumplir con los estándares de seguridad PCI PTS (para "Payment Card Industry Pin Transaction Security" en inglés) o "Industria de Tarjeta de Pago" y "Seguridad de Transacción de Pin". La última versión del estándar "PCI PTS V5" impone estándares de protección relacionados con la seguridad física activa de un terminal de pago.

La seguridad física activa según el estándar PCI PTS consiste en proteger las claves de cifrado (así como otros datos sensibles como un número

de tarjeta, un número de identificación personal del titular de la tarjeta, etc.) contenidos en el terminal de pago contra, entre otros:

- intrusión física por perforación u otro;
- apertura del terminal;
- el uso del terminal en zonas de temperatura donde ya no está calificado;
- variación de las frecuencias de reloj en un intento de hacer funcionar el terminal en situaciones en las que no está calificado; y
- el funcionamiento del terminal con tensiones demasiado bajas o demasiado altas.

En consecuencia, en el caso de que la seguridad física del terminal se vea comprometida, se deberán borrar todas las claves de cifrado almacenadas en el terminal para proteger las transacciones, identidades, aplicaciones, etc.

Los métodos actuales para lograr la seguridad física PCI PTS consisten en utilizar microprocesadores específicos desarrollados precisamente para este mercado de seguridad. Estos integran un circuito de hardware capaz de gestionar y controlar las protecciones físicas requeridas por el estándar PCI PTS. Estos microprocesadores de seguridad son microprocesadores criptográficos dedicados, especialmente diseñados para proteger las claves de cifrado durante todo su ciclo de vida.

Sin embargo, estos microprocesadores seguros, que dependen de mecanismos de hardware específicos, tienen un costo significativo.

Además, si se detecta una anomalía de seguridad física, el terminal requiere un regreso a mantenimiento para su puesta en funcionamiento, que puede resultar perjudicial en el caso de que la anomalía no corresponda a un ataque real de seguridad física.

Por lo tanto, existe la necesidad de proporcionar una solución de pago bancario menos costosa, al tiempo de ofrecer un nivel de seguridad por lo menos equivalente a las soluciones actuales, que permita proteger las claves criptográficas y brindar servicios de cifrado, descifrado, de autenticación y de firma electrónica para todo tipo de aplicaciones.

También existe la necesidad de proporcionar una solución que permita diferenciar fácilmente las anomalías de funcionamiento de un ataque de seguridad física real.

DIVULGACIÓN DE LA INVENCION

El objetivo de la presente invención es paliar las necesidades descritas anteriormente proponiendo un método de protección de software de un terminal de pago intentando mantener de forma segura las claves de cifrado de dicho terminal.

Más específicamente, la invención tiene como objetivo un método para proteger un terminal de pago, comprendiendo dicho terminal de pago:

- una clave de cifrado de datos a la que se accede para el cifrado de datos para asegurar;
- un módulo de vigilancia de por lo menos un parámetro físico del terminal que comprende un medio de medición de dicho parámetro físico;
- un microcontrolador;
- una unidad de contador del microcontrolador;
- un reloj, la unidad de contador está subordinada a dicho reloj para incrementar a un intervalo regular un valor de contador de la unidad de contador;

dicho método comprende las siguientes etapas:

- medición un parámetro físico por los medios de medición del dispositivo de vigilancia;
- comparación del valor medido con un valor umbral predeterminado, correspondiente a un ataque físico a dicho terminal por parte del dispositivo de vigilancia;
- eliminación del acceso a la clave de cifrado de datos por parte del microprocesador si se cumple una de las siguientes condiciones:
 - o por lo menos un valor de los parámetros físicos es significativo de un ataque físico de dicho terminal;
 - o el valor de contador es mayor que un valor umbral predefinido;
- reinicio de la unidad de contador por el dispositivo de vigilancia; y en el que las etapas de medición, comparación y restablecimiento se realizan periódicamente.

Ventajosamente, pero opcionalmente, el sistema de acuerdo con la invención también puede comprender por lo menos una de las siguientes características:

- la clave de cifrado se genera combinando una pluralidad de claves almacenadas en por lo menos una memoria del terminal;
- la etapa de eliminación implementa los métodos de eliminación separados para la primera clave almacenada en una zona de memoria de tipo de registro del microprocesador, y la segunda clave almacenada en una zona de memoria de tipo de memoria volátil del microprocesador;
- la clave de cifrado se genera mediante la combinación de una primera y una segunda clave, la primera clave se almacena en una zona de memoria del tipo de registro del microprocesador, la segunda clave se almacena en un área de memoria del tipo de memoria volátil del microprocesador, y en el que la eliminación comprende eliminar por lo menos una clave de la zona de memoria asociada;
- estando incluido el período de ejecución T1 aleatoriamente en un intervalo de valores predefinidos [T1a; T1 b];
 - el método (100) también comprende las siguientes etapas:
 - comprobar que el valor del retardo predefinido T1 está incluido en un rango de valores predefinido, y si el valor no está en el rango:
 - eliminación de por lo menos una clave de la pluralidad de claves;
- el terminal de pago también incluye un segundo reloj con una frecuencia predefinida para controlar el período de ejecución T1, estando dicho reloj configurado para transmitir en una etapa una interrupción al

microcontrolador para la ejecución periódica de las etapas de medición, de comparación y de reinicio;

- el terminal también incluye un tercer reloj con una frecuencia predefinida y en el que el método comprende una etapa de verificación de la frecuencia del segundo reloj que comprende:
 - incremento de una unidad de contador en cada señal de reloj del segundo reloj;
 - determinación de una frecuencia calculando el número de incrementos en un ciclo de reloj del tercer reloj;
 - comparación de la frecuencia calculada con un intervalo de frecuencia predeterminado; y si la frecuencia calculada no es el intervalo;
 - eliminación de por lo menos una clave de la pluralidad de claves;
- el terminal también incluye una etapa de verificación de la frecuencia del tercer reloj que comprende:
 - incremento de una unidad de contador en cada señal de reloj del tercer reloj;
 - determinación de una frecuencia calculando el número de incrementos en un ciclo de reloj del primer reloj;
 - comparación de la frecuencia calculada con un intervalo de frecuencia predeterminado; y si la frecuencia calculada no es el intervalo;
 - eliminación de por lo menos una clave de la pluralidad de claves;
- el terminal también comprende un filtro de paso de banda configurado para filtrar una banda de frecuencia predefinida y para recibir una señal

de reloj del primer reloj, y dicho método también comprende una etapa que comprende la eliminación de por lo menos una clave de la pluralidad de claves si no hay señal presente en la salida del filtro;

- el dispositivo de vigilancia de por lo menos un parámetro físico del terminal comprende uno o más medios de medida configurados para:
 - medir el voltaje de alimentación del terminal de pago;
 - medir la temperatura de funcionamiento del terminal de pago;
 - detectar la apertura del terminal de pago; y
 - detectar la perforación del terminal de pago.
- el dispositivo de vigilancia está configurado para emitir una señal de violación significativa de un ataque físico a dicho terminal; y en que el microcontrolador comprende por lo menos un puerto de recepción de dicha señal de violación; y en el que el acceso a la clave de cifrado de datos se elimina al recibir dicha señal en dicho puerto de recepción;
- el microprocesador también está configurado para, antes de la eliminación del acceso a la clave de cifrado de datos, ejecutar un algoritmo de filtrado tras la recepción de la señal de violación, detectando dicho algoritmo de filtrado si la señal de violación es significativo de un ataque físico a dicho terminal, y en este caso proceder a la eliminación del acceso a la clave de cifrado de datos;

Dicho método tiene la ventaja de utilizar una arquitectura de seguridad de software basada en un microcontrolador estándar de uso general para ejecutar esta arquitectura. Dado que el método cumple con los requisitos de la industria de tarjetas de pago, la arquitectura del software brinda protección contra

ataques físicos y ataques de software para garantizar la confidencialidad y autenticidad de los datos. Por lo tanto, es posible construir un punto de venta sin el uso de un "procesador seguro", lo que hace que la solución propuesta sea más económica.

El objetivo de la invención es también un producto de programa informático que comprende instrucciones de código para la ejecución de un método para proteger un terminal de pago, como el que se describió anteriormente, cuando este programa es ejecutado por un procesador de dicho terminal. de pago.

La invención también tiene como objetivo un terminal de pago seguro que comprende:

- una clave de cifrado de datos, accediendo a dicha clave de cifrado para el cifrado de datos a proteger;
- un dispositivo para vigilar por lo menos un parámetro físico del terminal que comprende medios de medición de dicho parámetro físico;
- un microcontrolador;
- una unidad de contador del microcontrolador,
- un reloj, siendo la unidad de contador esclava de dicho reloj para incrementar a un intervalo regular un valor de contador de la unidad de contador;
- dicho dispositivo de vigilancia está configurado para periódicamente:
- medir un parámetro físico por los medios de medición;

- comparar el valor medido con un valor umbral predeterminado, correspondiente a un ataque físico a dicho terminal por parte del dispositivo de vigilancia;
- eliminar el acceso a la clave de cifrado de datos (MK) si se cumple alguna de las siguientes condiciones:
 - o por lo menos un valor de los parámetros físicos es significativo de un ataque físico a dicho terminal;
 - o el valor de contador es mayor que un valor umbral predefinido; y
- reiniciar la unidad de contador.

Ventajosamente, pero opcionalmente, el terminal de pago seguro de acuerdo con la invención también puede configurarse para la puesta en funcionamiento de un método de acuerdo con una de las características descritas anteriormente.

DESCRIPCION DE FIGURAS

Otras características, objetivos y ventajas de la invención se desprenderán de la siguiente descripción, que es puramente ilustrativa y no limitativa, y que debe leerse con referencia a los dibujos adjuntos, en los que:

- La Figura 1 ilustra una arquitectura de hardware de un terminal seguro que implementa un método para proteger dicho terminal de acuerdo con la invención;
- La Figura 2 ilustra con más detalle la arquitectura de hardware del terminal seguro de acuerdo con la invención;

- La Figura 3 ilustra las etapas de un método para proteger el terminal de acuerdo con la invención.

DESCRIPCIÓN DETALLADA DE LA INVENCION

Con referencia a la Figura 1, se ilustra un terminal de pago 1.

Dicho terminal 1 comprende un microcontrolador 10, por ejemplo basado en una arquitectura electrónica basada en un núcleo ARM, tal como un microcontrolador del tipo STM32 (microcontrolador de 32 bits). Esto generalmente comprende un procesador, memorias (memoria de solo lectura y memoria de acceso aleatorio), unidades periféricas e interfaces de entrada-salida.

El microcontrolador 10 también incluye un reloj de tiempo real interno (término a veces abreviado como HTR, llamado RTC para "Real-Time Clock" en inglés), lo que permite un recuento muy preciso de tiempo, para fechar o desencadenar eventos según la hora. Este reloj utiliza, por ejemplo, un cuarzo piezoeléctrico.

El reloj de tiempo real requiere una señal de reloj o un oscilador de cristal para mantener el tiempo. El microcontrolador 10, por lo tanto, comprende medios de interacción con un reloj externo 13 (llamado LSE para "Low Speed External" en inglés) utilizado para controlar el reloj de tiempo real interno. Dicho reloj 13, incluido en el terminal de pago 1, proporciona preferentemente una

frecuencia de reloj entre aproximadamente 30 y aproximadamente 34 KHz, preferentemente 32.768 KHz.

El microcontrolador 10 también incluye medios de suministro 12, por ejemplo, una batería donde las celdas proporcionan la fuente de alimentación eléctrica principal. Cuando se desconecta la alimentación principal, una fuente de alimentación de respaldo (como una pila de botón interna) mantiene el terminal 1 encendido.

Ventajosamente, el terminal de pago está diseñado para operar en un rango de voltaje predeterminado.

Preferentemente, el voltaje de alimentación principal (V_{cc}) se regula a 3.3 V para alimentar todo el terminal de pago 1.

El microcontrolador 10 también comprende medios de interacción con componentes de hardware de seguridad (por ejemplo, mediante puertos de entrada / salida, conocidos como GPIO por General Purpose Input/Output en inglés) que se describe con más detalle en el resto de la descripción.

Con referencia a la Figura 2, se detalla ahora el microcontrolador 10.

El microcontrolador 10 tiene dos relojes internos 161 y 162.

El reloj interno 161 es del tipo HSI (“HiSpeed internal” en inglés). Está configurado para proporcionar una frecuencia de reloj de aproximadamente 16 MHz.

El reloj interno 162 es del tipo LSI (“LowSpeed Internal” en inglés). Está configurado para proporcionar una frecuencia de reloj entre aproximadamente 30 y aproximadamente 34 KHz, preferentemente 32.768 KHz.

Ventajosamente, estos relojes internos (161,162) son difíciles de atacar (en el sentido de perturbación) sin destruir el microcontrolador 10, porque se encuentran dentro de este.

El microcontrolador 10 también incluye una memoria 151 que comprende registros. Un registro es una ubicación de memoria interna de un procesador. Los registros se encuentran en la parte superior de la jerarquía de memoria, presentando un tiempo de acceso de lectura / escritura muy rápido.

El microcontrolador 10 también incluye una memoria volátil 152: no puede prescindir de la alimentación so pena de ver la información almacenada en ella, borrada irremediabilmente

La memoria 152 es preferentemente tipo SRAM (por el inglés “Static Random Access Memory”), está en particular adecuada para aplicaciones que requieren tiempos de acceso cortos y/o bajo consumo de energía.

El procesador μ del microcontrolador 10 está configurado para implementar una biblioteca criptográfica de software segura protegida contra escritura e incluye un verdadero generador de números aleatorios, así como protecciones contra ataques de canal lateral, como ataques por análisis de consumo de energía (por ejemplo, por DPA y SPA) y ataques de sincronización de relojes.

El procesador μ del microcontrolador 10 también está configurado para implementar instrucciones para la ejecución de un método de protección de software.

Preferentemente, las funcionalidades de hardware y de seguridad se agrupan en un núcleo del sistema de aplicación seguro, que está separado de las funciones no relacionadas con la seguridad, a un nivel de ejecución muy bajo utilizando las capacidades del microcontrolador 10. De hecho, dicho núcleo tiene privilegios para acceder a todos los recursos de hardware del procesador μ (en particular los GPIO), y controla una parte del procesador que gestiona la protección del acceso a la memoria (por ejemplo, MPU por Memory Protection Unit para un procesador ARM). Por lo tanto, los datos sensibles (como las claves criptográficas) son privados a un nivel seguro, y cifrado por una clave principal MK solamente conocida (es decir, accesible) del núcleo seguro.

La parte principal del método de protección del software incluye un “bucle seguro de software” ejecutado por el procesador μ para proteger la clave

principal MK. Dicho bucle permite detectar eventos de ataque a la seguridad y desencadenar el borrado de la clave principal MK en caso de tal evento, lo que dificulta enormemente el robo de datos o la clonación del terminal de pago 1.

Además, el bucle de seguridad puede hacer posible recopilar estos eventos de ataque de seguridad, para posiblemente aplicar un filtrado a estos eventos y luego activar la supresión de la clave principal MK.

La clave principal MK a proteger, responde a un estándar de encriptación, por ejemplo el estándar AES 128, AES ("Advanced Encryption Standard" en inglés) con una clave de longitud 128, 192 o 256 bits.

Preferentemente, para tener redundancia en la eliminación de la clave MK, esto es generado a partir de la combinación de por lo menos dos claves K1 y K2, por ejemplo, mediante la combinación XOR de K1 y K2. Solo las claves K1 y K2 se almacenan en la memoria, por lo que la clave MK nunca se almacena en una memoria persistente en claro. Las claves K1, K2 se pueden almacenar en zonas de memoria separadas.

Así, la clave K1 se almacena, por ejemplo, en una ubicación de memoria interna de un registro 151 del microprocesador. Ventajosamente, una ubicación de memoria de este tipo permite obtener un tiempo de acceso de lectura / escritura muy rápido.

Además, estos registros 151 se pueden restablecer mediante una entrada GPIO externo que permite un borrado casi instantáneo.

A este respecto, la clave K2 se almacena, por ejemplo, en una memoria RAM interna 152 estática o dinámica de tipo RAM, SRAM o FLASH. Opcionalmente, la clave K2 se almacena en una dirección determinada aleatoriamente.

Opcionalmente, los registros 151 y RAM interna 152 usan diferentes mecanismos de borrado, lo que permite ventajosamente incrementar la protección del software.

Con referencia también a la Figura 3, ahora se describirá un método 100 para proteger el terminal de pago 1, implementando un bucle de seguridad BS.

El método 100 permite proteger el terminal de pago 1 supervisando contra ataques físicos y ataques de software para garantizar la confidencialidad y autenticidad de los datos. Los ataques físicos considerados están vinculados, por ejemplo, a la alteración del aparato, a la temperatura, al reloj, a la baja o sobretensión, a los pulsos electromagnéticos, al barrido de frecuencia.

Después de iniciar (“**boot**”) o reiniciar después de una reiniciación (“**reset**”) del terminal de pago 1, el método 100 incluye una etapa **E0** de inicialización del método 100. En el lanzamiento, si las claves K1 y K2 ya tienen

un valor, se borran, entonces el generador aleatorio se utiliza para generar un número aleatorio que se convierte en el nuevo conjunto de claves K1, K2. El bucle de seguridad BS se puede ejecutar continuamente. Sin embargo, preferentemente, para limitar el consumo, el bucle de seguridad BS se ejecuta periódicamente, por ejemplo, cada 100 ms, por lo que el resto del tiempo el microcontrolador 10 está en estado de espera.

Por tanto, después de la etapa de inicialización E0, en una etapa **E1**, el bucle de seguridad BS es iniciado, pero puesto en estado de espera.

Una unidad de temporizador 111 permite activar periódicamente la ejecución del bucle BS.

La unidad de temporizador 111 está controlada, por ejemplo, por el reloj LSE. Dicha unidad 111 genera periódicamente interrupciones destinadas al procesador μ del microcontrolador 10. Estas interrupciones permiten detener temporalmente la ejecución normal de un programa en el fondo del terminal de pago 1, con el fin de ejecutar el bucle de seguridad BS.

Ventajosamente, para evitar que un atacante se sincronice en el bucle de seguridad BS, el valor del tiempo entre dos iteraciones del bucle es aleatorio entre aproximadamente 80 ms y aproximadamente 100 ms.

Para este propósito, la unidad de temporizador 111 usa el generador aleatorio interno al microcontrolador 10 para generar una variación aleatoria inferior a unos 20 ms.

El microcontrolador 10 asegura además que este número aleatorio no se utilice para otros fines.

Ahora se describen con más detalle los ataques físicos en los que la precedencia 100 y el terminal de pago seguro 1 tienen la responsabilidad de vigilar.

Apertura

Un ataque de sabotaje de apertura ocurre cuando se abre el terminal de pago 1 o un componente inesperado ingresa a una zona segura de dicho terminal 1. Se detecta la apertura del terminal 1 gracias a una pluralidad de interruptores (Sw1, Sw2, ..., SwN) ubicados en diferentes lugares de una caja del terminal 1. Los interruptores son implementados por ejemplo por GPIO configurados en la entrada del microcontrolador 10, elevado por resistencias internas al microcontrolador 10.

Cuando el terminal 1 está cerrado, estas señales se conectan a una fase a tierra del terminal de pago 1.

Un módulo de supervisión de apertura 121 tiene la función de detectar un estado abierto del terminal supervisando el valor de la pluralidad de señales transmitidas por los conmutadores (Sw1, Sw2, ..., SwN). Por ejemplo, en un estado sin abrir del terminal, los interruptores tienen por valor un primer valor (por ejemplo, igual a 0). Si el terminal está abierto, por lo menos uno de los interruptores, dependiendo de su ubicación, es capaz de detectar esta apertura, tomando un segundo valor (por ejemplo, igual a 1). La función del módulo de vigilancia de apertura 121 es detectar tal cambio en el valor de por lo menos un interruptor, y transmitir una señal de violación S1 al microcontrolador 10. Este último puede configurarse para incluir medios de recepción de dicha señal de violación S1, de modo que uno o varios puertos específicos, en los que la recepción de una señal a dicho microprocesador 10 significa la detección de un ataque. Dichos puertos pueden o no ser específicos de un tipo de ataque.

Perforación

Se detecta la perforación del terminal 1 gracias a por lo menos una malla de alambre. Esto se implementa, por ejemplo, mediante una malla conectada a los medios de interfaz del microcontrolador. Por ejemplo, una malla es un par (A, B) de GPIO. Por lo tanto, son necesarias por lo menos dos mallas M1, M2 para proteger el dispositivo de la perforación. GPIO A está configurado como entrada y GPIO B como salida.

Un módulo de vigilancia de perforación 122 escribe una secuencia aleatoria, por ejemplo, de 16 bits, hacia la salida (el flujo se elige de acuerdo con las limitaciones del material, de hecho, el tiempo de subida y bajada depende de la longitud de la malla). La misma secuencia debe leerse en la entrada GPIO. Luego GPIO B se define como entrada y A como salida y se aplica la misma verificación.

La secuencia aleatoria de bits aplicada se elige al azar de un conjunto de posibles secuencias (por ejemplo, por lo menos 20). La secuencia aleatoria es generada por el generador aleatorio real interno del microcontrolador 10. Además, esto asegura que la secuencia aleatoria no se utilice para otros fines.

En consecuencia, en el caso de que la secuencia recibida no se corresponda con la secuencia predefinida, el módulo de vigilancia 122 está configurado para transmitir una señal de violación S1 al microprocesador 10.

Temperatura

Se detecta un ataque de temperatura cuando la temperatura actual del terminal de pago 1 está por encima o por debajo de un rango de temperatura esperado. Por ejemplo, el rango de temperatura de funcionamiento normal es de aproximadamente [-30°C; +85°C].

El microcontrolador 10 comprende, por ejemplo, por lo menos un sensor de temperatura integrado para implementar la vigilancia de temperatura.

Un módulo de vigilancia de temperatura 123 tiene la función de medir las temperaturas, a través del (de los) sensor(es) de temperatura, de detectar un estado anormal de la temperatura si la temperatura medida no está en la zona de funcionamiento normal esperada, y en este caso transmitir una señal de violación S1 al microprocesador 10.

Frecuencia de reloj

Las funciones de vigilancia realizadas por los distintos módulos de vigilancia 121 a 123 permiten, por lo tanto, llevar a cabo una vigilancia segura para detectar una variedad de ataques físicos.

Un módulo de seguridad 120 agrupa los módulos de vigilancia 121 a 123 y las funciones de vigilancia asociadas. Dicho módulo de seguridad 120 está configurado para realizar una vigilancia segura y detectar ataques de seguridad en el terminal de pago 1 implementando el bucle de seguridad BS (que realiza las funciones del módulo de seguridad 120) descrito anteriormente.

Por lo tanto, en una etapa **E10a**, se procede a la medición de los diversos parámetros físicos a vigilar por los medios de medición del módulo de seguridad 120.

En una etapa **E10b**, se procede luego a la comparación de los valores de umbral predeterminados, correspondientes a ataques físicos al terminal 1. En el caso de que una comparación sea significativa de un ataque físico al

terminal 1, una señal de la violación S1 se transmite a un puerto de recepción del microprocesador 10

Para garantizar que el bucle de seguridad BS se ejecute sistemáticamente a pesar de los ataques, la frecuencia del reloj que impulsa la unidad de temporizador 111, el reloj LSE (Low Speed External Clock), es verificada por un contador controlado por el HSI (High Speed Internal Clock).

Para ello, en una etapa **E10c**, el módulo de seguridad 120 hace en el bucle de seguridad BS, un primer recuento del número de "golpes" (como un flanco ascendente o descendente de una señal de reloj) del reloj HSI en un ciclo del reloj LSI para verificar que la frecuencia del reloj HSI está dentro de un rango de frecuencia aceptable (por ejemplo, en el rango de funcionamiento normal ± 30 KHz). En el caso de que no se verifique esta condición, se transmite una señal de violación S1 a un puerto de recepción del microprocesador 10.

Luego, en una etapa **E10d**, el módulo de seguridad 120, hace en el bucle de seguridad BS, un segundo recuento del número de golpes del reloj LSE en un ciclo del reloj HSI para verificar que la frecuencia del reloj LSE esté dentro de un rango de frecuencia aceptable (por ejemplo, en el rango de funcionamiento normal $[30 \text{ KHz} - 34 \text{ KHz}]$). En caso de que no se verifique esta condición, se transmite una señal de violación S1 a un puerto de recepción del microprocesador 10.

El método 100 también incluye otro mecanismo para asegurar que el bucle BS se ejecute de forma regular.

Por lo tanto, el método 100 comprende un mecanismo llamado “perro guardián” implementado por un módulo 126 de vigilancia del bucle BS, configurado por ejemplo para implementar un contador que se pone regularmente a cero. Si el contador excede un valor dado (tiempo de espera), se realiza un reinicio (reset) del terminal de pago 1.

Más precisamente, el módulo 126 de vigilancia del bucle BS se inicia cuando se inicia el terminal de pago 1 y nunca puede detenerse mientras este último está en funcionamiento. Si el contador alcanza un valor preestablecido, el terminal 1 se reinicia.

Solo el bucle seguro BS “actualiza” el contador del módulo 126 de vigilancia del bucle en una etapa **E13**, de hecho, preferentemente solo el bucle seguro BS tiene acceso al contador 126. La operación de actualización se lleva a cabo solo una vez durante el ciclo de vigilancia (es decir, el perro guardián se actualiza en el proceso del bucle seguro periódico cuando todas las vigilancias se han procesado dentro de un período de tiempo determinado).

El módulo 126 de vigilancia del bucle compara el contador con un valor límite predefinido para vigilar si el bucle de seguridad BS no se ejecuta dentro del tiempo predefinido.

Por lo tanto, si surge esta situación, el contador del perro guardián alcanzará el valor límite predefinido y, en una etapa de **E50**, el módulo 126 de vigilancia del bucle transmite una señal de reinicio S2 al microprocesador 10.

Ventajosamente, este mecanismo de perro guardián asegura que todo el código relacionado con la seguridad se ejecute en el tiempo.

Preferentemente, el tiempo de expiración del perro guardián es aproximadamente igual al "Período de vigilancia periódico + 50 ms".

El contador del módulo 126 de supervisión del bucle es impulsado por el reloj LSI, que se incrementa en cada golpe del reloj (recepción de un flanco ascendente o descendente de la señal de reloj). Por lo tanto, la frecuencia del reloj LSI también se monitorea para verificar que no haya ningún ataque al terminal de pago 1 por manipulación de la frecuencia de dicho reloj LSI en una zona de frecuencia anormal, con el objetivo de provocar un mal funcionamiento del terminal de pago 1.

Así, la verificación de la frecuencia del LSI (por ejemplo, en el rango de funcionamiento normal [30 KHz - 34 KHz]) se lleva a cabo en una etapa **E30** por un módulo 131 de verificación de frecuencia, implementada, por ejemplo, por un circuito de paso de banda externo asociado con un transistor cuya salida controla la transmisión de una señal de violación S1 a un puerto de recepción del microprocesador 10.

Voltaje de alimentación

Se considera un ataque de seguridad cuando el voltaje de alimentación del microcontrolador 10 es inferior o superior a los valores de umbral.

El voltaje Vcc del microcontrolador 10 es suministrado por la fuente de alimentación principal cuando la batería está conectada y por la fuente de alimentación de emergencia cuando se retira la batería. Como el sistema normalmente siempre está encendido, la pérdida de Vcc se considera un ataque de seguridad.

El microcontrolador 10 comprende un módulo de vigilancia de bajo voltaje 141 para medir el voltaje y detectar cuando el voltaje está por debajo de un valor umbral, por ejemplo, cuando el voltaje cae por debajo de aproximadamente 2.2V.

El módulo de vigilancia de bajo voltaje 141 está implementado, por ejemplo, por un circuito de reinicio de bajo voltaje interno al microprocesador 10, estando dicho circuito configurado para transmitir una señal de reinicio S2 al microprocesador 10.

El reinicio tiene el efecto de eliminar los datos de la RAM y, por lo tanto, la eliminación de la clave K2. Preferentemente, el reinicio se detecta “instantáneamente” (es decir, se trata de un circuito solo de hardware). Para

optimizar el periodo de eliminación de la clave K2, se puede colocar en una zona de direcciones al inicio de la memoria 152.

Se considera un ataque de seguridad cuando Vcc está, por ejemplo, por encima de 3.7 V.

Un módulo de vigilancia de alta tensión 132 es responsable de la vigilancia de alta tensión, por ejemplo, en forma de un controlador externo de vigilancia de muy baja potencia. Esto se configura para, en una etapa **E40**, detectar cuando el voltaje es mayor que un valor de umbral, por ejemplo 3.7 V, y si se cumple esta condición, transmitir una señal de violación S1 tiene un puerto de recepción del microprocesador 10.

El microcontrolador 10 tiene puertos dedicados a recibir una señal de violación S1.

Así, en una etapa **E11**, al recibir una señal en uno de estos puertos, el microcontrolador 10 procede al borrado de la clave K1 almacenada en una memoria de registro 151. Este procesamiento permite tener un tiempo de borrado lo más breve posible, la ejecución del comando de eliminación no depende de una señal de reloj. Además, el comando se puede ejecutar si los distintos relojes no funcionan.

Paralelamente, también se procede borrar la zona de memoria 152 en la que se almacena la clave K2.

El microcontrolador 10 también incluye puertos dedicados a recibir una señal de reinicio S2.

Así, al recibir una señal en uno de estos puertos, el microcontrolador 10 procede a la eliminación de la clave K2 almacenada en una memoria volátil en una etapa de **Reset**.

A modo de ejemplo, el tiempo de borrado de una palabra de 32 bits en una memoria de tipo SRAM2 es de 1 ciclo de reloj (es decir, 250 ns). El tiempo de borrado SRAM2 completo es $8192 * 250 = 2.048$ ms.

Además, en una etapa **E12**, el módulo de seguridad puede registrar los tipos de alarmas y la “fecha” en la que se ve cada alarma. Ventajosamente, los datos registrados se pueden analizar cuando el terminal entra en mantenimiento después de un ataque de seguridad.

En una modalidad alternativa, el terminal de pago 1 también puede incluir un módulo de filtrado (no mostrado).

Así, en una etapa **E12'**, cuando el módulo de seguridad detecta una anomalía, el módulo de filtro está configurado para procesar la alarma y luego ejecutar un algoritmo de filtrado / detección para una o más iteraciones del bucle de seguridad BS. Dicho algoritmo permite declarar si la alarma es significativa de una situación real de pirateo del terminal para en este caso proceder al borrado de una o más claves almacenadas.

En una modalidad particular, un algoritmo de filtrado / detección es tal que al recibir una o más señales de violación S1 dentro de un primer período de tiempo determinado, el microcontrolador 10 procede a la supresión de la clave K1 en el caso de que se reciba una señal de violación S1 dentro de un segundo período determinado (por ejemplo, aproximadamente un segundo). Tal algoritmo permite realizar un filtrado sobre los eventos de origen mecánico (apertura, perforación...). Así, si se detectan varios eventos mecánicos simultáneamente y se detienen antes del tiempo determinado, la clave MK no se borra, ya que esto es por ejemplo la firma de un modelo de una caída del Terminal 1.

Para evitar que los piratas informáticos, después de haber tenido acceso a este algoritmo, utilicen este filtrado para atacar el terminal más fácilmente, el sistema de filtrado no se puede activar durante un retraso (en el que uno de los componentes es aleatorio) siguiendo una secuencia de filtrado de un mismo evento. Esto no tiene ningún efecto en el uso normal ya que, por ejemplo, en el caso de una caída, es poco probable que se caiga el mismo terminal dos veces en menos de unos pocos segundos.

Además, ventajosamente, el terminal 1 puede configurarse con modelos de ataque físico, utilizando eventos de ataque recopilados sobre varios tipos de caídas, durante la fase de diseño o prueba del terminal. Por lo tanto, este modelo se utiliza para filtrar los ataques durante la vida útil del producto.

Ventajosamente, esto permite evitar declarar un ataque mientras que la detección de un ataque corresponde solo a un parásito transitorio.

Además, los datos registrados (alarma(s) y fecha(s)) por el terminal de pago 1, o por una pluralidad de terminales de pago 1, se pueden analizar para refinar el algoritmo de filtrado / detección.