(19) RÉPUBLIQUE FRANÇAISE

INSTITUT NATIONAL DE LA PROPRIÉTÉ INDUSTRIELLE

COURBEVOIE

(11) **N° de publication :** **3 131 020**

(à n'utiliser que pour les commandes de reproduction)

(21) **N° d'enregistrement national :** **21 14271**

(51) Int Cl[8] : ***G 06 F 12/02*** *(2022.01), G 06 F 21/70, 13/14*

(12) DEMANDE DE BREVET D'INVENTION A1

(22) **Date de dépôt :** 22.12.21.

(30) **Priorité :**

(43) **Date de mise à la disposition du public de la demande :** 23.06.23 Bulletin 23/25.

(56) **Liste des documents cités dans le rapport de recherche préliminaire :** *Se reporter à la fin du présent fascicule*

(60) **Références à d'autres documents nationaux apparentés :**

( ) **Demande(s) d'extension :**

(71) **Demandeur(s) :** *ORANGE Société anonyme* — FR.

(72) **Inventeur(s) :** SANCHEZ-LEIGHTON Vicente et GONZALEZ Laurent.

(73) **Titulaire(s) :** ORANGE Société anonyme.

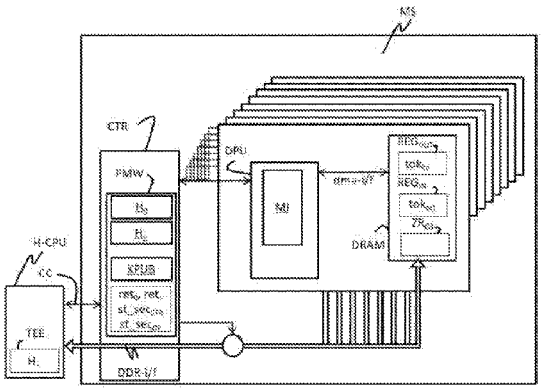
(74) **Mandataire(s) :** CABINET BEAU DE LOMENIE.

(54) Module et procédé de sécurisation d'un équipement informatique.

(57) Module et procédé de sécurisation d'un équipement informatique

Ce module de sécurisation (MS) d'un équipement informatique (OC) comporte :- au moins une mémoire RAM accessible par un processeur hôte (H-CPU) externe audit module de sécurisation (MS), via un bus mémoire ;- un contrôleur (CTR) configuré pour pouvoir être programmé par le processeur hôte (H-CPU) via un canal de contrôle (CC) ;- des unités de traitement de données (DPU) configurées pour pouvoir accéder à des zones de ladite mémoire via une interface DMA, lesdites unités de traitement de données (DPU) étant contrôlées par ledit contrôleur (CTR) ;- ledit contrôleur (CTR) étant configuré pour contrôler l'accès à ladite mémoire soit par le processeur hôte (H-CPU) soit par lesdites unités de traitement de données (DPU) ;- ledit contrôleur (CTR) étant configuré pour mettre en œuvre au moins une évaluation heuristique d'un état de sécurité (st_secCTR) de l'équipement informatique (OC) tel que perçu par ledit contrôleur (CTR), ladite évaluation heuristique comportant :(i) le chargement d'instructions dans une mémoire d'instructions (MI) de ladite unité de traitement de données (DPU) pour que celle-ci effectue au moins une opération sur au moins une partie de ladite mémoire ;(ii) l'obtention d'un résultat (ret0, ret+) de ladite opération ;(iii) la détermination dudit état de sécurité (st_secCTR) à partir dudit résultat - ledit contrôleur (CTR) étant configuré pour corrompre ladite mémoire en fonction dudit état de sécurité (st_secCTR) pour déclencher un arrêt ou un redémarrage du fonctionnement du processeur hôte (H-CPU) dudit équipement informatique (OC).

Fig. 2



FR 3 131 020 - A1

Description

Titre de l'invention : Module et procédé de sécurisation d'un équipement informatique

Arrière-plan de l'invention

[0001] La présente invention se rapporte au domaine de la sécurisation des systèmes informatiques.

[0002] Elle trouve un avantage particulier, mais non limitatif, pour la sécurisation des objets connectés (en anglais « Internet of Things », IoT).

[0003] Ces objets connectés sont aujourd'hui dépourvus de moyens de sécurisation efficaces, en particulier pour des raisons de coûts de fabrication.

[0004] Ils sont donc vulnérables.

[0005] L'invention propose une solution pour remédier à cette situation qui n'est pas satisfaisante.

Objet et résumé de l'invention

[0006] Ainsi, et selon un premier aspect, l'invention concerne un module de sécurisation d'un équipement informatique, ce module comportant :

- au moins une mémoire RAM accessible par un processeur hôte externe audit module de sécurisation, via un bus mémoire;
- un contrôleur configuré pour pouvoir être programmé par le processeur hôte via un canal de contrôle ;
- des unités de traitement de données configurées pour pouvoir accéder à des zones de ladite mémoire via une interface DMA, ces unités de traitement de données étant contrôlées par le contrôleur ;
- ledit contrôleur étant configuré pour contrôler l'accès à ladite mémoire soit par le processeur hôte soit par les unités de traitement de données ;
- ledit contrôleur étant configuré pour mettre en œuvre au moins une évaluation heuristique d'un état de sécurité de l'équipement informatique tel que perçu par ledit contrôleur, ladite évaluation heuristique comportant :

(i) le chargement d'instructions dans une mémoire d'instructions de ladite unité de traitement de données pour que celle-ci effectue au moins une opération sur au moins une partie de ladite mémoire;

(ii) l'obtention d'un résultat de ladite opération ;

(iii) la détermination dudit état de sécurité à partir dudit résultat,

- ledit contrôleur étant configuré pour corrompre ladite mémoire en fonction dudit état de sécurité pour déclencher un arrêt ou un redémarrage du fonctionnement dudit processeur hôte dudit équipement informatique.

[0007] Ainsi, et d'une façon générale, l'invention propose un module de sécurisation d'un équipement informatique, ce module présentant l'avantage de pouvoir être installé en remplacement de la mémoire RAM de l'équipement informatique à sécuriser.

[0008] Cette caractéristique est particulièrement avantageuse car le module de sécurisation peut être très facilement intégré dans l'équipement informatique à sécuriser, sans modification importante de son logiciel ni de son architecture matérielle.

[0009] Dans le mode de réalisation décrit ici, ces modules de sécurisation ont une architecture du type de celles des PIM DRAMs (PIM pour, en anglais, « processor in memory ») décrites dans le document FR3032814A1.

[0010] Les modules de sécurisation proposés par l'invention peuvent être fabriqués avec les technologies utilisées pour fabriquer des DRAM conventionnelles comme celles des PIM décrites dans le document FR3032814A1. Ils se comportent et sont vus du processeur hôte comme une DRAM classique mais ils offrent en plus une fonction de sécurisation de l'équipement informatique. En ce sens le contrôleur embarqué dans le module de sécurisation et son unité de traitement de données peuvent être vus comme un coprocesseur de sécurisation du processeur hôte.

[0011] Ce contrôleur met en œuvre une ou plusieurs évaluations heuristiques de l'état de sécurité de l'équipement informatique en analysant la mémoire utilisée par le processeur hôte. Comme le contrôleur n'a pas lui-même directement accès à la mémoire, il s'appuie, notamment pour toutes les opérations d'accès à la mémoire, sur les unités de traitement de données, chacune d'entre elles pouvant accéder, sous le contrôle du contrôleur, à une partie de mémoire partagée avec le processeur hôte.

[0012] Les heuristiques utilisées pour ces évaluations peuvent être de différentes natures.

[0013] Dans un mode particulier de réalisation, une heuristique détermine que l'état de sécurité de l'équipement informatique, tel que perçu par ledit contrôleur, représente un risque important de dysfonctionnement dudit équipement informatique, par exemple si une zone mémoire normalement destinée à recevoir une constante du système d'exploitation a été modifiée ou accédée selon un schéma considéré suspect.

[0014] Par exemple, un schéma d'accès à une zone mémoire peut être considéré suspect en fonction d'un nombre d'accès, d'une fréquence d'accès, ou d'une variation d'un de ces paramètres.

[0015] L'homme du métier sait en particulier que les systèmes d'exploitation habituels (Linux, Windows, ...), ainsi que la plupart des logiciels obtenus par compilation depuis un langage évolué notamment, ont pour habitude d'écrire des données constantes dans certaines zones réservées de la mémoire. Une heuristique peut surveiller ces zones et supposer un état de dysfonctionnement du dispositif informatique dès lors que le contenu de ces zones réservées est modifié ou exploité de manière inhabituelle.

[0016] Dans un mode particulier de réalisation, une heuristique utilise une méthode d'apprentissage, par exemple un réseau de neurones pour déterminer, à partir d'une signature du contenu d'au moins une partie d'au moins une mémoire calculée par au moins dite unité de traitement de données, une probabilité que ledit équipement informatique soit dans un état de dysfonctionnement, l'état de sécurité de l'équipement informatique tel que perçu par ledit contrôleur étant déterminé à partir de cette probabilité.

[0017] Ce réseau de neurones peut préalablement être entraîné en mode supervisé avec des signatures de contenus de mémoires d'équipements à différents niveaux de dysfonctionnement.

[0018] Mais l'évaluation ou la détermination d'un état de sécurité de sécurité de l'équipement informatique ne suffit pas à le sécuriser.

[0019] De façon remarquable, l'invention propose, pour déclencher l'arrêt ou le redémarrage du fonctionnement de l'équipement informatique de corrompre au moins une partie de sa mémoire pour placer l'équipement dans un état imprévisible du point de vue du processeur hôte, de sorte à déclencher des actions sécuritaires du processeur hôte.

[0020] Dans un mode particulier de réalisation, pour corrompre une mémoire, le contrôleur est configuré bloquer l'accès à ladite mémoire par ledit processeur hôte.

[0021] Dans un mode particulier de réalisation, pour corrompre une dite mémoire, ledit contrôleur est configuré pour charger des instructions dans une mémoire d'instructions d'une dite unité de traitement de données pour que celle-ci enregistre une donnée dans une zone de ladite mémoire réservée à un autre usage.

[0022] Lorsque le processeur hôte est confronté à l'une ou l'autre situation (accès impossible à la mémoire ou mémoire corrompue), il peut déclencher une action de sécurité consistant à arrêter le fonctionnement du système et à le redémarrer dans un état sain.

[0023] L'invention vise également un procédé de sécurisation d'un équipement informatique, ce procédé étant mis en œuvre par un contrôleur embarqué dans un module de sécurisation, ledit contrôleur étant configuré pour contrôler l'accès à une zone de mémoire RAM soit par un processeur hôte via un bus mémoire, soit par une unité de traitement de données via une interface DMA, ledit procédé comportant :

- la mise en œuvre d'au moins une évaluation heuristique d'un état de sécurité de l'équipement informatique, ladite évaluation heuristique comportant :

(i) le chargement d'instructions dans une mémoire d'instructions de ladite unité de traitement de données pour que celle-ci effectue au moins une opération sur au moins une partie de ladite mémoire;

(ii) l'obtention d'un résultat de ladite opération ;

(iii) la détermination dudit état de sécurité à partir dudit résultat

- la corruption de ladite mémoire en fonction dudit état de sécurité pour déclencher un arrêt ou un redémarrage du fonctionnement du processeur hôte de l'équipement informatique.

[0024] L'invention vise aussi un équipement informatique, par exemple un objet connecté, comportant un processeur hôte et au moins un module de sécurisation tel que mentionné ci-dessus.

[0025] Dans un mode particulier de réalisation, une heuristique est enregistrée de façon certifiée dans une mémoire non volatile dudit contrôleur.

[0026] Ce mode de réalisation permet de prévoir une heuristique dite de base qui sera systématiquement exécutée. Elle peut par exemple être préenregistrée dans le micro-logiciel ou firmware du contrôleur de sorte que le module de sécurisation incorpore nativement une solution complète de sécurisation (surveillance de la mémoire, déclenchement d'une action sécuritaire).

[0027] Dans un mode particulier de réalisation, le contrôleur est configuré pour recevoir une heuristique du processeur hôte via le canal de contrôle.

[0028] Dans un mode particulier de réalisation dans lequel le processeur hôte comporte un environnement d'exécution sécurisé, cette heuristique peut être envoyée par une application certifiée de cet environnement d'exécution sécurisé.

[0029] Une signature de cette heuristique effectuée par le processeur hôte ou par une application certifiée de l'environnement sécurisée peut être vérifiée par le contrôleur avant utilisation de cette heuristique.

[0030] Cette heuristique peut être préférentiellement exécutée par le contrôleur en complément de l'heuristique de base.

[0031] Dans les modes de réalisation de l'invention mentionnés ci-dessus, le contrôleur met en œuvre au moins une heuristique, pour déterminer un état de sécurité de l'équipement informatique tel que perçu par ce contrôleur.

[0032] Dans un mode particulier de réalisation, le contrôleur prend aussi en considération un état de sécurité de l'objet informatique tel que perçu par un tiers de confiance.

[0033] Ainsi, dans un mode particulier de réalisation, le module de sécurisation est caractérisé en ce que le contrôleur est configuré pour charger des instructions dans une mémoire d'instructions d'une dite unité de traitement de données pour que celle-ci :

- obtienne un jeton dans un registre de la mémoire accessible par cette unité de traitement de données ;
- déchiffre ledit jeton avec une clé publique du tiers de confiance pour obtenir un état de sécurité de l'équipement informatique tel que perçu par ce tiers de confiance ;
- retourne cet état de sécurité audit contrôleur ;
- ledit contrôleur étant configuré pour corrompre ladite mémoire en fonction :

(i) de l'état de sécurité de l'objet informatique tel que perçu par le contrôleur ; et

(ii) de l'état de sécurité de l'objet informatique tel que perçu par le tiers de confiance.

[0034] Ces jetons envoyés par le tiers de confiance permettent au contrôleur d'obtenir de l'information sur l'environnement de l'équipement informatique. Par exemple, si le contrôleur cesse de recevoir des jetons du tiers de confiance, il peut en déduire un dysfonctionnement de l'équipement informatique et décider de corrompre sa mémoire pour déclencher l'arrêt et éventuellement le redémarrage de son fonctionnement.

[0035] Dans un mode particulier de réalisation, le contrôleur est configuré pour charger des d'instructions dans une mémoire d'instructions d'une unité de traitement de données pour que celle-ci :

- obtienne un jeton par chiffrement de l'état de sécurité vu du contrôleur avec une clef publique du tiers de confiance ; et
- enregistre ce jeton dans un registre de la mémoire accessible par cette unité de traitement de données.

[0036] Ce mode de réalisation de l'invention permet de remonter au tiers de confiance l'état de sécurité de l'objet connecté vu du contrôleur.

[0037] Dans un mode de réalisation, la communication entre le tiers de confiance et l'équipement informatique est assurée par un processus applicatif exécuté par le processeur hôte.

[0038] Dans ce mode de réalisation, le processeur hôte est configuré pour exécuter un processus applicatif comportant des instructions pour :

- enregistrer les jetons reçus du tiers de confiance dans un registre d'une dite mémoire, ces jetons étant les états de sécurité de l'équipement informatique perçus par le tiers de confiance et signés par la clé privée dudit tiers de confiance; et pour
- envoyer au tiers de confiance, les jetons lus dans un registre de la mémoire et correspondants à des états de sécurité de l'équipement informatique perçus par ledit contrôleur et signés par la clé publique dudit tiers de confiance.

[0039] Dans un mode particulier de réalisation, les différentes étapes du procédé de sécurisation sont déterminées par des instructions de programmes d'ordinateurs ou sont implémentées par une puce en silicium qui comprend des transistors adaptés pour constituer des portes logiques d'une logique câblée non programmable.

[0040] En conséquence, l'invention vise aussi un programme d'ordinateur sur un support d'informations, ce programme étant susceptible d'être mis en œuvre dans contrôleur, ce programme comportant des instructions adaptées à la mise en œuvre des étapes d'un procédé de gestion d'urgence tel que décrit ci-dessus.

[0041] Ce programme peut utiliser n'importe quel langage de programmation, et être sous la forme de code source, code objet, ou de code intermédiaire entre code source et code objet, tel que dans une forme partiellement compilée, ou dans n'importe quelle autre forme souhaitable.

[0042] L'invention vise aussi un support d'informations lisible par un ordinateur, et comportant des instructions d'un programme d'ordinateur tel que mentionné ci-dessus. Le support d'informations peut être n'importe quelle entité ou dispositif capable de stocker le programme. Par exemple, le support peut comporter un moyen de stockage, tel qu'une ROM, une mémoire non volatile de type flash ou encore un moyen d'enregistrement magnétique, par exemple un disque dur. D'autre part, le support d'informations peut être un support transmissible tel qu'un signal électrique ou optique, qui peut être acheminé via un câble électrique ou optique, par radio ou par d'autres moyens. Le programme selon l'invention peut être en particulier téléchargé sur un réseau de type Internet. Alternativement, le support d'informations peut être un circuit intégré dans lequel le programme est incorporé, le circuit étant adapté pour exécuter ou pour être utilisé dans l'exécution du procédé en question.

Brève description des dessins

[0043] D'autres caractéristiques et avantages de la présente invention ressortiront de la description faite ci-dessous, en référence aux dessins annexés qui en illustrent des exemples de réalisation dépourvus de tout caractère limitatif. Sur les figures :

[0044] [Fig.1] La [Fig.1] représente de façon schématique un équipement informatique conforme à un mode particulier de réalisation de l'invention;

[0045] [Fig.2] La [Fig.2] représente un processeur hôte et un module de sécurisation dans un mode particulier de mise en œuvre de l'invention ;

[0046] [Fig.3] La [Fig.3] représente un exemple d'environnement logiciel du processeur hôte dans un mode particulier de mise en œuvre de l'invention ;

[0047] [Fig.4] La [Fig.4] représente, sous forme d'organigramme, les principales étapes d'un procédé de sécurisation conforme à un mode particulier de réalisation de l'invention ; et

[0048] [Fig.5] La [Fig.5] représente un contrôleur pouvant être utilisé dans un mode particulier de mise en œuvre de l'invention.

Description des modes de réalisation

[0049] La [Fig.1] représente schématiquement un équipement informatique OC conforme à un mode particulier de mise en œuvre de l'invention, par exemple un objet connecté.

[0050] Cet équipement informatique OC comporte un processeur hôte H-CPU configuré pour accéder, via un bus mémoire, à des mémoires de type DRAM (plus simplement appelées DRAMs).

[0051] Dans le mode de réalisation décrit ici, le processeur hôte H-CPU accède aux DRAMs en utilisant un protocole DDR.

[0052] Le terme « protocole DDR » (en anglais Double Data Rate) est utilisé ici pour désigner un protocole quelconque parmi DDR1, DDR2, DDR3, DDR4, RLDRAM

(mémoire à accès aléatoire à latence réduite), RLDRAM2, et tout protocole similaire à ces protocoles.

[0053] Dans le mode de réalisation toutes les mémoires DRAM accessibles par le processeur hôte H-CPU sont intégrées dans des modules de sécurisation MS conformes à l'invention.

[0054] Dans le mode de réalisation décrit ici, ces modules de sécurisation MS ont une architecture du type de celles des PIM DRAMs (PIM pour, en anglais, « processor in memory ») décrites dans le document FR3032814A1.

[0055] En variante, le processeur hôte H-CPU pourrait en outre accéder à une ou plusieurs DRAMs conventionnelles globalement référencées C-MEM et représentée en pointillés sur la [Fig.1].

[0056] Dans le mode particulier de réalisation décrit ici, les modules de sécurisation MS sont organisés en barrettes BAR, par exemple au format DIMM (an anglais Dual Inline Memory Module).

[0057] Selon le mode de réalisation de l'invention, le processeur hôte H-CPU peut supporter un ou plusieurs canaux de mémoire DDR (4 sont représentés dans l'exemple de la [Fig.1]), chaque canal de mémoire pouvant prendre en charge une ou plusieurs barrettes BAR de modules de sécurisation MS (2 dans l'exemple de la [Fig.1]).

[0058] Dans le mode de réalisation décrit ici, chaque barrette BAR comporte un ou plusieurs modules de sécurisation MS, par exemple 8.

[0059] La [Fig.2] représente schématiquement le processeur hôte H-CPU et un module de sécurisation MS dans un mode particulier de réalisation de l'invention.

[0060] Dans ce mode de réalisation, le module de sécurisation MS comporte :

- une interface DDR référencée DDR-i/f ;
- un contrôleur CTR configuré pour recevoir des commandes du processeur hôte H-CPU via un canal de contrôle CC; et
- une pluralité (8 dans l'exemple de la [Fig.2]) d'unités de traitement de données, ci-après DPU (en anglais Data Processing Unit), chaque DPU étant configuré pour accéder à une mémoire DRAM via une interface DMA (en anglais Direct Memory Access) référencée dma-i/f.

[0061] Dans le mode de réalisation décrit ici, l'interface DDR-i/f est commune à toutes les DRAMs, de sorte que ces mémoires DRAM sont également accessibles par le processeur hôte H-CPU via l'interface DDR-i/f.

[0062] Une quelconque de ces mémoires DRAM est donc partagée entre le processeur hôte H-CPU (via l'interface DDR) et le DPU (via l'interface DMA). Le contrôleur CTR gère le contrôle à ces mémoires DRAM de manière à empêcher un accès simultané à une plage de mémoire à la fois par le processeur hôte H-CPU et par le DPU associé à cette mémoire.

[0063] Le contrôleur CTR n'a pas d'accès direct aux mémoires DRAM mais il est configuré pour pouvoir demander à un DPUs d'exécuter des instructions pour lire ou pour écrire dans la DRAM associé à ce DPU.

[0064] Le contrôleur CTR peut également demander à un DPU d'effectuer d'autres calculs, notamment des calculs de chiffrement et de déchiffrement des jetons mentionnés précédemment.

[0065] A cet effet, chaque DPU comporte une mémoire d'instructions MI, de taille relativement réduite, typiquement 24ko.

[0066] Le seul environnement connu d'un DPU est sa propre DRAM. Il opère uniquement sur ordre du contrôleur CTR.

[0067] Conformément à l'invention, le contrôleur CTR est configuré pour s'appuyer sur les DPUs afin de sécuriser l'exécution de l'équipement informatique OC. En ce sens, le contrôleur CTR peut être considéré comme un coprocesseur de sécurité du processeur hôte H-CPU.

[0068] Dans le mode de réalisation décrit ici, le contrôleur CTR comporte un logiciel embarqué FMW (en anglais firmware) avec une mémoire dédiée à ce contrôleur.

[0069] Dans le mode de réalisation décrit ici, ce firmware FMW comporte une heuristique H_0 de base pour permettre au contrôleur CTR d'évaluer un état de sécurité st_sec_{CTR} de l'équipement informatique OC, tel que perçu par le contrôleur, sur la base des variations du contenu d'au moins une partie d'au moins une DRAM.

[0070] Dans le mode de réalisation décrit ici, le contrôleur CTR est configuré pour recevoir du processeur hôte H-CPU au moins une heuristique H_+, complémentaire de l'heuristique de base H_0, pour lui permettre d'affiner cet état de sécurité st_sec_{CTR} sur la base des variations du contenu d'au moins une partie d'au moins une DRAM. Cette heuristique complémentaire H_+ est par exemple reçue d'un environnement d'exécution sécurisé TEE du processeur hôte H-CPU (en anglais Trusted Execution Environment).

[0071] Le contrôleur CTR n'ayant pas accès direct aux DRAMs, il utilise, pour mettre en œuvre l'une ou l'autre de ces heuristiques H_0, H_+, les services des DPU ayant accès à ces DRAMs.

[0072] Plus précisément, pour mettre en œuvre une heuristique $H_{i,\, i=0,+,}$ le contrôleur CTR est configuré pour charger dans la mémoire d'instructions MI d'un DPU, par blocs de 24 ko maximum dans cet exemple, les instructions de cette heuristique H_i pour que ce DPU les exécute et lui retourne un résultat ret_i de cette exécution.

[0073] Ces instructions peuvent être de différentes natures. Elles peuvent être des instructions pour que le DPU retourne simplement au contrôleur CTR le contenu d'une partie de la DRAM accessible par ce DPU ou une valeur calculée à partir d'une partie du contenu de la DRAM accessible par ce DPU.

[0074] Le contrôleur CTR est configuré pour évaluer un état de sécurité st_sec_{CTR} de

l'équipement informatique OC, sur la base d'une ou plusieurs valeurs ret_i retournées par un ou plusieurs DPU. On dira que cet état de sécurité est un état de sécurité tel que perçu par le contrôleur CTR.

[0075] Dans un mode de réalisation, le contrôleur CTR est configuré pour demander à un DPU de chiffrer l'état de sécurité st_sec_{CTR} du dispositif informatique OC, tel que perçu par le contrôleur CTR, avec une clé publique KPUB d'un tiers de confiance OP pour générer un jeton tok_{int}, et pour demander à un DPU d'enregistrer ce jeton tok_{int} dans un registre REG_{OUT} de sa DRAM.

[0076] Dans un mode de réalisation, le contrôleur CTR est configuré pour demander à un DPU de déchiffrer un jeton tok_{ext} compris dans un registre REG_{IN} de sa DRAM avec cette clé publique KPUB et de lui fournir le résultat st_sec_{OP} de ce déchiffrement, ce résultat étant représentatif d'un état de sécurité de l'équipement informatique OC tel que perçu par ce tiers de confiance OP.

[0077] Dans un mode de réalisation de l'invention, le contrôleur CTR est configuré pour décider à partir de l'état de sécurité st_sec_{CTR} de l'équipement informatique OC tel que perçu par le contrôleur CTR et/ou à partir de l'état de sécurité st_sec_{OP} de l'équipement informatique OC tel perçu par le tiers de confiance OP, s'il doit mettre en œuvre une action ACT de corruption d'au moins une DRAM pour entraîner un arrêt ou un re-démarrage du fonctionnement de l'équipement informatique OC.

[0078] Dans un mode de réalisation de l'invention, le contrôleur CTR est configuré pour corrompre par lui-même une DRAM en bloquant les accès à cette DRAM par le processeur hôte H-CPU.

[0079] Dans un mode de réalisation de l'invention, le contrôleur CTR est configuré pour demander à un DPU d'écrire dans une zone déterminée ZR_{OS} de sa DRAM, par exemple une zone réservée à la mémorisation de constantes du système d'exploitation OS, de façon à corrompre cette DRAM pour entraîner un arrêt ou un redémarrage du fonctionnement de l'équipement informatique OC.

[0080] La [Fig.3] représente schématiquement un exemple de l'environnement logiciel du processeur hôte H-CPU dans un mode particulier de mise en œuvre de l'invention.

[0081] Sur cette figure on a représenté un processus applicatif PA. Dans le mode de réalisation décrit ici, ce processus applicatif est configuré pour communiquer avec un opérateur OP d'un réseau auquel est connecté l'équipement informatique OC, cet opérateur constituant un tiers de confiance au sens de l'invention.

[0082] Dans le mode de réalisation décrit ici, l'équipement informatique OC comporte un système d'exploitation OS, par exemple de type Linux, comportant notamment un module M-CONFIG de configuration et de gestion de la mémoire DRAM accessible par le processeur hôte H-CPU.

[0083] Dans le mode de réalisation décrit ici, l'environnement du processeur hôte H-CPU

comporte un environnement d'exécution sécurisé TEE.

[0084] Dans le mode de réalisation décrit ici, le tiers de confiance OP est configuré pour envoyer au processus applicatif PA, par exemple régulièrement, un jeton tok_{ext} comportant un état de sécurité st_sec_{OP} de l'équipement informatique OC tel que perçu par le tiers de confiance OP, ce jeton étant signé avec une clé privée KPRIV du tiers de confiance OP associée à la clé publique KPUB enregistrée dans une mémoire non volatile du contrôleur CTR du module de sécurisation MS.

[0085] Dans le mode de réalisation décrit ici, le processus applicatif PA enregistre ce jeton tok_{ext} dans un registre REG_{IN} de la DRAM, cette écriture se faisant par le processeur hôte H-CPU via l'interface DDR.

[0086] Dans le mode de réalisation décrit ici, le processus applicatif PA est configuré pour lire, via l'interface DDR, par exemple régulièrement, le jeton tok_{int} chiffré par la clé publique du tiers de confiance OP contenu dans le registre REG_{OUT} de la DRAM et pour envoyer ce jeton au tiers de confiance OP. Le tiers de confiance OP est configuré pour déchiffrer ce jeton tok_{int} avec sa clé privée KPRIV de sorte à obtenir l'état de sécurité st_sec_{CTR} du dispositif informatique OC perçu par le contrôleur CTR.

[0087] Le tiers de confiance OP peut utiliser cette information st_sec_{CTR} pour modifier sa perception st_sec_{OP} de l'état de sécurité de l'équipement informatique OC, ce nouvel état envoyé sous forme d'un jeton signé tok_{ext} au processus applicatif PA.

[0088] Dans le mode de réalisation décrit ici, l'environnement d'exécution sécurisé TEE du processeur hôte H-CPU offre un service SBOOT pour fournir, au moment du démarrage de l'équipement informatique OC, l'heuristique H_+ au contrôleur CTR d'au moins un module de sécurisation MS.

[0089] La [Fig.4] représente les principales étapes d'un procédé de sécurisation mis en œuvre par le contrôleur CTR du dispositif de sécurisation MS.

[0090] Au cours d'une étape E10 du démarrage de l'objet connecté (procédure de boot), le contrôleur CTR est configuré par le processeur hôte H-CPU via le canal de contrôle CC.

[0091] Dans le mode de réalisation décrit ici, le contrôleur CTR reçoit de l'environnement d'exécution sécurisé, au cours d'une étape E20, l'heuristique H_+ d'analyse complémentaire de la DRAM. Il l'enregistre dans une mémoire non volatile.

[0092] Dans le mode de réalisation décrit ici, le contrôleur CTR exécute ensuite une boucle comportant :

- une étape E30 d'exécution de l'heuristique de base H_0 installée dans le firmware FMW à partir du contenu d'au moins une partie d'au moins une DRAM, d'une évolution de ce contenu, ou d'un schéma d'accès à ce contenu. Cette étape comporte le chargement par le contrôleur CTR d'instruction de cette heuristique dans la mémoire d'instructions MI du DPU associé à cette DRAM pour que le DPU analyse cette

DRAM et retourne au contrôleur CTR le résultat ret_0 de cette analyse ;

- une étape optionnelle E40 d'exécution de l'heuristique complémentaire H_+, cette étape étant similaire à l'étape E30 ;
- une étape E50 d'obtention de l'état st_sec_{CTR} de sécurité de l'objet connecté vu du contrôleur CTR à partir des résultats ret_0 et ret_+ reçus des DPUs pour l'exécution des heuristiques H_0 et H_+ ;
- une étape E60 de contrôle d'au moins un DPU pour qu'il chiffre l'état st_sec_{CTR} avec la clé publique KPUB du tiers de confiance OP pour obtenir un jeton tok_{int} et pour qu'il enregistre ce jeton tok_{int} dans le registre REG_{OUT} de sa DRAM;
- une étape E70 de contrôle d'au moins un DPU pour qu'il vérifie la signature, à l'aide de la clé publique KPUB, du jeton tok_{ext} contenu dans le registre REG_{IN} de la DRAM et qu'il lui communique le résultat de cette vérification, celui-ci correspondant au dernier état de sécurité st_sec_{OP} de l'équipement informatique OC tel que perçu par le tiers de confiance OP ;
- une étape E80 de décision de mettre en œuvre ou non une action de corruption d'au moins une DRAM pour entraîner un arrêt ou un redémarrage du fonctionnement de l'équipement informatique OC en fonction des états de sécurité st_sec_{CTR}, st_sec_{OP} de l'objet connecté OC vus respectivement du contrôleur CTR et du tiers de confiance OP ; et
- une étape E90 de mise en œuvre d'au moins une action de corruption d'au moins une DRAM en fonction du résultat de l'étape E80 de décision.

[0093] Les heuristiques H_0 et H_+ d'analyse d'au moins une partie de la DRAM peuvent être de différentes natures. Elles visent d'une façon générale à obtenir un état de sécurité st_sec_{CTR} de l'équipement informatique OC tel que perçu par le contrôleur CTR au vu du contenu d'au moins une partie d'au moins une DRAM.

[0094] Par exemple, l'heuristique de base H_0 considère que l'état de sécurité st_sec_{CTR} de l'équipement informatique doit représenter un risque important de dysfonctionnement (sécurité faible) si une zone mémoire d'une DRAM normalement destinée à recevoir une constante du système d'exploitation OS a été modifiée.

[0095] Par exemple, l'heuristique complémentaire H_+ utilise une méthode d'apprentissage, par exemple de régression statistique, pour déterminer, à partir d'une signature du contenu d'au moins une partie d'au moins une DRAM calculée par au moins un DPU, une probabilité que cet équipement informatique OC soit dans un état de dysfonctionnement, l'état de sécurité st_sec_{CTR} de l'équipement informatique OC, vu du contrôleur CTR étant déterminé à partir de cette probabilité.

[0096] Dans le mode de réalisation décrit ici, à l'étape E80, le contrôleur CTR détermine à partir:

- de l'état de sécurité st_sec_{CTR} de l'équipement informatique OC tel que perçu par le

contrôleur CTR ; et

- de l'état de sécurité st_sec_{OP} de l'équipement informatique OC tel que perçu par le tiers de confiance OP,

s'il convient de corrompre au moins une DRAM pour entraîner un arrêt et éventuellement un redémarrage du fonctionnement de l'équipement informatique OC.

[0097] Dans un mode de réalisation, le contrôleur CTR donne plus de poids (80/20) à l'état de sécurité st_sec_{CTR} de l'équipement informatique OC tel que perçu par le contrôleur CTR il déclenche une telle action si $0.8.\ st\_sec_{CTR} + 0.2.\ st\_sec_{OP}$ est supérieur à un seuil prédéterminé.

[0098] Dans un autre mode de réalisation, le contrôleur CTR utilise la logique suivante :

- si l'état de sécurité st_sec_{CTR} est supérieur à un premier seuil (probabilité de dysfonctionnement important selon la perception du contrôleur CTR),

- attendre la réception de n jetons tok_{ext} pour laisser au tiers de confiance OP le temps d'envoyer, sous forme d'un jeton tok_{ext}, un état de sécurité st_sec_{OP} requérant un arrêt et éventuellement un redémarrage d'urgence,

- si ces n jetons ne sont pas reçus pendant une durée prédéterminée, ou si un de ces jetons porteur d'une demande d'arrêt d'urgence reçu, alors déclencher l'action de corruption d'au moins une DRAM pour entraîner un arrêt du fonctionnement de l'équipement informatique OC.

[0099] Si le contrôleur CTR décide de mettre en œuvre une ou plusieurs actions pour déclencher l'arrêt de l'objet connecté OC, celle-ci est mise en œuvre au cours d'une étape E90.

[0100] Cette étape E90 peut notamment consister à :

- empêcher l'accès à une mémoire DRAM par le processeur hôte H-CPU, ou

- à demander à un DPU décrire dans une zone réservée ZR_{OS} de la mémoire DRAM, par exemple à une zone dans laquelle le système d'exploitation OS stocke normalement des constantes.

[0101] La [Fig.5] représente l'architecture matérielle d'un contrôleur CTR pouvant être utilisé dans un module de sécurisation MS conforme à l'invention. Dans le mode de réalisation décrit ici, le contrôleur CTR comprend notamment un processeur 10, une mémoire non volatile 11, une mémoire vive 12, une interface DDR, un port P de communication avec un processeur hôte H-CPU et une interface de contrôle des DPUs.

[0102] La mémoire non volatile 11 constitue un support d'enregistrement conforme à l'invention, lisible par le processeur 10 et sur lequel est enregistré un programme d'ordinateur PG conforme à l'invention, ce programme comportant des instructions pour l'exécution des étapes d'un procédé de sécurisation selon l'invention dont les principales ont été décrites en référence à la [Fig.4] dans un mode de réalisation.

[0103] La mémoire non volatile 11 peut être utilisée pour mémoriser la clé publique KPUB

du tiers de confiance OP, le firmware FMW et les heuristiques.

Revendications

[Revendication 1] Module de sécurisation (MS) d'un équipement informatique (OC), ce module comportant :
- au moins une mémoire RAM accessible par un processeur hôte (H-CPU) externe audit module de sécurisation (MS), via un bus mémoire ;
- un contrôleur (CTR) configuré pour pouvoir être programmé par le processeur hôte (H-CPU) via un canal de contrôle (CC) ;
- des unités de traitement de données (DPU) configurées pour pouvoir accéder à des zones de ladite mémoire via une interface DMA, lesdites unités de traitement de données (DPU) étant contrôlées par ledit contrôleur (CTR);
- ledit contrôleur (CTR) étant configuré pour contrôler l'accès à ladite mémoire soit par le processeur hôte (H-CPU) soit par lesdites unités de traitement de données (DPU) ;
- ledit contrôleur (CTR) étant configuré pour mettre en œuvre au moins une évaluation heuristique d'un état de sécurité (st_sec_{CTR}) de l'équipement informatique (OC) tel que perçu par ledit contrôleur (CTR), ladite évaluation heuristique comportant :
(i) le chargement d'instructions dans une mémoire d'instructions (MI) de ladite unité de traitement de données (DPU) pour que celle-ci effectue au moins une opération sur au moins une partie de ladite mémoire ;
(ii) l'obtention d'un résultat (ret_0, ret_+) de ladite opération ;
(iii) la détermination dudit état de sécurité (st_sec_{CTR}) à partir dudit résultat
- ledit contrôleur (CTR) étant configuré pour corrompre ladite mémoire en fonction dudit état de sécurité (st_sec_{CTR}) pour déclencher un arrêt ou un redémarrage du fonctionnement dudit processeur hôte (H-CPU) dudit équipement informatique (OC).

[Revendication 2] Module de sécurisation (MS) selon la revendication 1, caractérisé en ce que ladite évaluation heuristique comporte une heuristique (H_0) enregistrée de façon certifiée dans une mémoire non volatile (11) dudit contrôleur (CTR).

[Revendication 3] Module de sécurisation (MS) selon la revendication 1 ou 2, caractérisé en ce que ledit contrôleur (CTR) est configuré pour recevoir une dite heuristique (H_+) dudit processeur hôte (H-CPU) via ledit canal de

contrôle (CC), ladite heuristique (H_+) étant utilisée pour ladite au moins une évaluation heuristique.

[Revendication 4] Module de sécurisation (MS) selon l'une quelconque des revendications 1 à 3, caractérisé en ce que ledit contrôleur (CTR) est configuré pour charger des d'instructions dans une mémoire d'instructions (MI) d'une dite unité de traitement de données (DPU) pour que celle-ci :
- obtienne un jeton (tok_{ext}) dans un registre (REG_{IN}) de la mémoire accessible par cette unité de traitement de données (DPU) ;
- vérifie une signature dudit jeton (tok_{ext}) avec une clé publique d'un tiers de confiance (OP) pour obtenir un état de sécurité (st_sec_{OP}) de l'équipement informatique (OC) tel que perçu par ledit tiers de confiance (OP) ;
- retourne cet état de sécurité (st_sec_{OP}) audit contrôleur (CTR) ;
- ledit contrôleur (CTR) étant configuré pour corrompre ladite mémoire en fonction :
(i) de l'état de sécurité (st_sec_{CTR}) de l'objet informatique tel que perçu par le contrôleur (CTR) ; et
(ii) de l'état de sécurité (st_sec_{OP}) de l'objet informatique tel que perçu par le tiers de confiance.

[Revendication 5] Module de sécurisation (MS) selon l'une quelconque des revendications 1 à 4, caractérisé en ce que ledit contrôleur (CTR) est configuré pour charger des d'instructions dans une mémoire d'instructions (MI) d'une dite unité de traitement de données (DPU) pour que celle-ci :
- obtienne un jeton (tok_{int}) par chiffrement dudit état de sécurité (st_sec_{CTR}) tel que perçu par ledit contrôleur (CTR) avec une clef publique (KPUB) d'un tiers de confiance (OP) ; et
- enregistre ledit jeton (tok_{int}) dans un registre (REG_{OUT}) de la mémoire accessible par cette unité de traitement de données (DPU).

[Revendication 6] Module de sécurisation (MS) selon l'une quelconque des revendications 1 à 5, caractérisé en ce que, pour corrompre une dite mémoire, ledit contrôleur (CTR) est configuré pour bloquer l'accès à ladite mémoire par ledit processeur hôte.

[Revendication 7] Module de sécurisation (MS) selon l'une quelconque des revendications 1 à 6, caractérisé en ce que, pour corrompre une dite mémoire, ledit contrôleur (CTR) est configuré pour charger des d'instructions dans une mémoire d'instructions (MI) d'une dite unité de traitement de données (DPU) pour que celle-ci enregistre une donnée dans une zone (ZR_{OS}) de ladite mémoire réservée à un autre usage.

[Revendication 8] Module de sécurisation (MS) selon l'une quelconque des revendications 1 à 7, caractérisé en ce que ladite au moins une heuristique détermine que l'état de sécurité (st_sec_{CTR}) de l'équipement informatique (OC) tel que perçu par ledit contrôleur (CTR), doit représenter un risque important de dysfonctionnement dudit équipement informatique si une zone d'une dite mémoire normalement destinée à recevoir une constante du système d'exploitation (OS) a été modifiée ou accédée selon un schéma considéré suspect.

[Revendication 9] Module de sécurisation (MS) selon l'une quelconque des revendications 1 à 8, caractérisé en ce que ladite au moins une heuristique utilise une méthode d'apprentissage pour déterminer, à partir d'une signature du contenu d'au moins une partie d'au moins une mémoire calculée par au moins une dite unité de traitement de données (DPU), une probabilité que ledit équipement informatique (OC) soit dans un état de dysfonctionnement, l'état de sécurité (st_sec_{CTR}) de l'équipement informatique (OC) tel que perçu par ledit contrôleur (CTR) étant déterminé à partir de cette probabilité.

[Revendication 10] Procédé de sécurisation (MS) d'un équipement informatique (OC), ce procédé étant mis en œuvre par un contrôleur (CTR) embarqué dans un module de sécurisation (MS), ledit contrôleur étant configuré pour contrôler l'accès à une zone de mémoire RAM soit par un processeur hôte (H-CPU) via un bus mémoire, soit par une unité de traitement de données (DPU) via une interface DMA, ledit précédé comportant :
- la mise en œuvre d'au moins une évaluation heuristique d'un état de sécurité (st_sec_{CTR}) de l'équipement informatique (OC), ladite évaluation heuristique (H_0, H_+) comportant :
(i) le chargement (E30, E40) d'instructions dans une mémoire d'instructions (MI) de ladite unité de traitement de données (DPU) pour que celle-ci effectue au moins une opération sur au moins une partie de ladite mémoire ;
(ii) l'obtention (E30, E40) d'un résultat (ret_0, ret_+) de ladite opération ;
(iii) la détermination (E50) dudit état de sécurité (st_sec_{CTR}) à partir dudit résultat
- la corruption (E80) de ladite mémoire en fonction dudit état de sécurité (st_sec_{CTR}) pour déclencher un arrêt ou un redémarrage du fonctionnement dudit processeur hôte (H-CPU) dudit équipement informatique (OC).

[Revendication 11] Equipement informatique (OC) comportant un processeur hôte (H-CPU)

et au moins un module de sécurisation (MS) selon l'une quelconque des revendications 1 à 9.

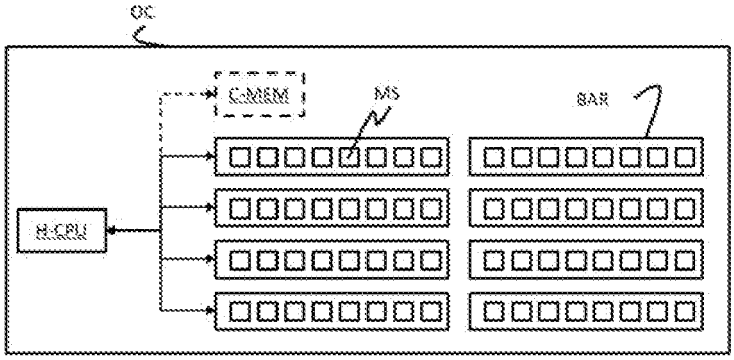
[Revendication 12] Equipement informatique (OC) selon la revendication 11, ledit processeur hôte (H-CPU) comportant un environnement d'exécution sécurisé (TEE) configuré pour envoyer une heuristique (H_+) utilisée pour ladite évaluation heuristique audit contrôleur (CTR) via ledit canal de contrôle (CC).

[Revendication 13] Equipement informatique (OC) selon la revendication 11 ou 12, ledit module de sécurisation (MS) étant selon les revendications 4 et 5, ledit processeur hôte (H-CPU) étant configuré pour exécuter un processus applicatif (PA) comportant des instructions pour :
- enregistrer des jetons (tok_{ext}) reçus d'un tiers de confiance (OP) dans un registre (REG_{OUT}) d'une dite mémoire, ces jetons (tok_{ext}) étant lesdits états de sécurité (st_sec_{OP}) de l'équipement informatique (OC) perçus par ledit tiers de confiance (OP) et signés par la clé privée (KPRIV) dudit tiers de confiance ;; et pour
- envoyer audit tiers de confiance (OP), des jetons (tok_{int}) lus dans un registre (REG_{IN}) d'une dite mémoire et correspondants à des états de sécurité (st_sec_{CTR}) de l'équipement informatique (OC) perçus par ledit contrôleur chiffrés par la clé publique (KPUB) dudit tiers de confiance.

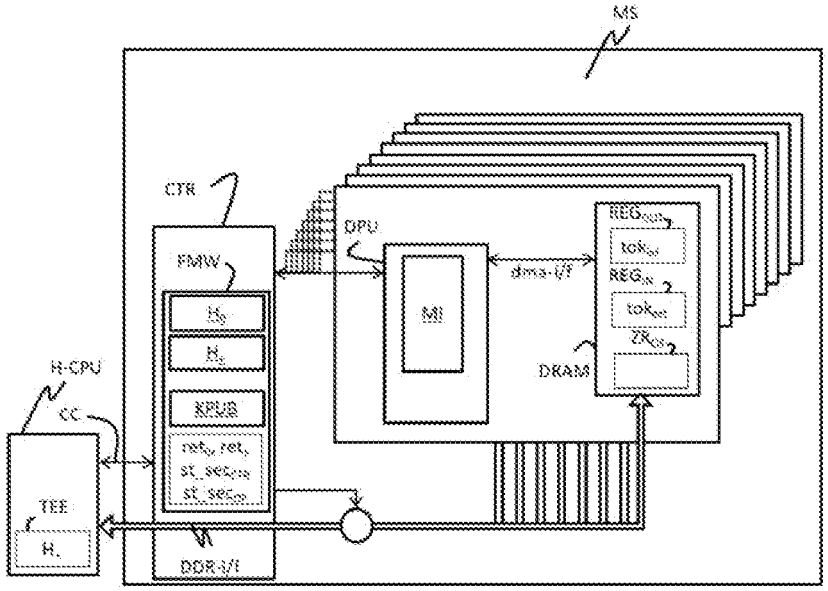
[Revendication 14] Programme d'ordinateur (PG) comportant des instructions pour l'exécution des étapes du procédé de sécurisation selon la revendication 10.

[Revendication 15] Support d'information (14) lisible par un contrôleur (CTR) et comportant des instructions d'un programme d'ordinateur selon la revendication 14.

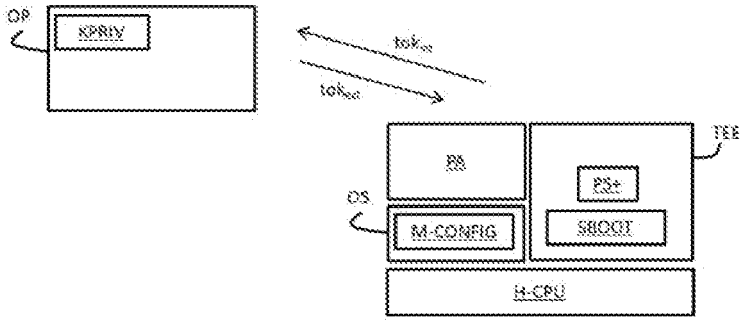
[Fig. 1]



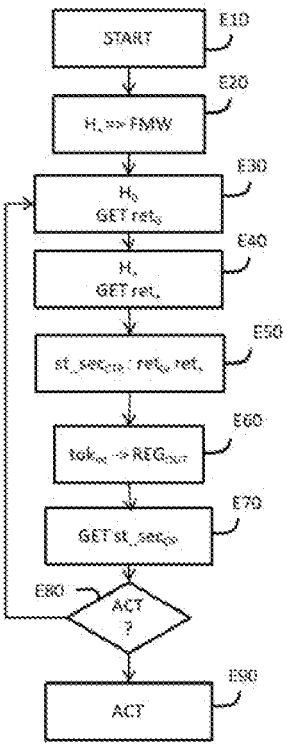
[Fig. 2]



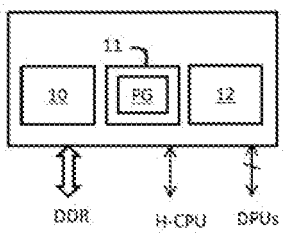
[Fig. 3]



[Fig. 4]


START
E10
E20
E30
GET ret0
E40
GET ret1
E50
E60
E70
GET st_sec0
E80
ACT
?
E90
ACT

[Fig. 5]


11
10
PG
12
DDR
H-CPU
DPUs

RÉPUBLIQUE FRANÇAISE

INSTITUT NATIONAL DE LA PROPRIÉTÉ INDUSTRIELLE

RAPPORT DE RECHERCHE PRÉLIMINAIRE

établi sur la base des dernières revendications déposées avant le commencement de la recherche

N° d'enregistrement national

FA 902979
FR 2114271

DOCUMENTS CONSIDÉRÉS COMME PERTINENTS		Revendication(s) concernée(s)	Classement attribué à l'invention par l'INPI
Catégorie	Citation du document avec indication, en cas de besoin, des parties pertinentes		
A	US 2019/042781 A1 (LUKACS SANDOR [RO] ET AL) 7 février 2019 (2019-02-07) * alinéas [0020] - [0033], [0037] - [0047], [0048], [0053], [0063] - [0070] - alinéas [0076] - [0080]; figures 1,2,3A 4, 5 *	1-15	G06F12/02 G06F21/70 G06F13/14
A,D	FR 3 032 814 A1 (UPMEM [FR]) 19 août 2016 (2016-08-19) * page 2, ligne 10 - page 5, ligne 2; revendications 1-15; figures 1-3 *	1-15	
A	US 2011/078791 A1 (PRAKASH GYAN [US] ET AL) 31 mars 2011 (2011-03-31) * alinéas [0006] - [0008], [0016] - [0021] - alinéas [0047] - [0063]; revendications 1-11; figures 3, 4 *	1-15	
A	US 2020/379923 A1 (LIDMAN PONTUS EVERT [US] ET AL) 3 décembre 2020 (2020-12-03) * alinéas [0005], [0021] - [0031], [0054]; figures 1,5 *	1-15	
A	EP 1 056 010 A1 (HEWLETT PACKARD CO [US]) 29 novembre 2000 (2000-11-29) * alinéas [0054] - [0055] *	1-15	

DOMAINES TECHNIQUES RECHERCHÉS (IPC)

G06F
G06N

Date d'achèvement de la recherche	Examinateur
11 août 2022	Jardon, Stéphan

CATÉGORIE DES DOCUMENTS CITÉS

X : particulièrement pertinent à lui seul
Y : particulièrement pertinent en combinaison avec un autre document de la même catégorie
A : arrière-plan technologique
O : divulgation non-écrite
P : document intercalaire

T : théorie ou principe à la base de l'invention
E : document de brevet bénéficiant d'une date antérieure à la date de dépôt et qui n'a été publié qu'à cette date de dépôt ou qu'à une date postérieure.
D : cité dans la demande
L : cité pour d'autres raisons
.....
& : membre de la même famille, document correspondant

EPO FORM 1503 12.99 (P04C14)

ANNEXE AU RAPPORT DE RECHERCHE PRÉLIMINAIRE RELATIF A LA DEMANDE DE BREVET FRANÇAIS NO. FR 2114271 FA 902979

La présente annexe indique les membres de la famille de brevets relatifs aux documents brevets cités dans le rapport de recherche préliminaire visé ci-dessus.
Les dits membres sont contenus au fichier informatique de l'Office européen des brevets à la date du **11-08-2022**
Les renseignements fournis sont donnés à titre indicatif et n'engagent pas la responsabilité de l'Office européen des brevets, ni de l'Administration française

Document brevet cité au rapport de recherche	Date de publication	Membre(s) de la famille de brevet(s)	Date de publication
US 2019042781 A1	07-02-2019	AU 2018311120 A1	30-01-2020
		CA 3069053 A1	07-02-2019
		CN 110998582 A	10-04-2020
		EP 3662385 A1	10-06-2020
		ES 2907777 T3	26-04-2022
		IL 272150 A	31-03-2020
		JP 2020529681 A	08-10-2020
		KR 20200035016 A	01-04-2020
		SG 11202000097T A	27-02-2020
		US 2019042781 A1	07-02-2019
		WO 2019025423 A1	07-02-2019
FR 3032814 A1	19-08-2016	CN 107257964 A	17-10-2017
		EP 3259674 A1	27-12-2017
		FR 3032814 A1	19-08-2016
		JP 6710219 B2	17-06-2020
		JP 2018511860 A	26-04-2018
		US 2018039586 A1	08-02-2018
		WO 2016132052 A1	25-08-2016
US 2011078791 A1	31-03-2011	AUCUN	
US 2020379923 A1	03-12-2020	CN 113853594 A	28-12-2021
		KR 20220005511 A	13-01-2022
		US 2020379923 A1	03-12-2020
		WO 2020242890 A1	03-12-2020
EP 1056010 A1	29-11-2000	EP 1056010 A1	29-11-2000
		EP 1181642 A1	27-02-2002
		US 7457951 B1	25-11-2008
		WO 0073904 A1	07-12-2000

EPO FORM P0465