



Guide d'installation et d'utilisation du Minidriver CPS

Guide d'installation et d'utilisation du Minidriver CPS

Version 2.1.0 du 03/12/2018

Historique du document

Version	Date	Auteur	Commentaires
1.0.0	25/05/2016	ASIP Santé	Création
1.1.0	20/09/2016	ASIP Santé	Prise en compte remarques
1.2.0	07/11/2016	ASIP Santé	Prise en compte remarques: <ul style="list-style-type: none">- Stratégie Minidriver CPS versus PKCS#11- Lecteurs PC/SC USB CCID -> Lecteurs PC/SC USB- Références pour déploiement par GPO du MSI Minidriver CPS- Précisions et conseils sur les paramètres exposés par le Minidriver CPS
2.0.0	20/11/2018	ASIP Santé	Déploiement via le Windows Update
2.1.0	03/12/2018	ASIP Santé	Déploiement via le Windows Service Update Server

1 Références

N°	Version	Date	Auteur	Document
[1]	v2.02.09		PC/SC Workgroup	PC/SCv2 Part 10
[2]	v110		USB	CCID
[3]			USB	USB LangID
[4]	V2.40		OASIS	PKCS#11
[5]	V5.1.3	21/03/2016	ASIP Santé	Guide d'installation et d'utilisation de la Cryptolib CPS
[6]	V2.3.8	30/09/2015	ASIP Santé	Guide de mise en œuvre de la partie sans-contact des cartes CPx
[7]	V7.07	25/02/2016	Microsoft	Smart Card Minidriver Specification

Tableau 1 : Références

2 Résumé

Le déploiement des usages basés sur la carte CPS se heurte aux traditionnels problèmes de déploiement rencontrés dans la PKI et dans la carte à puce en général:

1. qualité et d'exhaustivité des collectes d'identité
2. transmission de la carte au porteur
3. installation de lecteur de carte
4. installation des "middlewares" de la carte à puce, permettant aux applications d'utiliser au mieux les capacités cryptographiques de carte
5. configuration et de qualité des applications exploitant fonctionnellement les capacités cryptographiques de carte

Le problème **1**- "collecte des identités" est couvert par l'activité d'enregistrement menée par l'ASIP Santé et cadré par l'arrêté RPPS et différents projets tels qu'EPARSE qui en élargit le périmètre.

Le problème **2**- "transmission de la carte au porteur" est historiquement porté par l'ASIP Santé avec l'aide de partenaires reconnus du secteur qui a permis d'assurer la bonne délivrance de plus d'1 million de cartes CPS en production: cet aspect est bien couvert.

L'ASIP Santé adresse les problèmes 3, 4 et 5 en se basant sur les paradigmes suivants:

- **P1**: Suivre le plus fidèlement possible les normes et standards - internationaux si possible - en vigueur
- **P2**: S'adapter le plus possible aux architectures propriétaires identifiées comme étant largement diffusées et adoptées ("standards de facto")

Le paradigme **P1** est à l'origine de la carte CPS3, qui fait passer la CPS au format standard **IAS-ECC**.

Il est à l'origine de l'**abandon des API CPS**, propriétaires, au profit du standard **PKCS#11**. Ceci a permis une intégration native de la carte CPS avec le navigateur Mozilla **Firefox** ou avec **Linux** par exemple.

Le paradigme **P1** permet aussi de répondre au problème d'installation des lecteurs de cartes en identifiant **PC/SC comme standard international reconnu** d'accès au lecteur de carte ainsi qu'à la carte.

Les protocoles PSS, le GALSS et les lecteurs SESAM-Vitale sont à l'origine d'une hétérogénéité du parc de lecteurs en Santé&Social, de problèmes de qualité et de cycle de vie logicielle, de distorsions d'architecture ainsi que d'une inertie technique et projet qui freine les déploiements effectifs de nouveaux usages avec la CPS.

Au contraire, **PC/SC** permet une intégration **native** avec les **principaux OS** de lecteurs **simples** d'usage, **peu chers**, couvrant une **large gamme de fonctionnalités** (PINPad...) tout en autorisant les extensions propriétaires (certains lecteurs facilitent les opérations de signature, la gestion du PIN, les appels de fonctions lecteurs propriétaires...), **plug-and-play** et faisant l'objet de **processus de certification** indépendant (PC/SC Workgroup).

Le paradigme **P2** identifie:

- 1 PKCS#11 sous Linux
- 2 CryptoTokenAPI sous Mac OS X (anciennement TokenD)
- 3 CryptoAPI / CNG sous Windows

comme étant les 3 standards de facto d'accès aux capacités cryptographiques d'une carte à puce.

Le présent document décrit le projet Minidriver CPS de l'ASIP Santé, qui s'attache à suivre le standard de facto "CryptoAPI" sous Microsoft Windows.

Ce document s'adresse à des développeurs, éditeurs, chefs de projet informatique désireux d'adresser, de façon industrielle, les problématiques de déploiement et de mise à jour d'applications utilisant les capacités de la carte CPS tout en agilisant les installations et en minimisant les tâches demandées aux utilisateurs finaux pour y parvenir.



Accompagnement

Pour toute question et échange sur ce document :

editeurs@asipsante.fr

Tableau 2 : contact accompagnement ASIP Santé

3 Sommaire

1	Références	3
2	Résumé	4
3	Sommaire	6
4	Glossaire	7
5	Liste des entreprises citées.....	8
6	Avertissements	9
7	Cas d'usage couverts par le Minidriver CPS.....	10
7.1	Architecture d'accès à la carte CPS sous Windows XP	10
7.2	Architecture de type Windows Vista ou supérieur avec les fournitures Santé&Sociale actuelles.....	11
7.3	Architecture Minidriver cible	12
7.4	Principe général de déploiement de "pilotes" ("drivers").....	15
7.5	Déploiement de "pilotes" avec l'OS	16
7.6	Déploiement de "pilotes" via Windows Update	17
7.7	Mise en application du déploiement de "pilotes" via Windows Update adapté au cas d'usage "CPS"	18
8	Autres modes d'installation du Minidriver CPS	21
8.1	Installation du Minidriver CPS via le Catalogue Microsoft	21
8.1.1	Installation hors Windows Update	21
8.1.2	Installation avec le WSUS depuis le « Windows Catalog »	24
8.2	Mise à jour du Minidriver CPS via Windows Update	26
9	Utilisation du Minidriver CPS	28
9.1	Vérification de la présence des certificats CPS dans le magasin de certificats	28
9.2	Authentification web par carte CPS	28
9.3	Configuration avancée du Minidriver CPS	29
9.4	Utilisation avec une application .NET "bureau classique"	29
9.5	Utilisation avec une application .NET "UniversalApp"	30
9.5.1	Modèle de déploiement d'application Universelle utilisant le Windows Store	30
9.5.2	Wizard UniversalApp	31
9.5.3	Capacités de l'application universelle	31
9.5.4	Signature numérique avec la carte CPS dans une application Universelle avec le Minidriver CPS	32
9.5.5	Vérification de signature numérique.....	33
9.5.6	Recherche de certificat.....	34
9.5.7	Récupération logicielle du contenu d'une page Web protégée par authentification mutuelle par carte CPS.....	35
9.5.8	Récupération des données métier CPS via le Minidriver CPS.....	38
10	Annexe - Gestion du minidriver en ligne de commande	40
10.1	Installation	40
10.2	Désinstallation	40
11	Annexe - Message d'erreur suite à une insertion	40
	de carte CPS dans un lecteur PC/SC en l'absence de Minidriver ou de CSP	40
12	Annexe – Problèmes connus	42
13	Annexe – Liste des figures	43
14	Annexe – Liste des tableaux	45
15	Notes	46

4 Glossaire

Abréviation	Signification
API	Application Programming Interface
ASIP Santé	Agence des Systèmes d'Information Partagés de Santé
CAPI	Windows Cryptographic API
CNG	Cryptography API: Next Generation
CPx	Famille de cartes à puce émises par l'ASIP Santé comprenant CDA, CDE, CPA, CPE, CPF et CPS
CSP	Cryptographic Service Provider
KSP	Key Storage Provider
GPO	Group Policy Object (stratégies de groupe en français)
MSI	MicroSoft Installer
OS	Operating System (Système d'exploitation en français)
PC/SC	Personal Computer / Smart Card
PKCS#11	Public Key Cryptographic Standards #11: API définissant une interface générique pour les périphériques cryptographiques
URL	Universal Ressource Locator
WSUS	Windows Server Update Services

Tableau 3 : Glossaire

5 Liste des entreprises citées

Le présent document cite les produits des entreprises ou organismes suivants:

Nom	Site Web	Lien avec le document
ASIP Santé	esante.gouv.fr	CPx, Cryptolib CPS v5, testssl.asipsante.fr, Minidriver CPS
GIE SESAM-Vitale	www.sesam-vitale.fr	GALSS, PSS, lecteurs SESAM-Vitale
Microsoft	www.microsoft.com	Windows, CAPI/CAPI2/CNG, Minidriver, .NET, Windows Store, UniversalApp
Mozilla	www.mozilla.org	Mozilla Firefox, AMO
PC/SC Workgroup	www.pcscworkgroup.com	PC/SC

Tableau 4 : Entreprises citées

6 Avertissements

Sur le nécessaire strict respect des procédures décrites dans le document

L'attention de l'utilisateur est attirée sur l'importance de respecter strictement les procédures décrites dans le présent document.

Toutes les procédures qui y sont décrites ont été préalablement testées par l'ASIP Santé. En cas de non-respect de ces procédures, des dysfonctionnements dans l'environnement de travail de l'utilisateur peuvent apparaître.

En cas de dysfonctionnement, quel qu'il soit, l'ASIP Santé prêtera dans la mesure du possible assistance à l'utilisateur, qui ne pourra rechercher sa responsabilité en cas de non-respect des procédures décrites dans le présent document.

Sur les liens externes

Le présent document contient des liens vers des sites Internet.

Ces liens ne visent qu'à informer l'utilisateur. Ces sites Web ne sont pas gérés par l'ASIP Santé et l'ASIP Santé n'exerce sur eux aucun contrôle : leur mention ne saurait engager l'ASIP Santé quant à leur contenu. L'utilisation des sites tiers mentionnés relève de la seule responsabilité du lecteur ou de l'utilisateur des produits documentés.

Sur les copies d'écran

Les copies d'écran présentées dans ce document sont données à titre illustratif.

Les pages ou écrans réellement affichés peuvent être différents, notamment en raison de montées de version ou de configurations d'environnements différentes.

Citations

L'ASIP Santé est contrainte de citer le nom de certaines entreprises recensées au Tableau 4 afin d'apporter toute l'aide nécessaire au lecteur.

Les entreprises citées peuvent prendre contact avec l'ASIP Santé à l'adresse email editeurs@asipsante.fr pour toute demande en lien avec la citation les concernant.

Les entreprises non citées dans ce manuel et ayant une activité en lien avec la carte CPx, les IGC, les scénarios de déploiement ou les "stores" peuvent également se faire connaître auprès de l'ASIP Santé en la contactant à la même adresse.

Tableau 5 : Avertissements

7 Cas d'usage couverts par le Minidriver CPS

Afin de bien comprendre le fonctionnement des cartes à puce sous Windows, il est nécessaire d'avoir en tête le point particulier suivant:



Approche Microsoft de l'intégration des cartes à puce dans le système Windows

Microsoft Windows traite les cartes à puces comme tous les autres périphériques matériels (imprimantes, clavier, souris...): pour rendre les cartes à puces visibles du système, il est juste nécessaire de leur faire correspondre un "pilote" (driver).

Cette approche permet aux émetteurs de cartes à puces de bénéficier des méthodes et infrastructures mises en place par Microsoft pour gérer et déployer des drivers.

Tableau 6 : Sous Windows, les cartes à puce sont des périphériques parmi tant d'autres

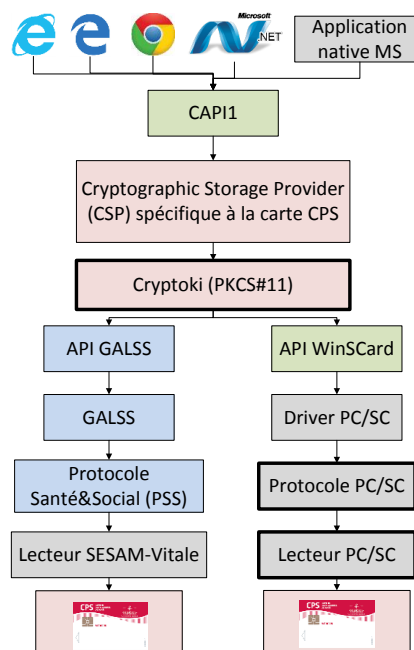
7.1 Architecture d'accès à la carte CPS sous Windows XP

Sous Windows, la Cryptolib CPS déclare depuis quelques années déjà un "CSP" (Crypto Service Provider).

Le **CSP de la Cryptolib CPS** agit et s'installe comme un **driver** de carte à puce dédié à la CPS et permet au reste du système de "voir" la carte CPS via:

1. la propagation automatique des certificats de la CPS dans ses magasins de certificats
2. l'utilisation "native" de la carte (sans "addons") par les navigateurs (Internet Explorer, Edge, Google Chrome) ou par toute autre application utilisant les APIs "**CAP11**" (CryptoAPI version "**1**").

Les appels applicatifs utilisant la carte passent par les chemins suivants:



Légende:

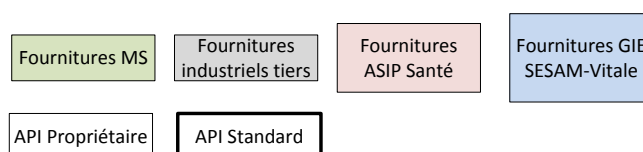


Figure 1: Architecture classique de la Cryptolib CPS sous Windows

7.2 Architecture de type Windows Vista ou supérieur avec les fournitures Santé&Social actuelles

Depuis Windows Vista, Microsoft a introduit un **changement dans son architecture cryptographique** qui oblige l'ASIP Santé à s'adapter: les API de CAPI1 et de l'API "CNG" ("Cryptography API: Next Generation") ont été abstraite par CAPI2.

Dans cette nouvelle architecture, qui fait intervenir un nouveau composant dit "BaseCSP", le fournisseur de carte à puce écrit des **drivers de cartes à puces** appelés "**Minidrivers**" qui exposent toutes les fonctionnalités d'une carte à puce tout en nécessitant d'écrire moins de code source. L'architecture d'accès à la CPS sur un poste de travail Windows Vista ou supérieur se complexifie encore un peu plus:

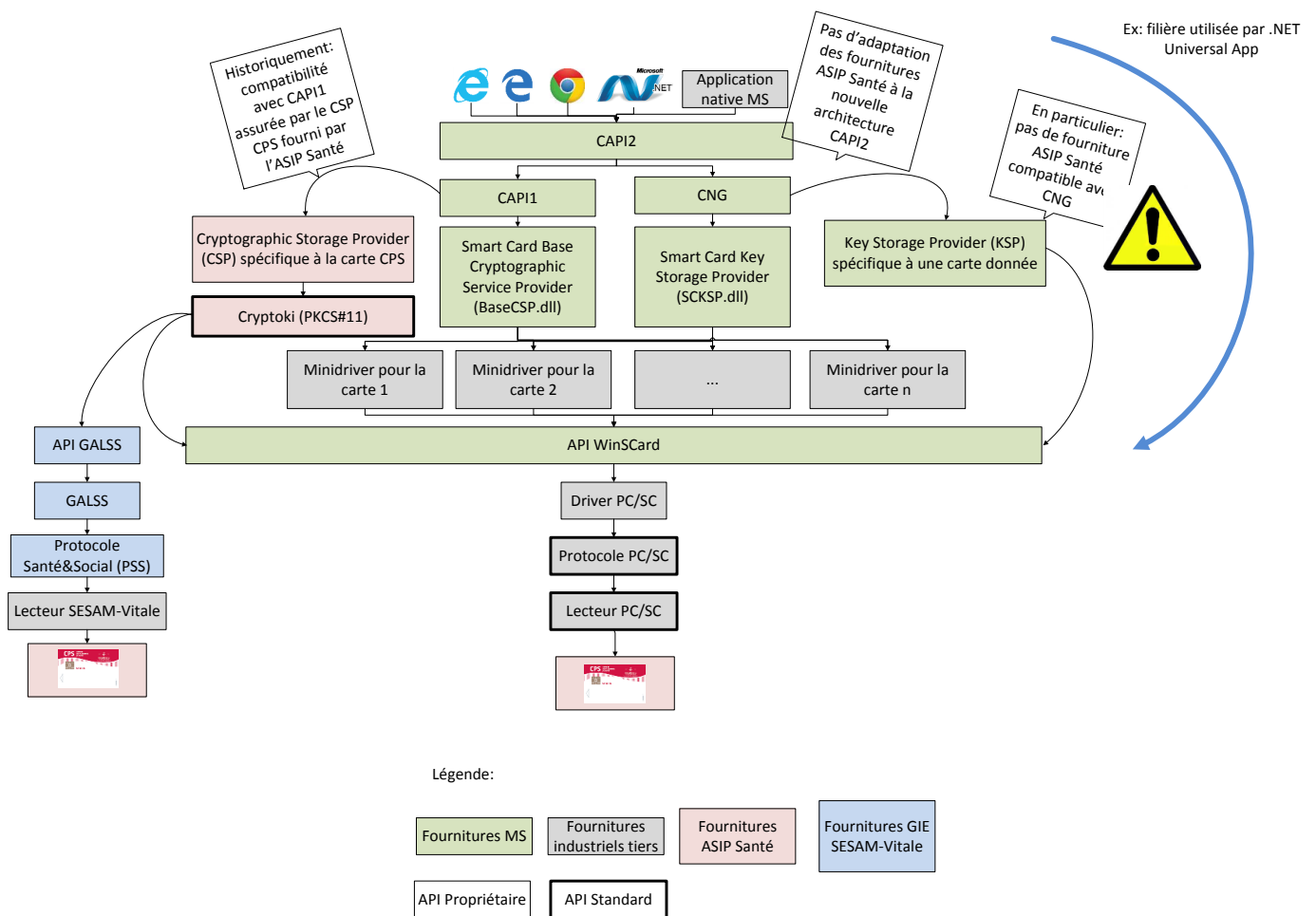


Figure 2: Architecture de la Cryptolib CPS sous Windows depuis Windows Vista



Impact 00

L'architecture **Minidriver** est une architecture **100% Microsoft**: le minidriver CPS n'est pas adaptable à Mac OS X, Linux, Android ou iOS.

Tableau 7 : Impact 00: Minidriver uniquement sous Windows

7.3 Architecture Minidriver cible

Il est dès lors possible d'identifier les principaux impacts des changements induits par l'architecture Minidriver.



Impact 01

Le principal impact de cette nouvelle architecture se passe au niveau de l'utilisation des capacités cryptographiques de la carte (pour l'ASIP santé, de la carte CPS) par les applications, ces dernières disposant d'appels CAPI2 au lieu de CAPI1.

Tableau 8 : Impact 01: Changement d'API pour accéder aux capacités de la carte CPS

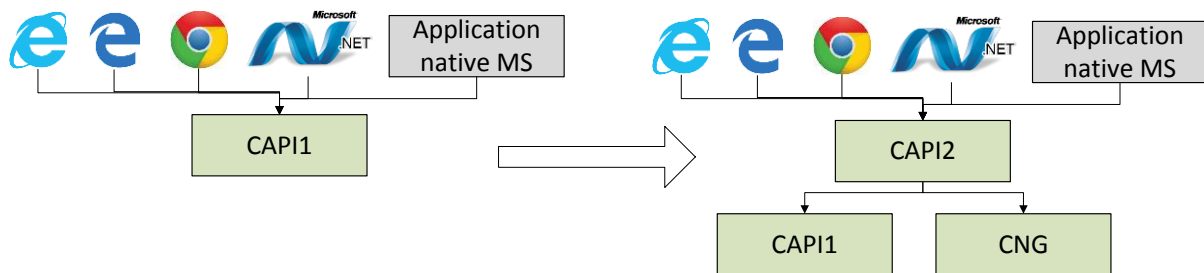


Figure 3: Différences d'appels applicatifs depuis l'introduction de l'architecture CNG par Microsoft

- ⇒ A partir de Windows Vista (sorti en 11/2006), toutes les applications utilisant des API CAPI2 non "mappées" sur les anciennes API CAPI1 ne fonctionnent pas avec les fournitures CPS de type Cryptolib CPS v5.
- ⇒ Malgré les années, ce n'est toujours pas le cas pour les navigateurs Internet Explorer, Edge et Chrome mais cela arrivera bientôt: quand ce sera le cas, ces navigateurs ne fonctionneront plus avec la Cryptolib CPS et le CSP CPS "monolithique" actuels.
- ⇒ A contrario, les navigateurs font de plus en plus appel à l'exhaustivité de l'API CAPI. L'incomplétude des implémentations successives du CSP Cryptolib CPS est à l'origine de dysfonctionnements complets de l'authentification forte par carte CPS (cas Chrome 41 et Edge sous Windows 10), obligeant à compléter son implémentation.
- ⇒ Le premier cas d'utilisation généralisé de CAPI2 est arrivé avec Windows 8 (sorti en 10/2012): WinRT utilise des API .NET d'accès aux composants cryptographiques appelant des fonctions CAPI2 qui nécessitent une implémentation {CAPI1; CNG} complète par le fournisseur de la carte à puce utilisée.



Impact 02: Impact du changement CAPI1 vers CAPI2

Le changement d'API de CAPI1 vers CAPI2 ainsi que l'adoption progressive par les applications de l'API CNG justifient à eux seul que l'on adapte pleinement le middleware d'accès à la CPS à la nouvelle architecture Minidriver introduite par Microsoft depuis Windows Vista.

Tableau 9 : Impact 02: nécessité de s'adapter à l'architecture Minidriver

Reprenant le schéma d'architecture générale d'accès à la carte sur les OS Microsoft récent, l'architecture cible avec la CPS et le Minidriver CPS est la suivante:

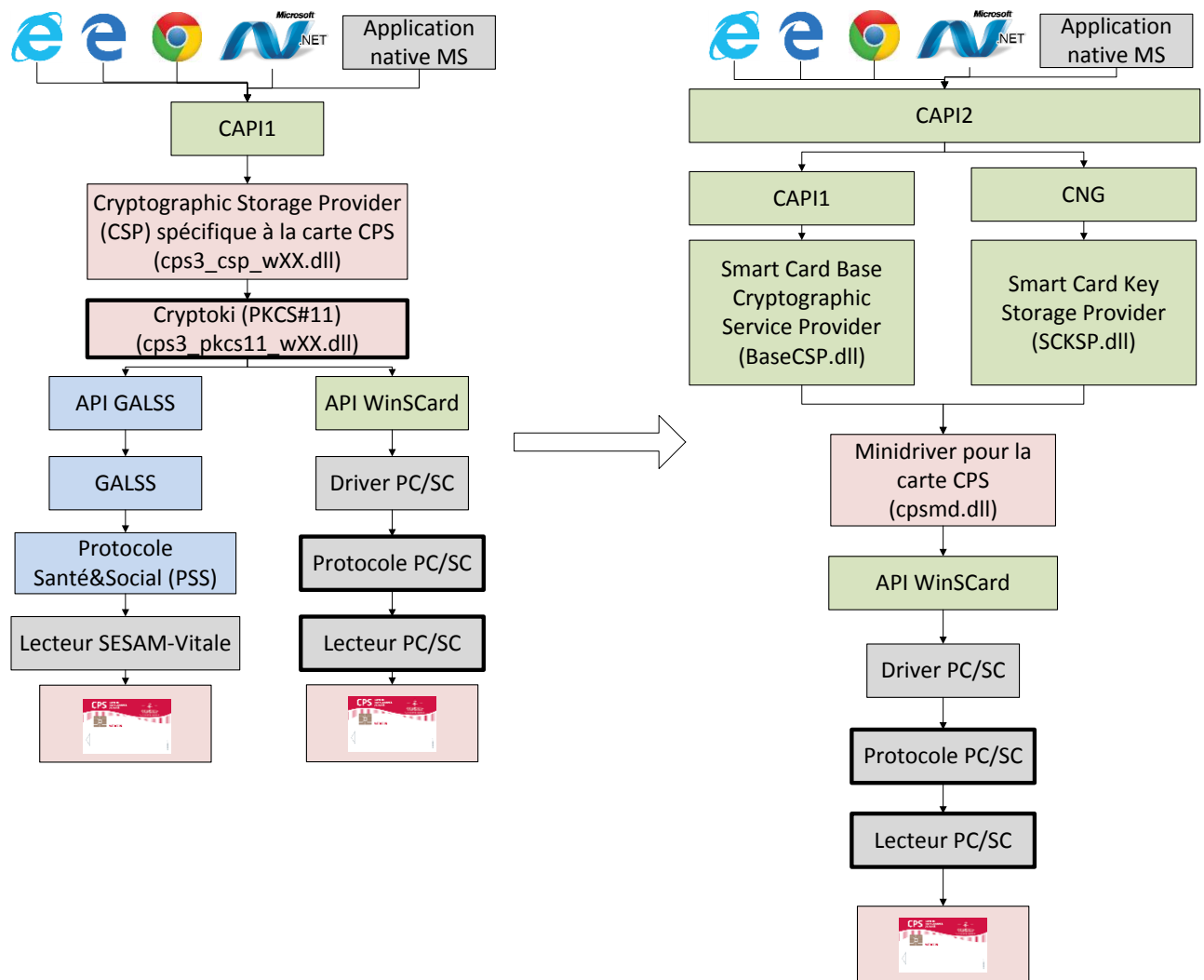


Figure 4: Architecture de la Minidriver CPS cible

La DLL "Cryptoki/PKCS#11" cps3_pkcs11_wXX.dll n'est pas distribuée par l'installateur du Minidriver CPS.

Cela ne signifie pas pour autant que l'ASIP Santé abandonne Cryptoki/PKCS#11 pour la CPS. Bien au contraire, conformément à sa stratégie poste de travail, l'ASIP Santé prévoit de maintenir et de développer ce composant qui est décrit dans d'autres documentations.



Impact 03

Cependant et à l'image du modèle retenu par Microsoft avec Windows Update et le Windows Store, la DLL "Cryptoki/PKCS#11" cps3_pkcs11_wXX.dll a vocation à être distribuée via le site addons.mozilla.org ("AMO") de la fondation Mozilla, via un XPI par les serveurs ASIP Santé (mais signé par Mozilla).

Tableau 10 : Impact 03: DLL cps3_pkcs11_wXX.dll et Minidriver CPS

**Impact 04**

- Appliquée à la carte CPS, l'architecture Minidriver Microsoft:
- minimise grandement les volumes et les complexités des fournitures logicielles Santé&Social
 - minimise les tâches de déploiement à gérer par la sphère Santé&Social par elle-même
 - minimise les installations requises du côté de l'utilisateur final

Réciproquement, cette nouvelle architecture augmente significativement la part des fournitures Microsoft et des tiers, sur qui se reportent les responsabilités en termes de déploiement, mises à jour, tests et qualité.

Tableau 11 : Impact 04: Intérêt du Minidriver CPS pour Santé&Social

L'architecture Minidriver n'est pas compatible avec l'architecture propriétaire GALSS/PSS: comme on peut le constater sur le schéma ci-dessus, toutes les fournitures Santé&Sociales du GIE SESAM-Vitale ne figurent plus sur le schéma de l'architecture Minidriver cible.

Ceci est dû au fait:

1. que le minidriver adresse uniquement l'API PC/SC
2. qu'il ne peut intrinsèquement pas être adapté aux protocoles carte propriétaires
3. qu'aucun driver PC/SC pour lecteurs SESAM-Vitale n'est actuellement disponible
4. que les drivers PC/SC pour lecteurs SESAM-Vitale ne prévoient actuellement qu'un support limité des évènements PC/SC

Le **Minidriver CPS couvre** donc les **usages** de la CPS hors lecture Vitale (et notamment hors FSE) **correspondant** à une **utilisation** de la **CPS** en **SI hospitalier** ou en **authentification publique** du porteur sur les télé-services nationaux (EspacePro, TLSi).

Tableau 12 : Impact 05: Pas de GALSS / PSS avec le Minidriver CPS**Impact 05**

L'installation du Minidriver CPS ne permet pas de faire fonctionner le navigateur Web Mozilla Firefox avec la carte CPS.

**Impact 06**

L'installateur Minidriver CPS ne déploie aucune DLL Cryptoki / PKCS#11.

La configuration de Mozilla Firefox, pour qu'il fonctionne avec la carte CPS, passera à terme par l'installation d'un module XPI distribué par l'AMO (Addons Mozilla Org).

Tableau 13 : Impact 06: Mozilla Firefox ne fonctionne pas avec le Minidriver CPS

La suite du document décrit l'installation et l'utilisation du minidriver CPS.

7.4 Principe général de déploiement de "pilotes" ("drivers")

Les pilotes - ou drivers - permettent aux éditeurs de systèmes d'exploitation de faire fonctionner des périphériques fabriqués par des tiers sans les connaître au préalable.

Réciproquement, les fabricants de matériel peuvent espérer faire fonctionner leurs matériels sur les systèmes de leur choix en écrivant des drivers dédiés conformément à ce que le système cible attend comme implémentation (notion de contrat ou d'interface).

Historiquement, les drivers sont donc écrits pour un système donné par le fabricant de périphérique qui le fournit à l'utilisateur au moment de l'achat. C'est l'utilisateur qui installe le driver après avoir branché le périphérique sur la machine cible:

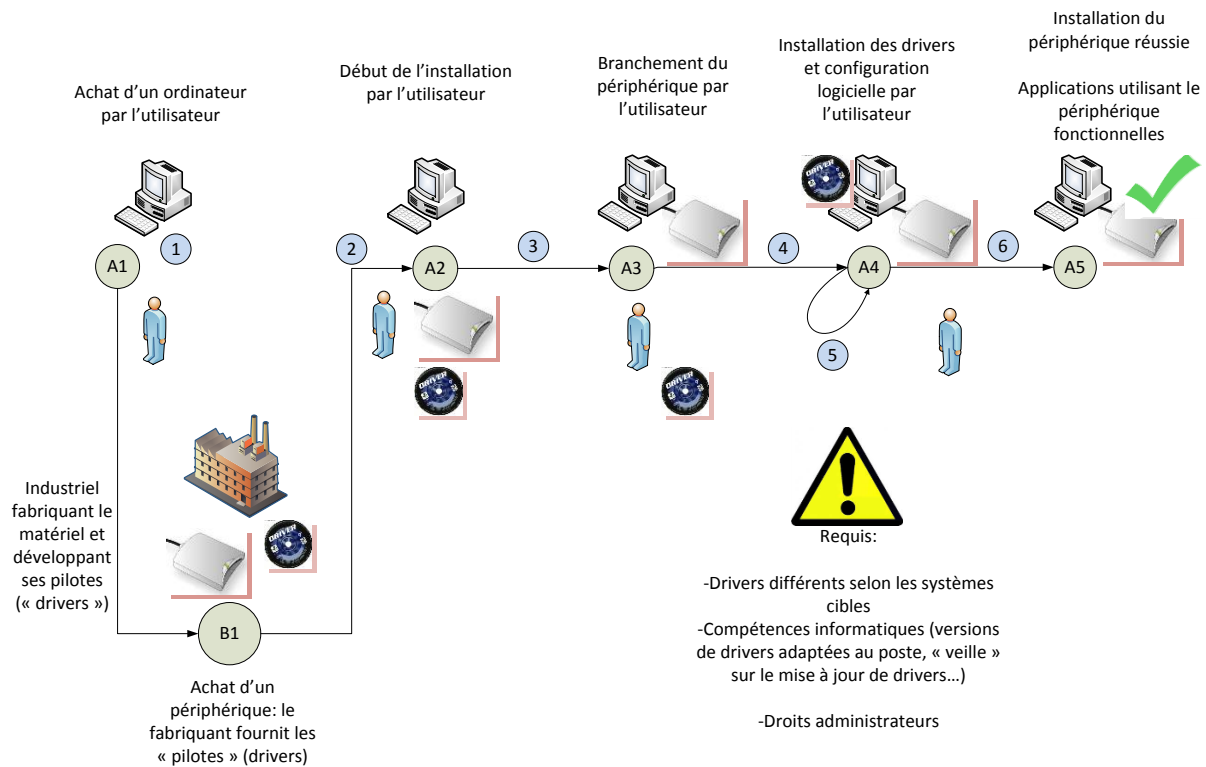


Figure 5: Déploiement: Principe général de déploiement de "pilotes" ("drivers")



**Carte CPS et
drivers**

Dans le cas d'usage "CPS" sous Windows, 2 périphériques sont concernés par le principe des "périphériques"/"drivers" :

- Le lecteur de cartes à puce
- La carte à puce CPS elle-même

Tableau 14 : Sous Windows, 2 périphériques et leurs drivers à installer: lecteur de carte et CPS

7.5 Déploiement de "pilotes" avec l'OS

Le principal problème de ce mode de diffusion est qu'il fait appel à des compétences informatiques qui peuvent être pointues d'autant plus que ces tâches sont répétées périphérie par périphérie :

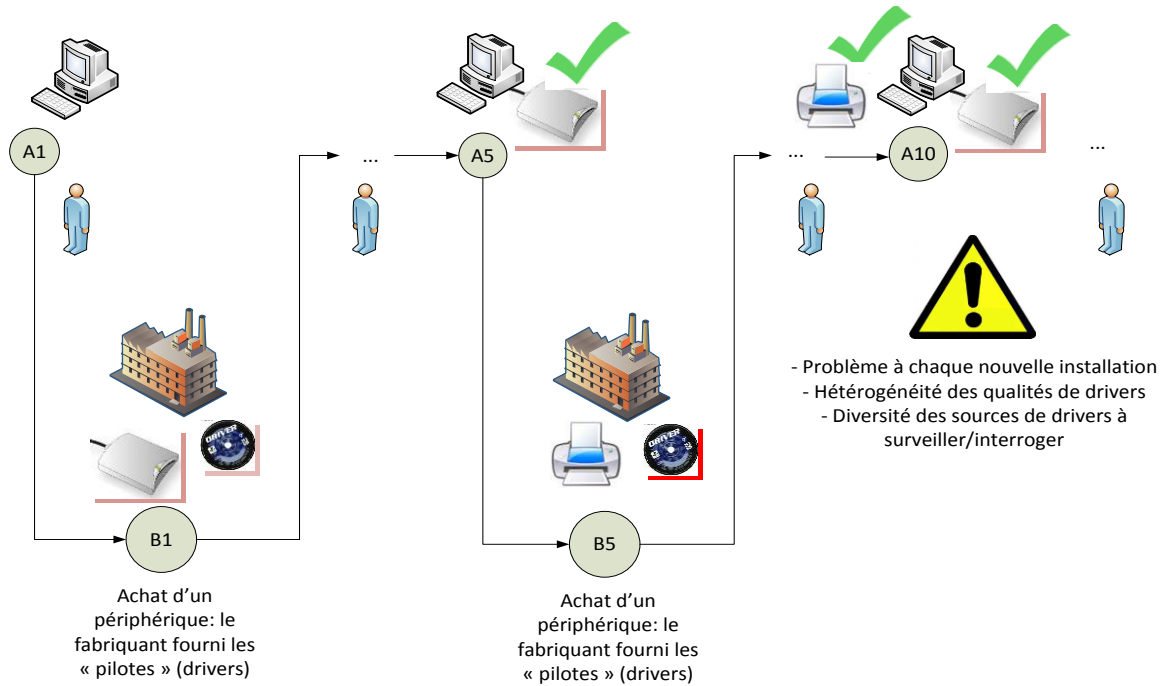


Figure 6: Déploiement: Répétition des installations de pilotes pour chaque périphérique

Afin de limiter les tâches de l'utilisateur final, Microsoft et des fabricants de périphériques se sont rapprochés pour que soient fournis les drivers de leurs périphériques en standard par Windows:

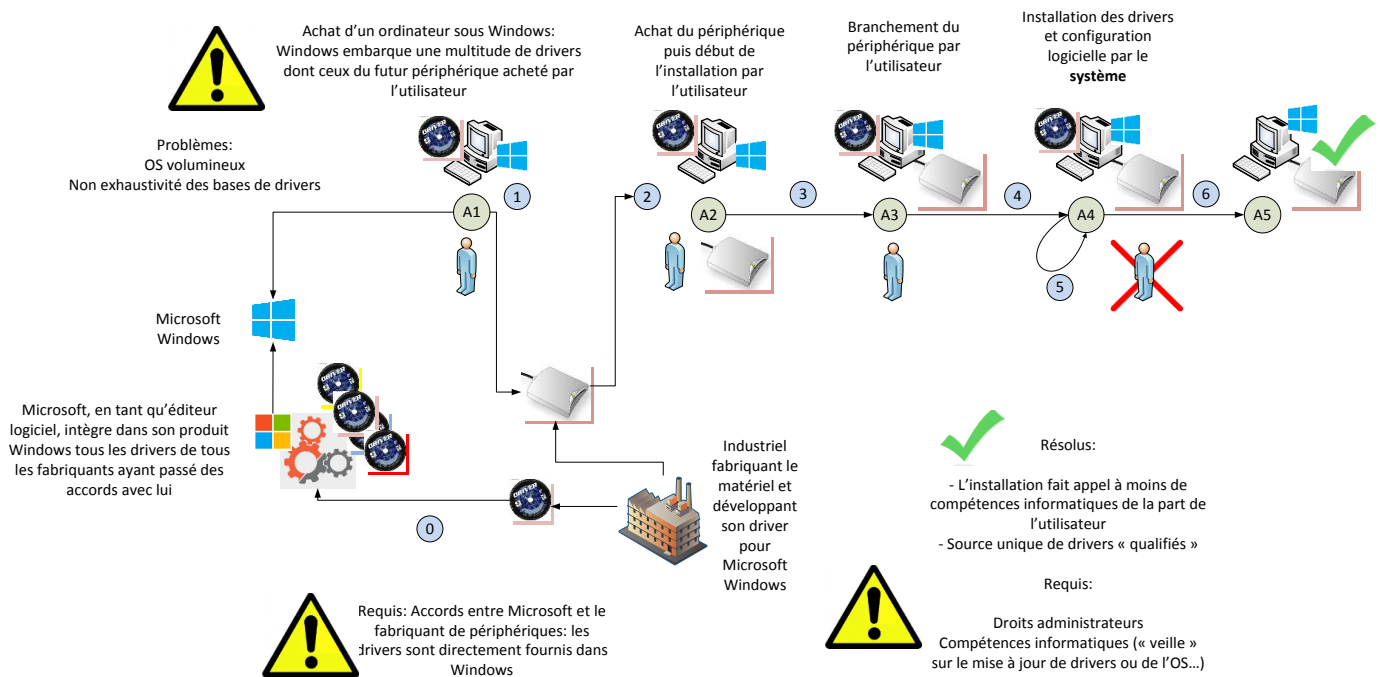


Figure 7: Déploiement: Déploiement de "pilotes" ("drivers") via l'OS

Le principal avantage de ce système est de reporter sur Microsoft, les industriels et le système lui-même des tâches que l'utilisateur final n'a plus à faire (identification du bon driver pour le système courant, installation du driver proprement dit) et d'agiliser les installations ("plug-and-play").

7.6 Déploiement de "pilotes" via Windows Update

Le raccourcissement des cycles de développements des OS (corrections de failles de sécurité...) ainsi la multiplication des sociétés fabriquant des périphériques (problèmes d'exhaustivité des bases de drivers dans les images des OS, évolutions rapides des matériels, changements de modèles, corrections et mises à jour de drivers) obligent à revoir les modèles de déploiement de driver en exploitant les possibilités des réseaux (Internet en particulier).

Le système de déploiements de drivers Microsoft repose sur Windows Update.

Le principe de déploiement et d'installation de drivers par Windows Update est le suivant:

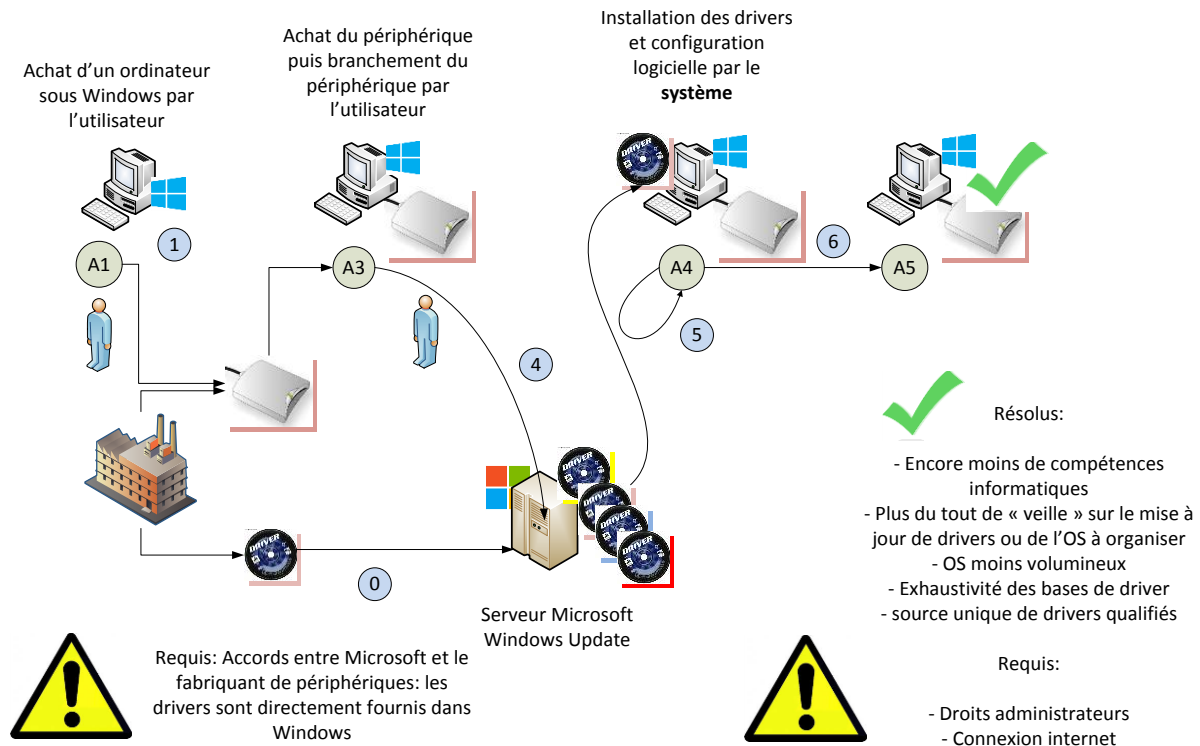


Figure 8: Déploiement: Déploiement de "pilotes" ("drivers") via Windows Update

Lors du branchement du périphérique (état **A3**), l'OS détecte l'événement d'insertion, se rend compte qu'il n'a pas les drivers du nouveau périphérique et se retourne seul vers Windows Update (lien **4**) pour le récupérer puis l'installer (état **A4**).

L'utilisateur n'a donc plus à identifier un pilote précis pour un OS précis puis à l'installer explicitement.

L'OS n'a pas non plus besoin de contenir en standard tous les drivers, ceux-ci pouvant alimenter le Store puis les OS clients sur demande a posteriori de la primo-installation.

7.7 Mise en application du déploiement de "pilotes" via Windows Update adapté au cas d'usage "CPS"

Appliqué à la carte CPS, le déploiement et l'installation de driver via Windows Update est le suivant. Il requiert un lecteur PC/SC USB et un Minidriver CPS développé par l'ASIP Santé :

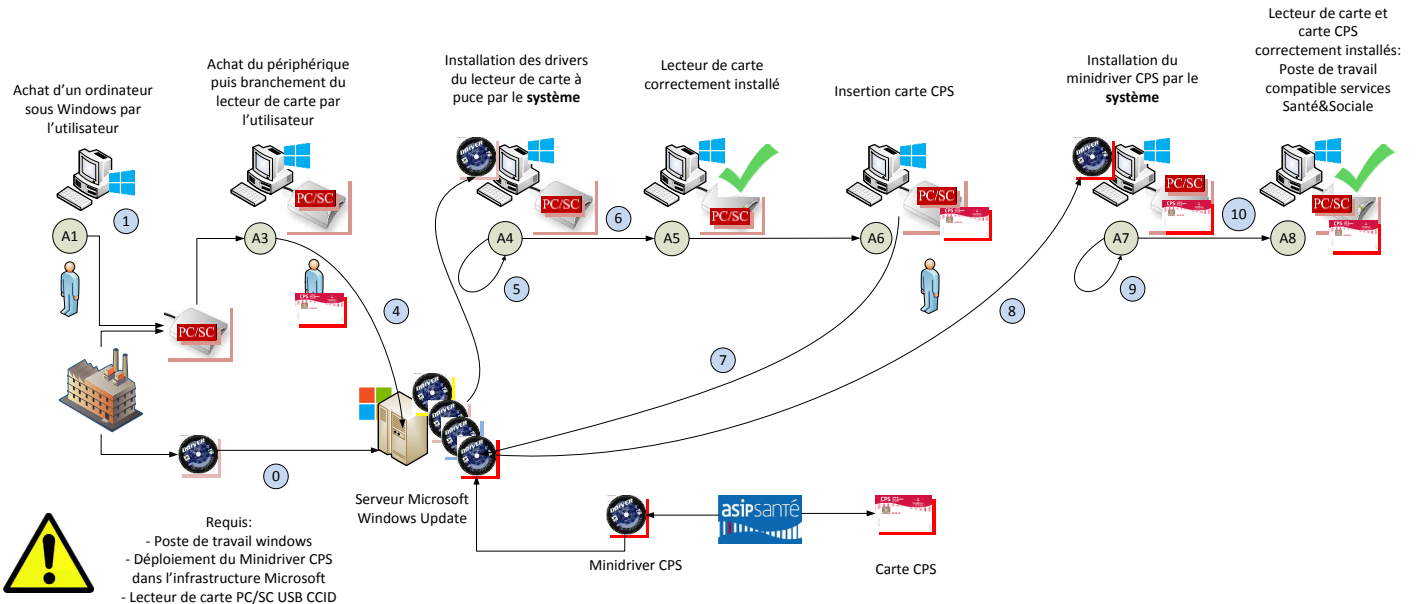


Figure 9: Déploiement: Déploiement de "pilotes" ("drivers") via Windows Update appliqué à la CPS

Lors du branchement du lecteur de carte à puce USB (état **A3**), l'OS détecte l'événement de branchement du lecteur, se rend compte qu'il est CCID et l'installe directement ou bien qu'il n'a pas les drivers du nouveau périphérique et se retourne seul vers Windows Update (lien **4**) pour le récupérer puis l'installer (état **A4**).

Lors de l'insertion de la carte CPS dans le lecteur de cartes (état **A6**), l'évènement **PC/SC** "insertion carte" (d'où l'importance d'avoir un lecteur PC/SC) est remontée par la stack PC/SC auprès de l'OS. Ce dernier repart sur Windows Update récupérer (lien **8**) et installer (lien **9**) le Minidriver CPS. Le Minidriver CPS fait office de driver CPS auprès du système pour en arriver à un poste de travail fonctionnel (état **A8**), i.e. prêt à utiliser la carte CPS pour accéder aux Téléservices Assurance Maladie ou à toutes autres fonctions de sécurisation offerte par la carte CPS.

Visuellement, pour l'utilisateur, le téléchargement des drivers du lecteur de carte est automatique au branchement du lecteur:

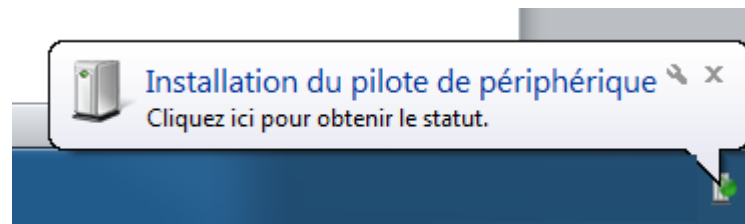


Figure 10: Windows 7: Installation de lecteur: message affiché lors du branchement du lecteur

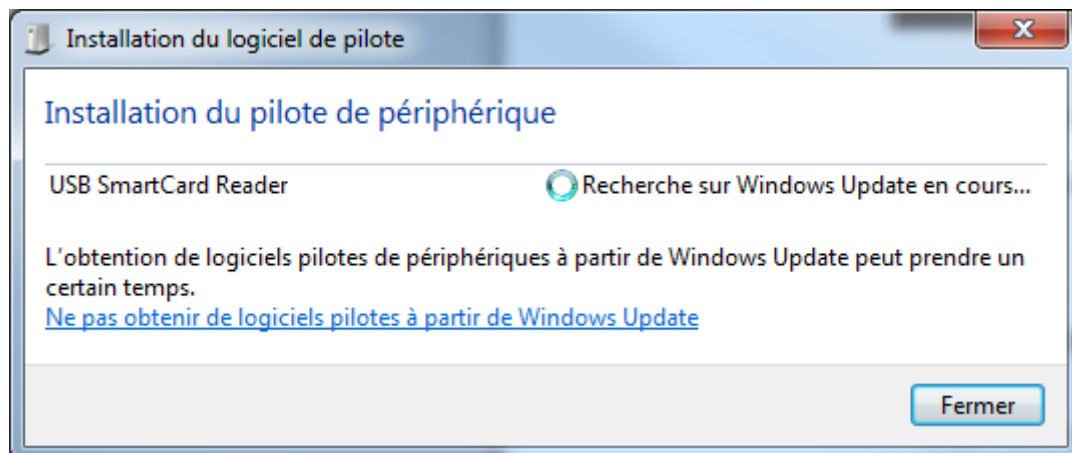


Figure 11: Windows 7: Installation de lecteur: message affiché pendant l'installation des pilotes du lecteur



Figure 12: Windows 7: Installation de lecteur: icône affiché pendant l'installation des pilotes du lecteur

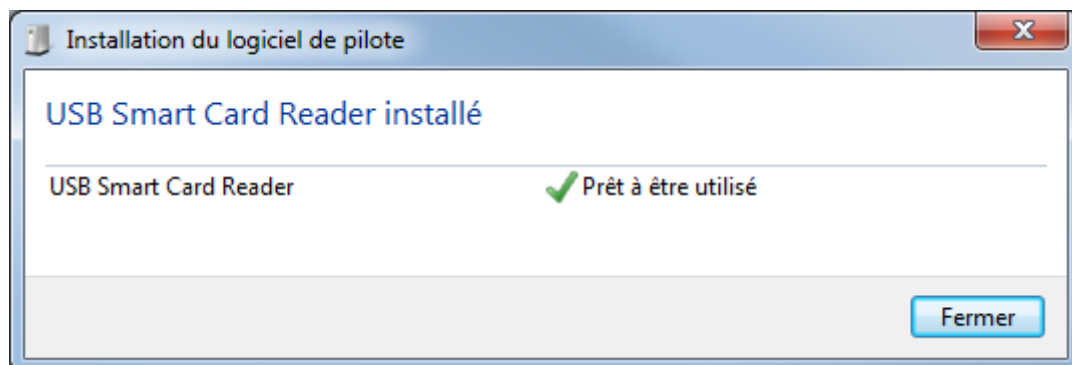


Figure 13: Windows 7: Installation de lecteur: message affiché en fin d'installation des pilotes du lecteur

Il en est de même par la carte CPS dès son insertion:

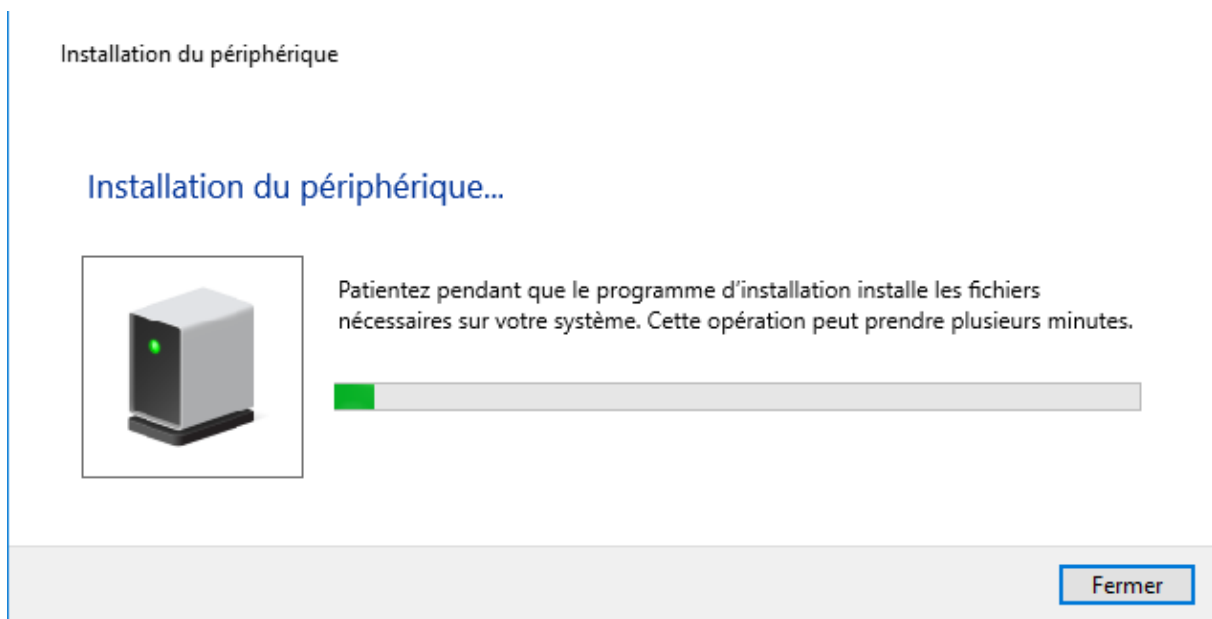


Figure 14: Windows 10: Installation de la carte CPS: message affiché lors de l'insertion de la carte

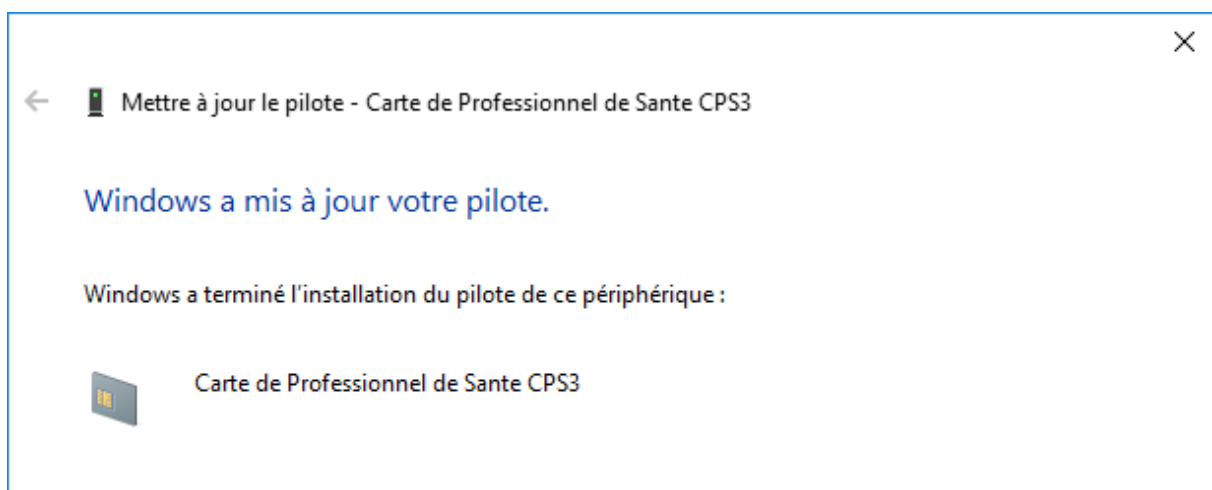


Figure 15: Windows 10: Installation de la carte CPS: message affiché en fin d'installation du Minidriver CPS



Carte CPS et Minidriver CPS

Le Minidriver CPS permet d'implémenter le cas d'usage où l'utilisateur n'a juste qu'à brancher son lecteur de carte et insérer sa carte CPS dedans afin d'en utiliser les capacités.

Les installations et mises à jour sont automatiques.

Elles ne dépendent plus des infrastructures Santé&Sociale pour le déploiement et ne requiert pas de connaissance informatique.

Tableau 15 : Résumé des cas d'usage permis par le Minidriver CPS.

8 Autres modes d'installation du Minidriver CPS

8.1 Installation du Minidriver CPS via le Catalogue Microsoft

Le Catalogue Microsoft contient l'ensemble des drivers qualifiés par Microsoft.

Il sert de base de drivers à Windows Update et WSUS.

Il permet aussi de récupérer manuellement des drivers afin:

- De les installer hors Windows Update
- De les déployer explicitement dans un WSUS local

8.1.1 Installation hors Windows Update

Aller sur le site suivant <http://catalog.update.microsoft.com/v7/site/Home.aspx>

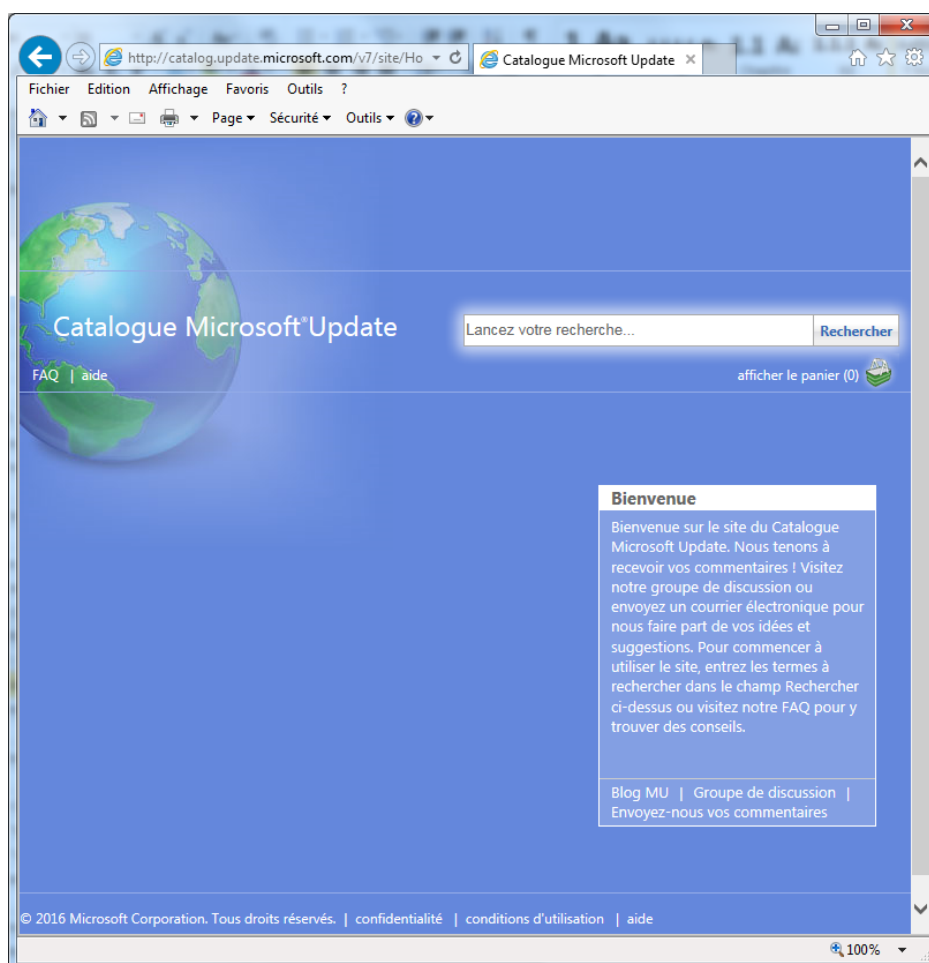


Figure 16: Windows Catalog: Page d'accueil

Recherchez les composants ASIP et « Ajouter » au panier ceux qui vous intéressent.

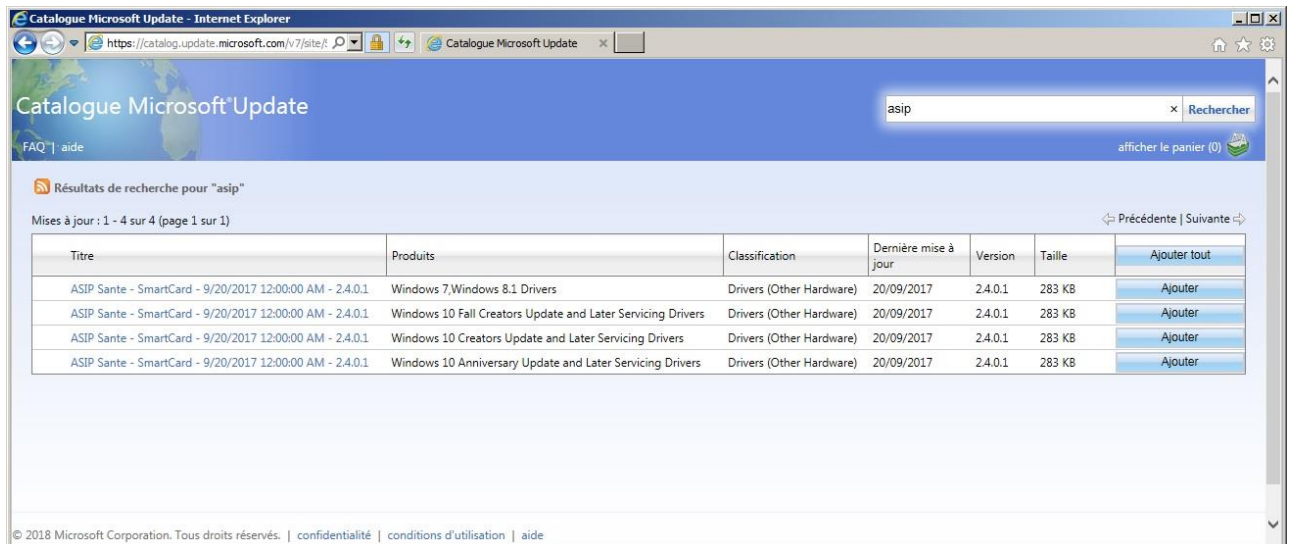


Figure 17: Windows Catalog: Résultat d'une recherche sur "asip"

Cliquez sur le lien « Afficher le panier »

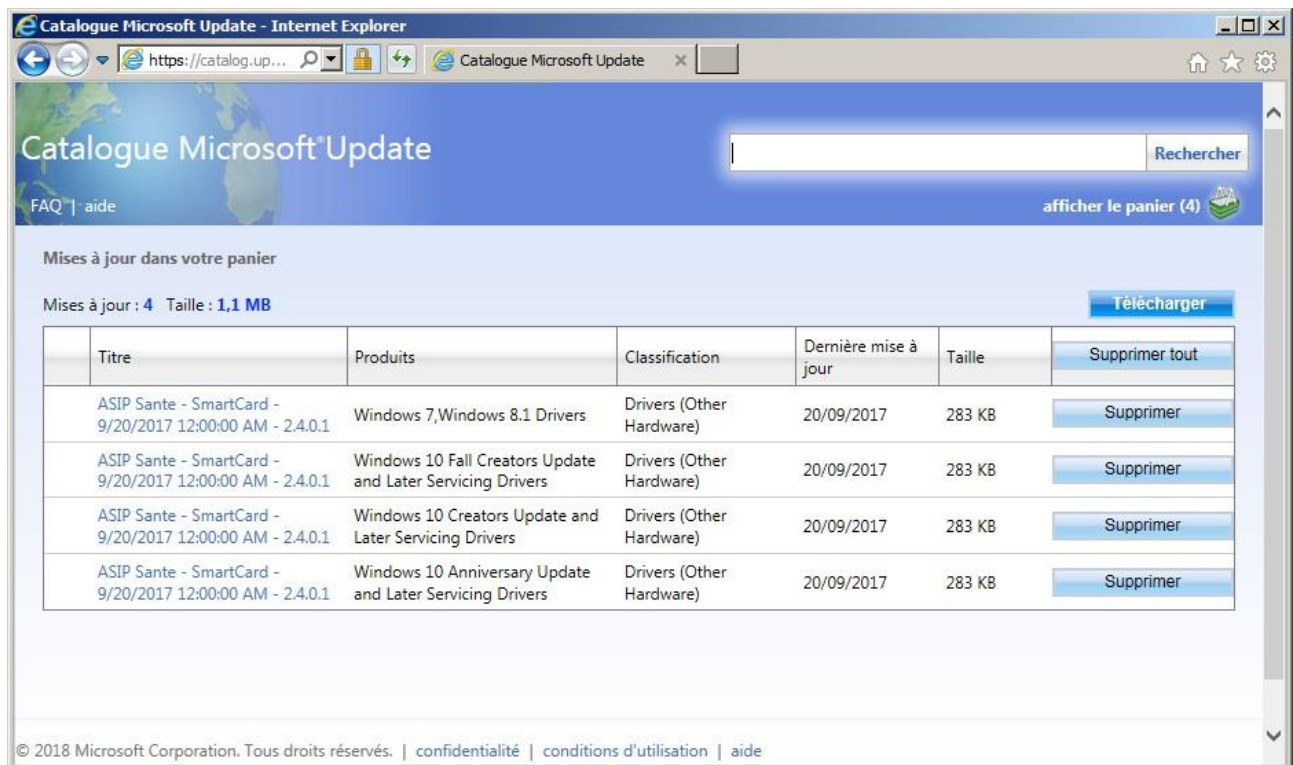


Figure 18: Windows Catalog: "afficher le panier"

Cliquez sur le bouton « Télécharger »

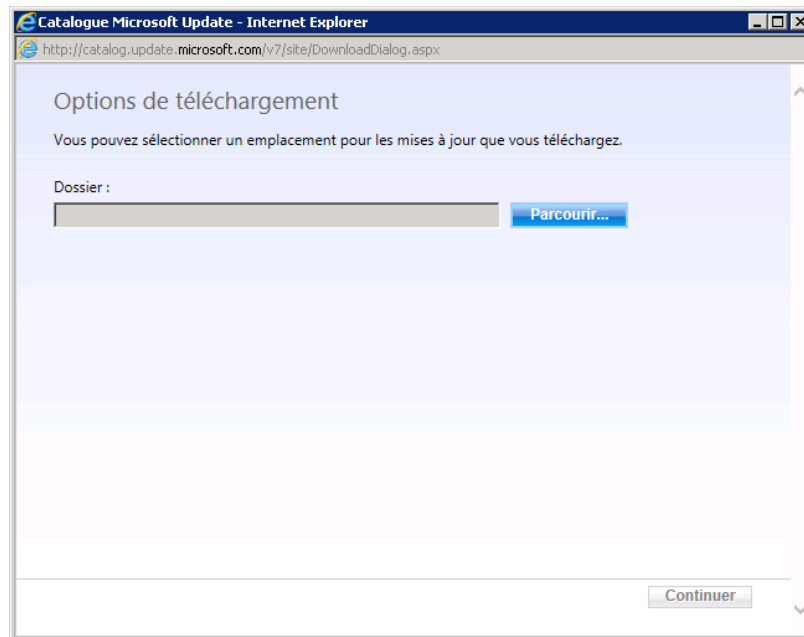


Tableau 16 : Windows Catalog: Choix du répertoire de destination

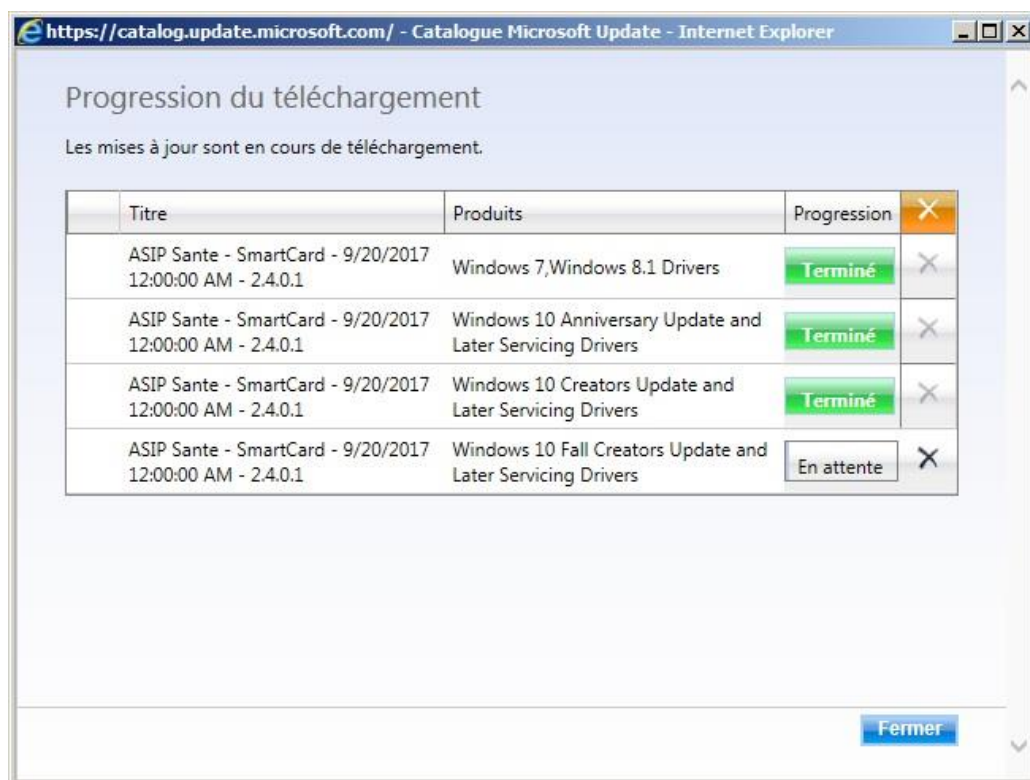


Tableau 17 : Windows Catalog: Progression du téléchargement

8.1.2 Installation avec le WSUS depuis le « Windows Catalog »

Depuis la console du WSUS ouvrez le site web du « Windows Catalog »

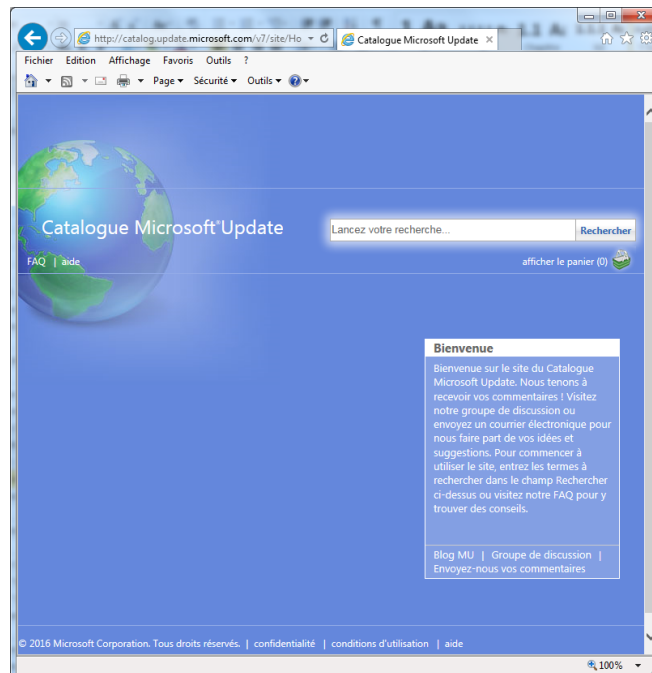
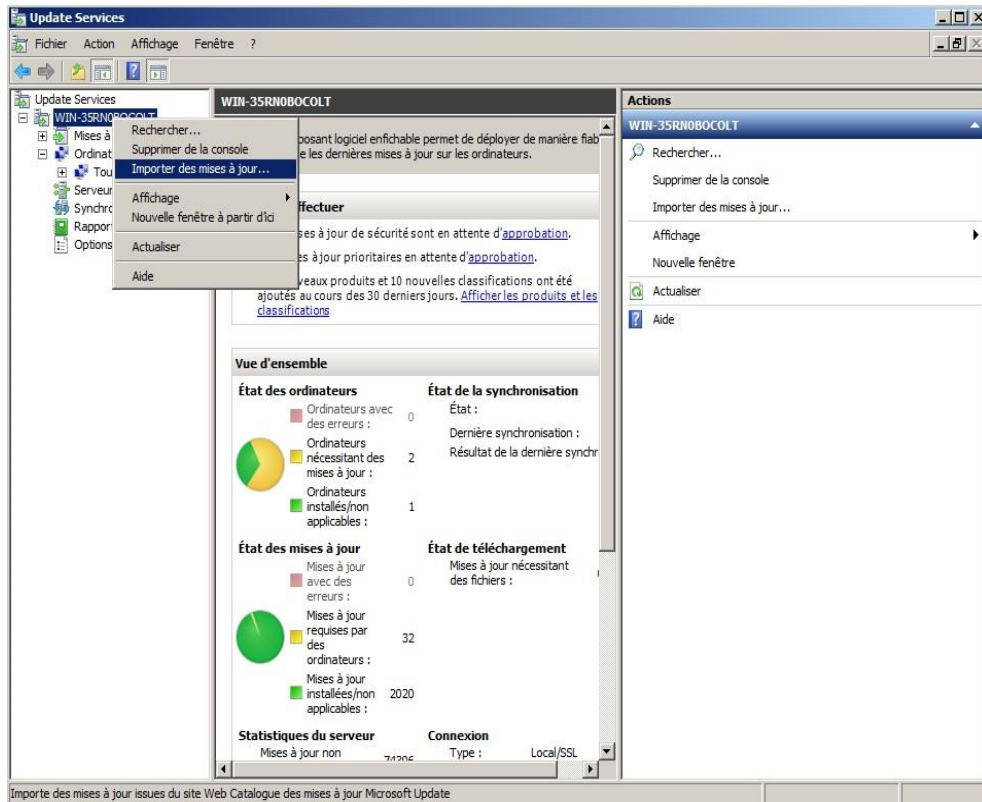


Figure 19: Windows Catalog: Page d'accueil

Recherchez les composants ASIP et « Ajouter » au panier ceux qui vous intéresse.

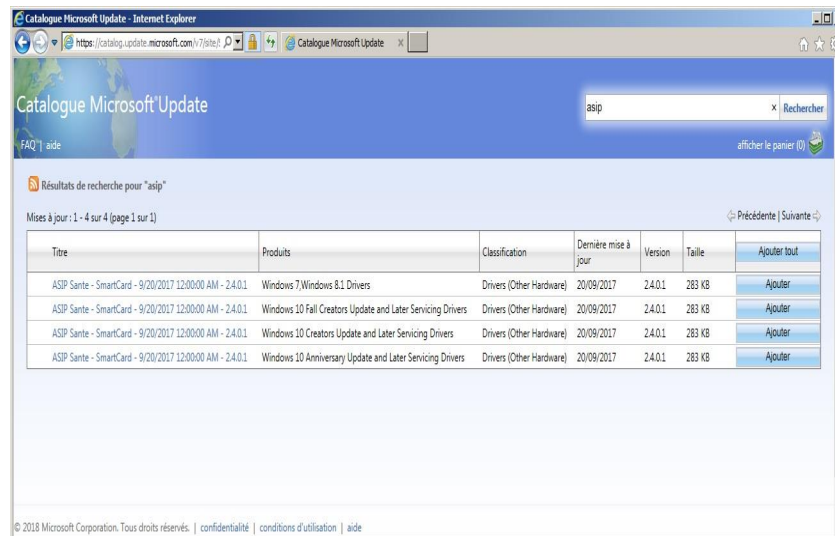


Figure 20: Windows Catalog: Résultat d'une recherche sur "asip"

Cliquez sur le lien « Afficher le panier »

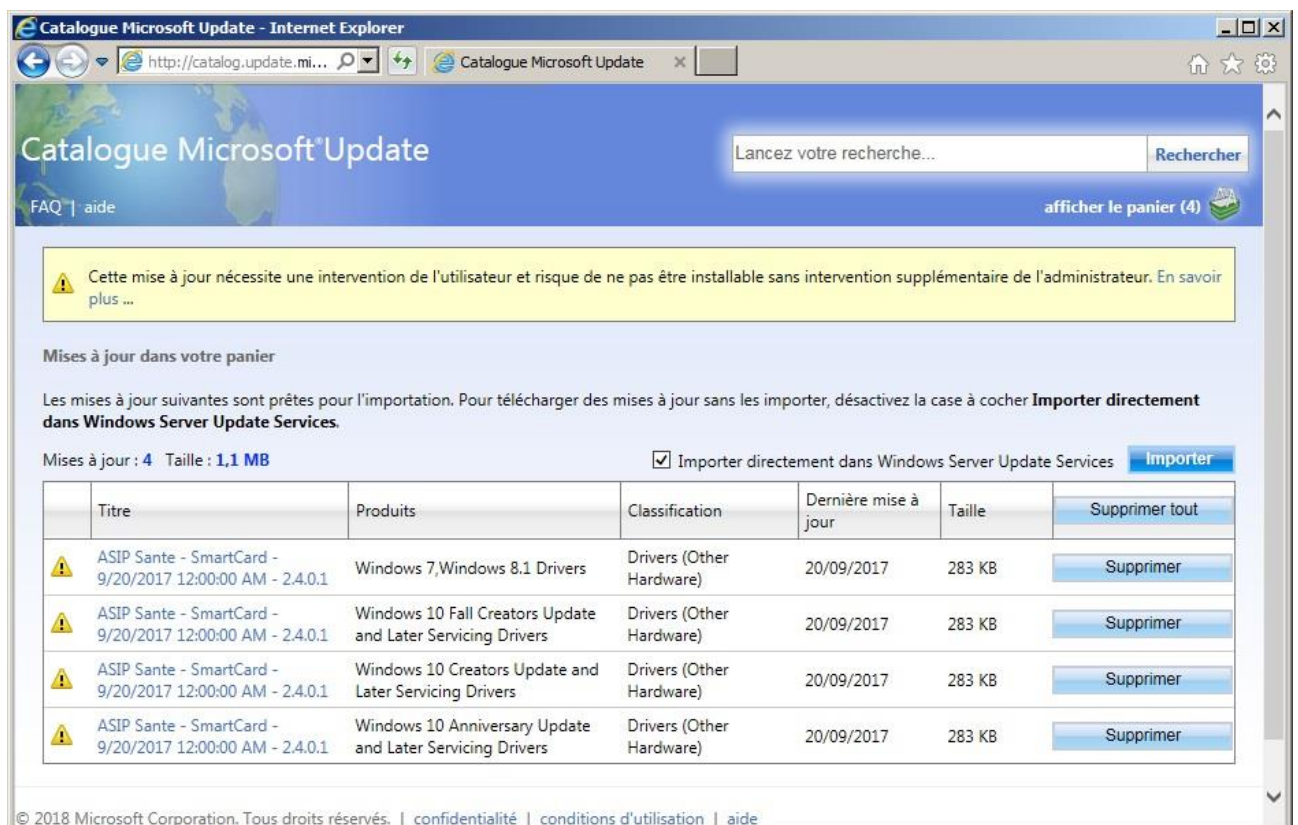


Figure 21: Windows Catalog: "afficher le panier"

Cliquer sur le bouton « Importer » afin de transférer directement les mini drivers vers le WSUS.

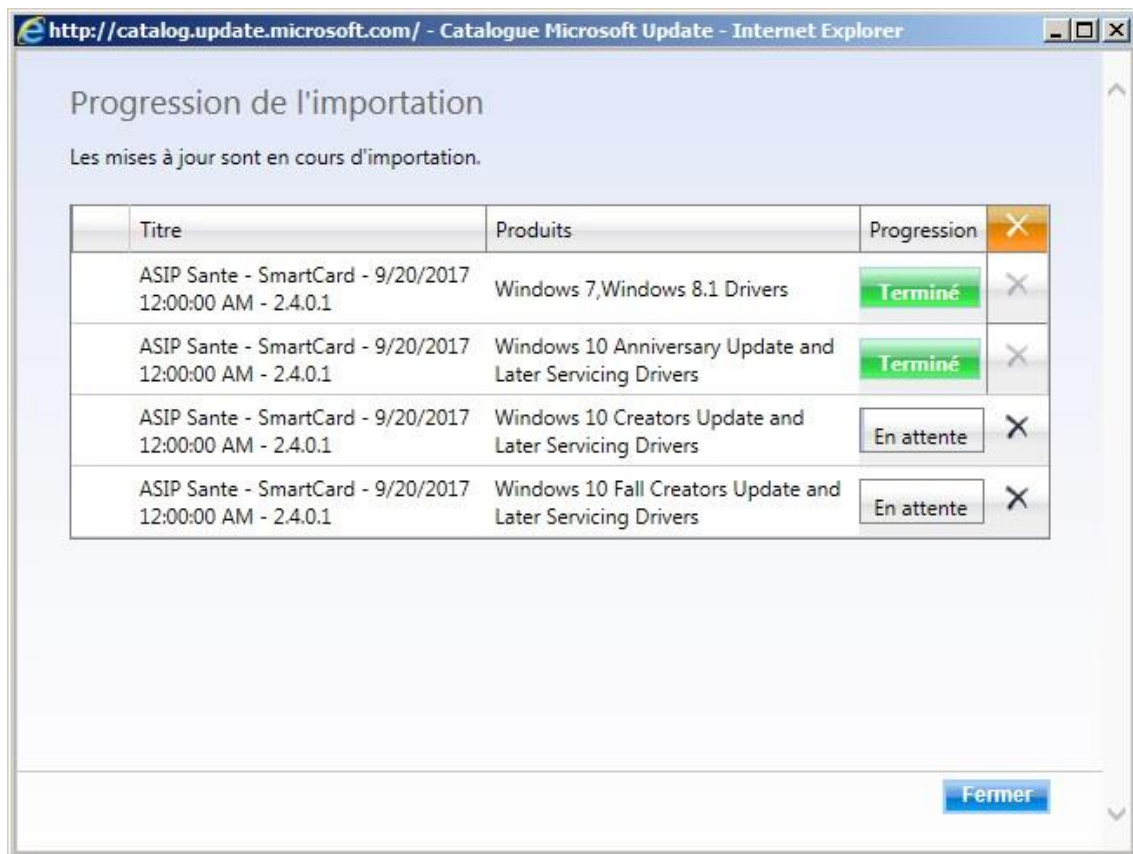


Figure 22: Windows Catalog: Importation des mini drivers vers le WSUS

8.2 Mise à jour du Minidriver CPS via Windows Update

La mise à jour de Minidriver CPS pourra s'initier manuellement depuis le "Gestionnaire de périphériques".

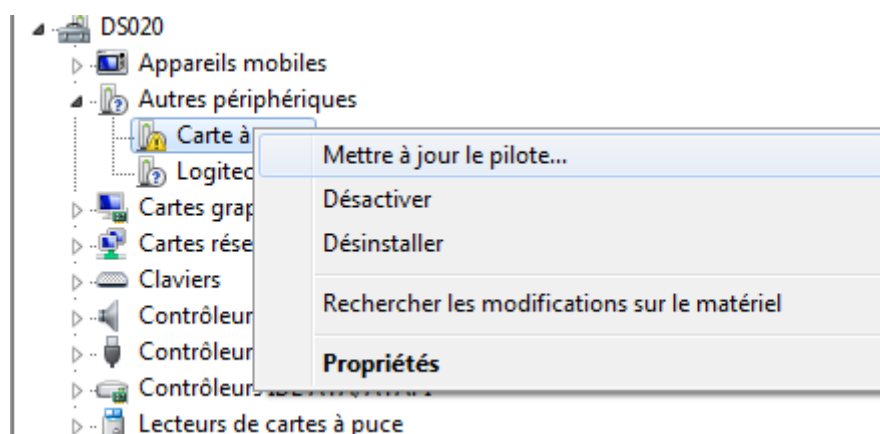


Figure 23: Minidriver: Windows 7: Initier manuellement une mise à jour de Minidriver CPS

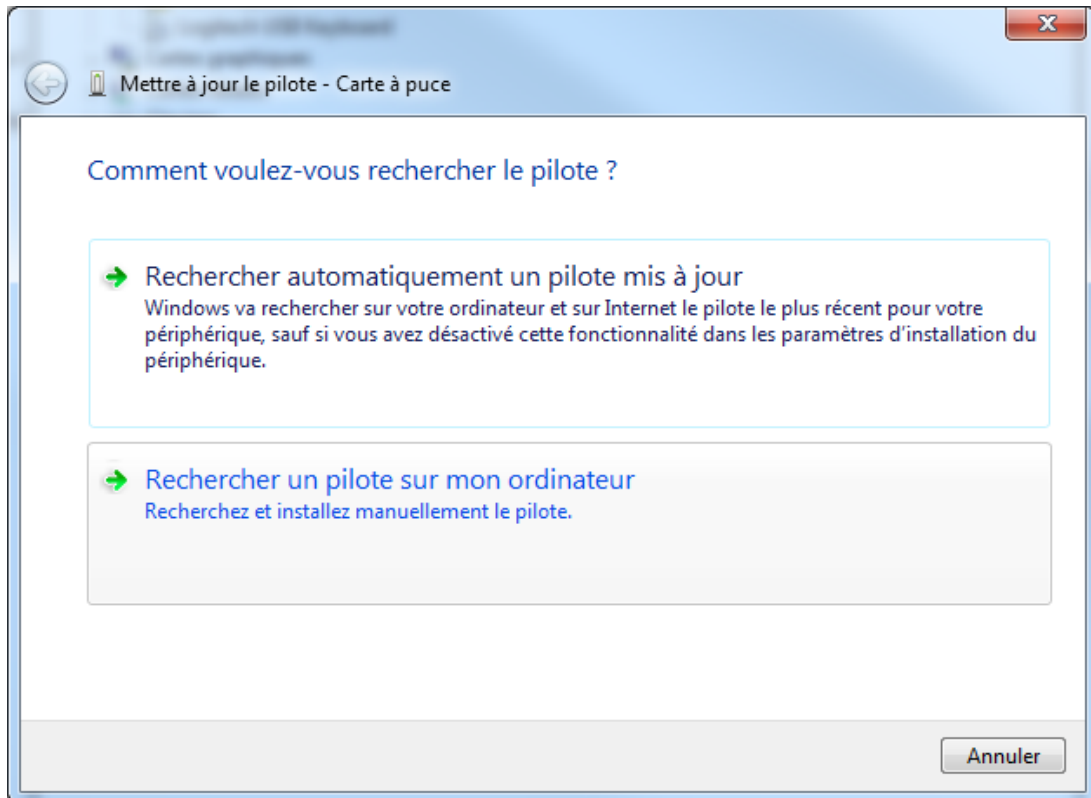


Figure 24: Minidriver: Windows 7: cliquer sur "rechercher automatiquement un pilote mis à jour"

9 Utilisation du Minidriver CPS

9.1 Vérification de la présence des certificats CPS dans le magasin de certificats

Voir le Guide d'installation et d'utilisation de la Cryptolib CPS [5].

1. Installer un lecteur de carte à puce de type PC/SC USB
2. Insérer une carte CPS dans le lecteur et installer le Minidriver CPS
3. Dans "Rechercher" > entrer la commande "inetcp1.cpl" > appuyer sur la touche "entrée" > Sélectionner l'onglet "Contenu" > Cliquer sur "Certificats" > Choisir "Personnel". 2 certificats CPS doivent être présents:

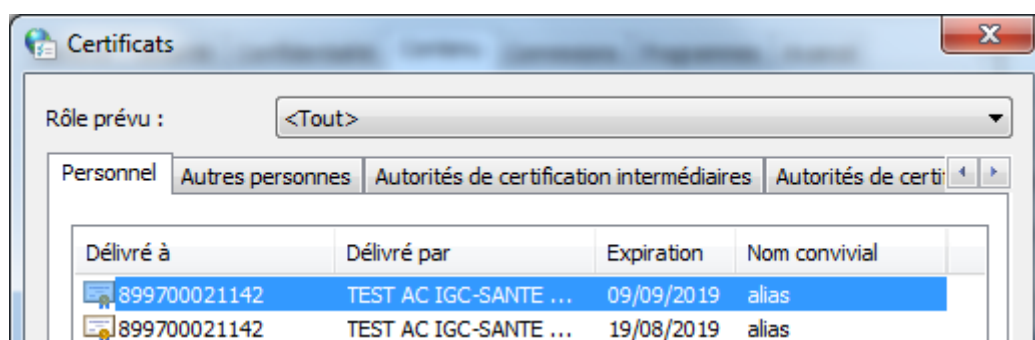


Figure 25: Minidriver: Windows 7: Vérification de bon fonctionnement du Minidriver CPS en consultant le magasin de certificats

9.2 Authentification web par carte CPS

Voir le Guide d'installation et d'utilisation de la Cryptolib CPS:

1. Installer un lecteur de carte à puce de type PC/SC USB
2. Insérer une carte CPS dans le lecteur et installer le Minidriver CPS
3. Lancer "Internet Explorer" ou "Edge" > Entrer l'URL "http://testssl.asipsante.fr" > cliquer sur le lien "https://testssl.asipsante.fr". Un tableau récapitulant les données d'identité du porteur doit apparaître.

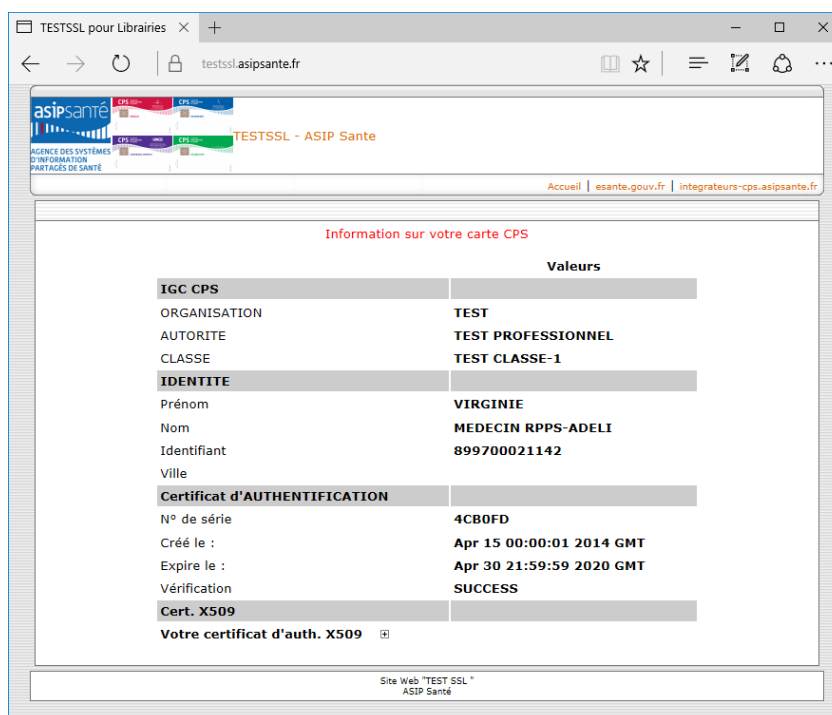


Figure 26: Minidriver: Windows 10: Vérification de bon fonctionnement du Minidriver CPS avec TestSSL

9.3 Configuration avancée du Minidriver CPS

Les paramètres de configuration gérés par le Minidriver CPS sont les suivants:

"[HKLM;HKCU]\Software\[N/A;Wow6432Node]\ASIP Sante\Minidriver"				
"Traces"	DWORD	0	INFO: 0x01 VERBOSE: 0x02 DEBUG: 0x03 PERFS: 0x04 ERROR: 0x10	Fichiers de logs créés dans %APPDATA%\santesocial\cps\log\
"Certify"	DWORD	0	0 ou toute valeur différente de 0	?
"ContainersFreshness"	DWORD	0	Valeur entière	?
"FilesFreshness"	DWORD	0	Valeur entière	?

Tableau 18: Minidriver CPS: Paramétrage avancé

9.4 Utilisation avec une application .NET "bureau classique"

Le Minidriver CPS s'utilise sous .NET dans les mêmes conditions qu'avec le CSP Cryptolib CPS: les mêmes codes exemples qu'avec la Cryptolib CPS v5 et un CSP "classique" (voir le Guide d'installation et d'utilisation de la Cryptolib CPS [5] page 171 et le guide sans-contact ASIP Santé [6]) restent valables.



Paramétrage du CSP en architecture Minidriver

En architecture Minidriver, le fournisseur cryptographique devient "**Microsoft Base Smart Card Crypto Provider**" (au lieu de "**ASIP Sante Cryptographic Provider**" en architecture Cryptolib CPS).

Le type de CSP reste **PROV_RSA_FULL (1)**, "Microsoft Base Smart Card Crypto Provider" se déclarant de type PROV_RSA_FULL (1) et non de type PROV_RSA_AES (24).

Il est toutefois tout à fait possible de faire du **SHA2** avec le Minidriver CPS.

Tableau 19 : Paramétrage du CSP en architecture Minidriver

En architecture Minidriver, le ContainerName exposé au niveau du CSP prend la forme :

802500000001_0000001234567890_AUTH:

- 802500000001 est une chaîne de caractère constante
- 0000001234567890 contient le numéro de la carte CPS sur 16 digits
- Le suffixe vaut AUTH ou SIGN en fonction du type de certificat



Impact 07: Impact du passage au Minidriver CPS: changement de ContainerName

Tableau 20 : Impact 07: Impact du passage au Minidriver CPS: changement de ContainerName

9.5 Utilisation avec une application .NET "UniversalApp"

Microsoft enrichit ses API .NET depuis Windows 8 avec WinRT.

La généralisation des applications "UniversalApp" installables sur PC, tablette et Windows Phone s'accompagne de changements d'API qui rendent l'utilisation d'un Minidriver désormais indispensable afin d'utiliser une carte à puce.

9.5.1 Modèle de déploiement d'application Universelle utilisant le Windows Store

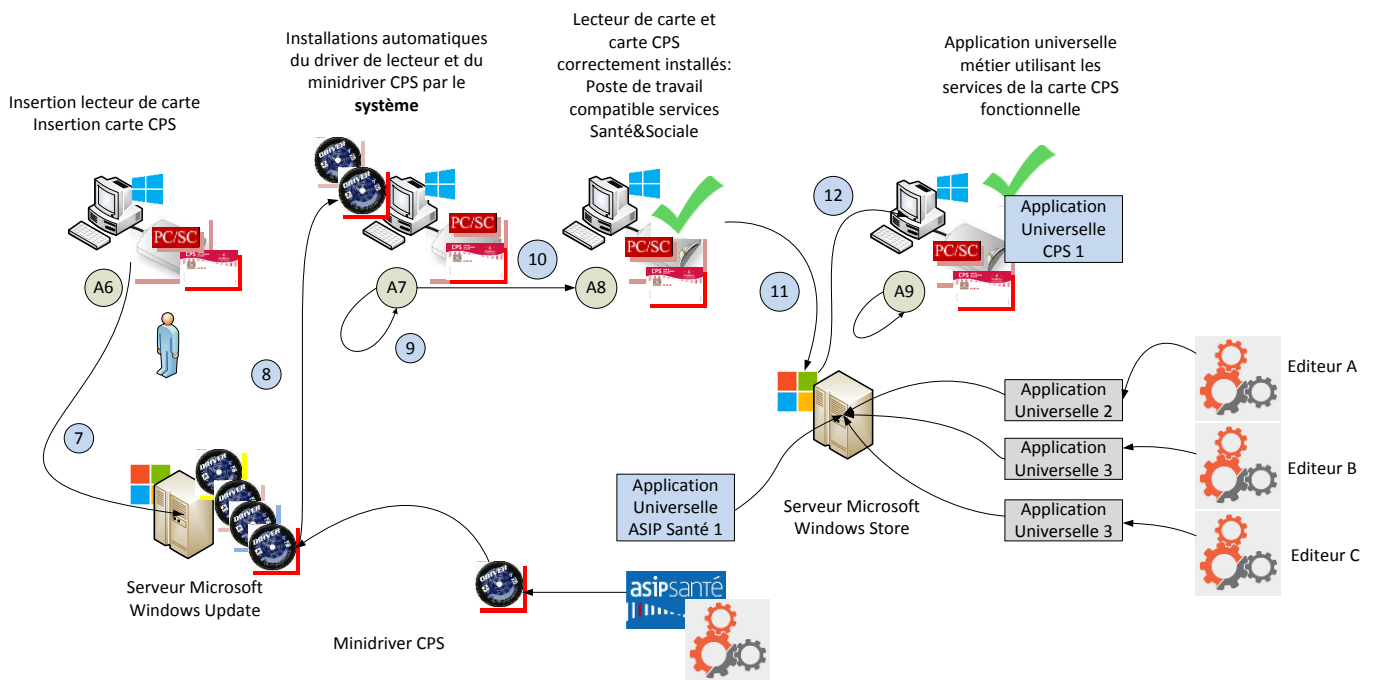


Figure 27: Windows Store: Modèle de déploiement d'applications



Sideload d'applications Windows Store

Il est possible de déployer des applications "Windows Store" ou "Universal" sur les postes d'un SI sans passer par le Store Microsoft: ce cas d'usage est couvert par Microsoft sous la terminologie "[Sideload](#)".

Tableau 21 : Sideload d'applications Windows Store

9.5.2 Wizard UniversalApp

Microsoft Visual Studio 2015 contient de nouveaux "wizards" permettant la création d'UniversalApp en C#:

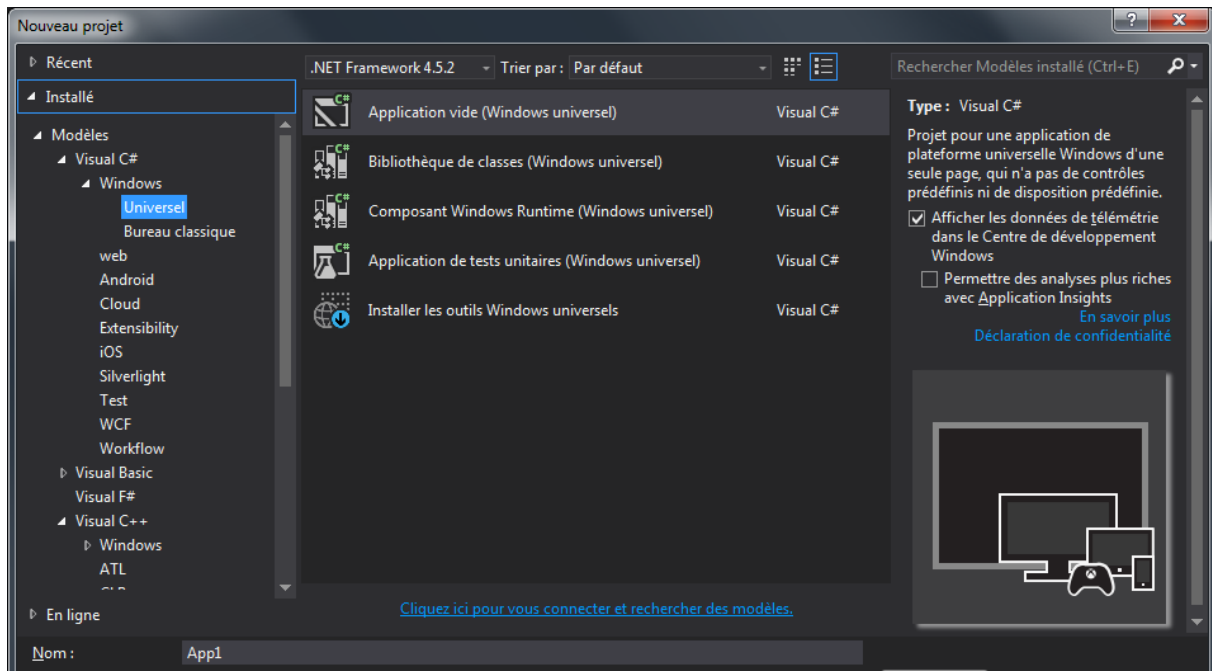


Figure 28: Visual Studio: Conception d'application Universelle

9.5.3 Capacités de l'application universelle

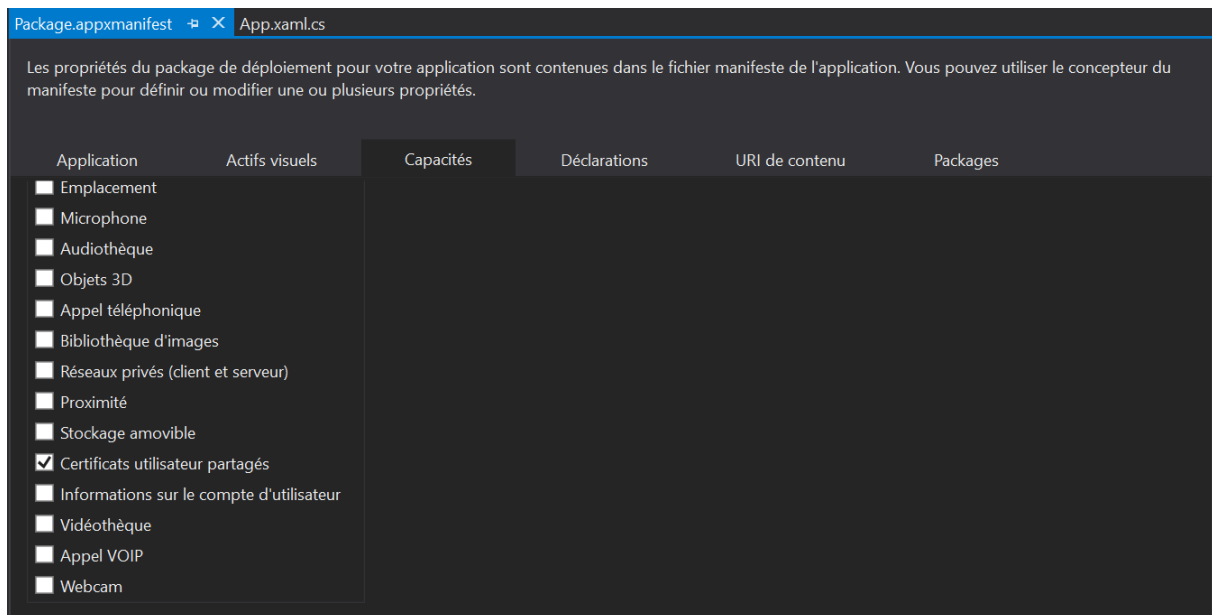


Figure 29: Visual Studio: Spécifications des capacités de l'application: cocher "Certificats utilisateur"

9.5.4 Signature numérique avec la carte CPS dans une application Universelle avec le Minidriver CPS

```
using Windows.Foundation;
using Windows.Foundation.Collections;
[...]
using System.Runtime.InteropServices.WindowsRuntime;
using Windows.ApplicationModel;
using Windows.ApplicationModel.Activation;
using Windows.Security.Cryptography;
using Windows.Security.Cryptography.Certificates; //Certificate, CertificateQuery...
using Windows.Security.Cryptography.Core;
using Windows.Storage.Streams; //IBuffer
using System.Threading.Tasks;
[...]

protected override void OnLaunched(LaunchActivatedEventArgs e) {
    [...]
    String dataToSign = [...];
    signWithCPS(dataToSign, [...]);
    [...]
}

//ASIP: String data: data to verify
private async Task<String> signWithCPS(String data, [...])
{
    try
    {
        //ASIP: find the signature certificate contained in the smartcard
        Certificate cert = await findCertificate([...]);

        //ASIP: get associated private key
        CryptographicKey keyPair = await PersistedKeyProvider.OpenKeyPairFromCertificateAsync(
            cert, HashAlgorithmNames.Sha1, CryptographicPadding.RsaPkcs1V15
        );

        //ASIP: convert the data to be signed as a Windows.Storage.Streams.IBuffer reference
        //ASIP: using Windows.Storage.Streams; required
        IBuffer Data = CryptographicBuffer.ConvertStringToBinary(data, BinaryStringEncoding.Utf16BE);

        //ASIP: sign the data by using the key
        IBuffer Signature = await CryptographicEngine.SignAsync(keyPair, Data);

        //ASIP: build signature hexstring
        //ASIP: CryptographicBuffer has an EncodeToHexString method, not tested here
        byte[] ba = Signature.ToArray();
        String signature = BitConverter.ToString(ba).Replace("-", "");

        return signature;
    }
    catch (Exception e)
    {
        //ASIP: gestion des exceptions :
        [...]
    }
    return null;
}
```

Figure 30: Visual Studio: Signature numérique avec la carte CPS dans une application Universelle avec le Minidriver CPS

9.5.5 Vérification de signature numérique

```
//ASIP: String data: data to verify
//ASIP: String signature: hexstring containing the signature
private async Task<bool> verifySignature(String data, String signature, [...])
{
    bool ret = false;
    try
    {
        //ASIP: signer certificate should have been imported
        //ASIP: somewhere within the Windows certificate stores
        Certificate cert = await findCertificate([...]);

        //ASIP: get keyPair reference from certificate
        //ASIP: private key presence is not required to verify a signature
        //ASIP: so, use OpenPublicKeyFromCertificate method
        CryptographicKey keyPair = PersistedKeyProvider.OpenPublicKeyFromCertificate(
            cert, HashAlgorithmNames.Sha1, CryptographicPadding.RsaPkcs1V15
        );

        IBuffer Data = CryptographicBuffer.ConvertStringToBinary(data, BinaryStringEncoding.Utf16BE);
        IBuffer Signature = CryptographicBuffer.CreateFromByteArray(StringToByteArray(signature));

        //ASIP: verify the signature
        ret = internalVerifySignature(keyPair, Data, Signature);
    }
    catch (Exception e)
    {
        //ASIP: gestion des exceptions :
        [...]
    }
    return ret;
}

private static bool internalVerifySignature(CryptographicKey keyPair, IBuffer Data, IBuffer Signature)
{
    bool ret = false;
    try
    {
        ret = CryptographicEngine.VerifySignature(keyPair, Data, Signature);
        [...]
        //ASIP: more operation based over bresult analysis if required
        [...]
    }
    catch (Exception e)
    {
        //ASIP: gestion des exceptions :
        [...]
        ret = false;
    }
    return ret;
}
```

Figure 34: Visual Studio: Vérification de signature numérique avec la carte CPS dans une application Universelle avec le Minidriver CPS

9.5.6 Recherche de certificat

```

String storeName = "MY";
String issuerName = "TEST CLASSE-1";

[...]

private static async Task<Certificate> findCertificate(string storeName, string issuerName, bool
signatureCertificate)
{
    Certificate cert = null;
    CertificateQuery cq = new CertificateQuery();
    cq.IssuerName = issuerName;
    cq.StoreName = storeName;

    //ASIP: no way to query over the "simple" KeyUsage
    //ASIP: use EnhancedKeyUsages instead
    IList<string> eku = cq.EnhancedKeyUsages;

    if (signatureCertificate) {
        //ASIP: to retrieve signature certificate
        eku.Add("1.3.6.1.5.5.7.3.4"); // Messagerie électronique sécurisée
    } else {
        //ASIP: to retrieve authentication certificate
        eku.Add("1.3.6.1.5.5.7.3.2"); // Authentification du client
        eku.Add("1.3.6.1.4.1.311.20.2.2"); // Ouverture de session par carte à puce
    }

    IEnumerable<Certificate> certificates = await CertificateStores.FindAllAsync(cq);
    foreach (Certificate value in certificates)
    {
        cert = value;
    }
    //ASIP: return the last certificate retrieved
    //ASIP: this should be challenged regarding the expected application behavior!!
    return cert;
}

```

Figure 34: Visual Studio: Exemple de recherche de certificat CPS dans une application Universelle avec le Minidriver CPS

La logique de recherche de certificat CPS sur le poste de travail doit être affinée au cas par cas.

Le **service de propagation de certificats** carte Microsoft **ne retire pas** les certificats du magasin de certificat Windows **lors du retrait** de la carte du lecteur: sans action additionnelle, une machine peut contenir d'innombrables certificats CPS sans que les cartes CPS associées soient présentes.

Ceci impacte particulièrement les applications protégées par CPS et accessibles en **mode kiosque ou en TSE/Citrix**, pour lesquelles il sera, en l'état, nécessaire de trouver des solutions ad-hoc.

La gestion des cas d'erreurs doit faire l'objet d'une attention particulière.



Impact 08: Recherche de certificat CPS sur le poste

Tableau 22 : Impact 08: Recherche de certificat CPS sur le poste

9.5.7 Récupération logicielle du contenu d'une page Web protégée par authentification mutuelle par carte CPS

```
using Windows.Web.Http;
using Windows.Web;
using Windows.Web.Http.Filters;

[...]

//ASIP: reference documentation is
//https://blogs.msdn.microsoft.com/dotnet/2015/07/28/net-networking-apis-for-uwp-apps/
//https://blogs.windows.com/buildingapps/2015/11/23/demystifying-httpclient-apis-in-the-universal-windows-
platform/#1eSg5r6ZGJpYGcy.97

private async Task<bool> makeWebHttpRequest(String urlParam)
{
    bool ret = false;
    String storeName = "MY";
    String issuerName = "TEST CLASSE-1";
    //ASIP: retrieve an authentication certificate:
    Certificate cert = await findCertificate(storeName, issuerName, false);

    if (cert != null)
    {
        HttpBaseProtocolFilter myFilter = new HttpBaseProtocolFilter();
        myFilter.ClientCertificate = cert;
        HttpClient client = new HttpClient(myFilter);
        try
        {
            Uri uri = new Uri(urlParam);
            HttpResponseMessage response = await client.GetAsync(uri);
            if (response.IsSuccessStatusCode)
            {
                String content = await response.Content.ReadAsStringAsync();
                ret = true;
            }
            else
            {
                //ASIP: handle error, see below
                // [...]
            }
        }
        catch (Exception e)
        {
            WebErrorStatus error = WebError.GetStatus(e.HResult);
            //ASIP: typical errors to handle are:
            handleError(error);
            // [...]
        }
        client.Dispose();
    }
    else
    {
        //ASIP: no authentication certificate found
        //ASIP: handle this situation:
        // [...]
    }
    return ret;
}
```

Figure 34: Visual Studio: Exemple d'authentification mutuelle HTTPS logicielle par carte CPS avec le Minidriver CPS

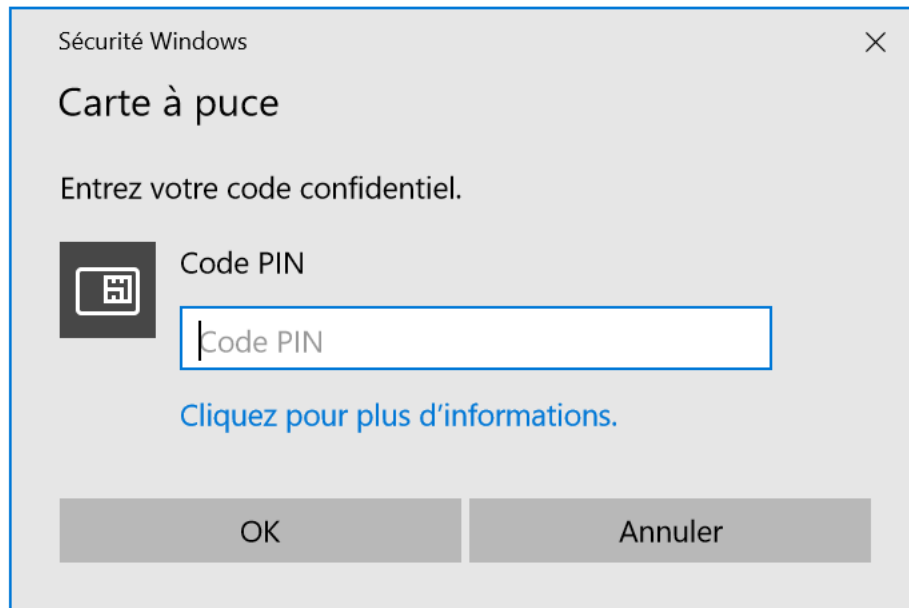


Figure 34: Visual Studio: Exemple d'authentification mutuelle HTTPS logicielle par carte CPS: la boîte de dialogue de saisie de code porteur est automatiquement affichée par Windows

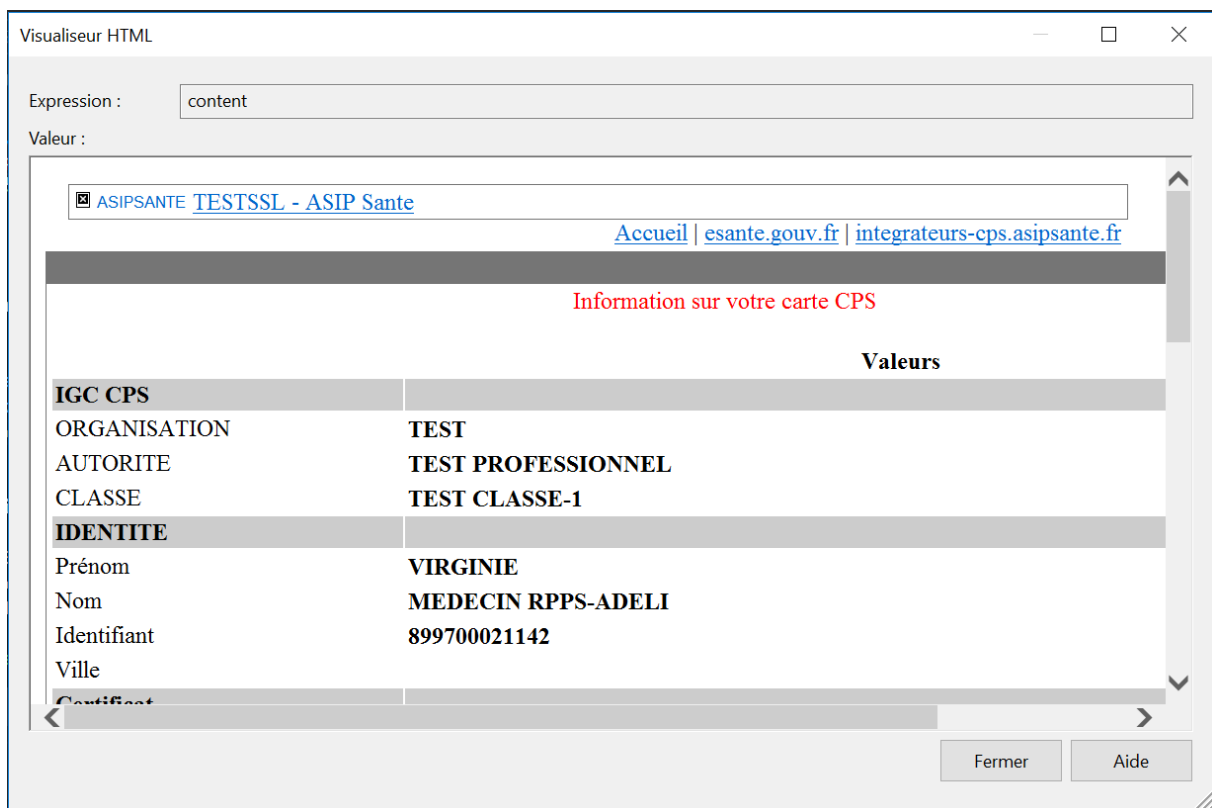


Figure 34: Visual Studio: Exemple d'authentification mutuelle HTTPS logicielle par carte CPS: les variables internes de l'application peuvent recevoir les données protégées par CPS

```
private static void handleError(WebErrorStatus error)
{
    switch (error)
    {
        case WebErrorStatus.CannotConnect:
            //ASIP: handle error
            //ASIP: no internet cable
            //ASIP: or: CPS was previously inserted
            //ASIP: but no CPS card has been inserted in the reader at this moment
            //ASIP: and user click "cancel" while he has benen asked for card insertion
            break;
        case WebErrorStatus.HostNameNotResolved:
            //ASIP: handle error
            break;
        case WebErrorStatus.ConnectionReset:
            //ASIP: handle error
            break;
        case WebErrorStatus.ConnectionAborted:
            //ASIP: handle error
            break;
        case WebErrorStatus.Disconnected:
            //ASIP: handle error
            break;
        case WebErrorStatus.Forbidden:
            //ASIP: handle error
            break;
        case WebErrorStatus.CertificateExpired:
            //ASIP: IIS HTTP subcodes required server-side?
            //ASIP: handle error
            break;
        case WebErrorStatus.CertificateRevoked:
            //ASIP: IIS HTTP subcodes required server-side?
            //ASIP: handle error
            break;
        case WebErrorStatus.ServiceUnavailable:
            //ASIP: handle error
            break;
        case WebErrorStatus.Unknown:
            //ASIP: no authentication certificate in certstore
            //ASIP: handle error
            break;
        //case WebErrorStatus.[many other errors to handle correctly]:
        //ASIP: handle error
        //break;
        default:
            break;
    }
    // [...]
}
```

Figure 34: Visual Studio: Exemple d'authentification mutuelle HTTPS logicielle par carte CPS: gestion des erreurs



Nouvelles API .NET

Les packages `Windows.Security.Cryptography`, `Windows.Web` et `Windows.Net` sont un bon exemple de refontes des API .NET introduites par Microsoft récemment (voir les 2 URLs données dans le code source plus haut pour le détail sur les API Web).

Tableau 23 : Nouvelles API .NET

9.5.8 Récupération des données métier CPS via le Minidriver CPS

La récupération des données "métier santé" incluses dans la carte CPS n'est pas préconisée.

La préconisation générale d'utilisation de la carte CPS est de s'en servir en tant que "token" cryptographique pour les seuls usages d'authentification et de signature et non comme source de données de santé.



Recommandations d'usage du Minidriver CPS

Dans un tel cas d'usage, les données de situations d'exercice, les Données Assurance Maladie... ne devraient pas être lues dans la carte CPS mais retrouvées dans les annuaires nationaux sur la seule base de l'IdNatPS (Identifiant National du Porteur CPS) récupéré dans les certificats X509 de la CPS.

Tableau 24 : Recommandations d'usage du Minidriver CPS

Les API CAPI2 ne permettent pas de récupérer les données métiers portées par la carte CPS.

Il est sans doute possible de récupérer ces données en utilisant les mécanismes classiques `pMDCPSDII = LoadLibrary((LPCWSTR)"cpsmd.dll")` puis `GetProcAddress(pMDCPSDII, "CardGetProperty");`

Le gabarit de la fonction `CardGetProperty` est (Cf. Spécifications Minidriver Microsoft [7]):

```

DWORD WINAPI CardGetProperty(
    __in          PCARD_DATA pCardData,
    __in          LPCWSTR  wszProperty,
    __out_bcount_part_opt(cbData, *pdwDataLen) PBYTE  pbData,
    __in          DWORD    cbData,
    __out         PDWORD   pdwDataLen,
    __in          DWORD    dwFlags
)

```

Tableau 25 : Données métier via le Minidriver CPS: `LoadLibrary`, `GetProcAddress` et `CardGetProperty`

Les propriétés exposées par le Minidriver CPS sont les suivantes:

Alias de la propriété	Valeur de la chaîne de caractères	Origine du paramètre	Conseil d'utilisation
CP_CARD_FREE_SPACE	FreeSpace	Microsoft	Ne pas utiliser
CP_CARD_CAPABILITIES	Capabilities	Microsoft	Ne pas utiliser
CP_CARD_KEYSIZES	KeySizes	Microsoft	Ne pas utiliser
CP_CARD_READ_ONLY	ReadOnlyMode	Microsoft	Ne pas utiliser
CP_CARD_CACHE_MODE	CacheMode	Microsoft	Ne pas utiliser
CP_SUPPORTS_WIN_X509_ENROLLMENT	SupportsWindowsx.509Enrollment	Microsoft	Ne pas utiliser
CP_CARD_GUID	CardIdentifier	Microsoft	Ne pas utiliser
CP_CARD_SERIAL_NO	CardSerialNumber	Microsoft	Ne pas utiliser
CP_CARD_PIN_INFO	PINInformation	Microsoft	Ne pas utiliser
CP_CARD_LIST_PINS	PINList	Microsoft	Ne pas utiliser
CP_CARD_AUTHENTICATED_STATE	AuthenticatedState	Microsoft	Ne pas utiliser
CP_CARD_PIN_STRENGTH_VERIFY	PINStrengthVerify	Microsoft	Ne pas utiliser
CP_CARD_PIN_STRENGTH_CHANGE	PINStrengthChange	Microsoft	Ne pas utiliser
CP_CARD_PIN_STRENGTH_UNBLOCK	PINStrengthUnblock	Microsoft	Ne pas utiliser
CP_KEY_IMPORT_SUPPORT	KeyImportSupport	Microsoft	Ne pas utiliser
CP_ENUM_ALGORITHMS	Algorithms	Microsoft	Ne pas utiliser
CP_PADDING_SCHEMES	PaddingSchemes	Microsoft	Ne pas utiliser
CP_CARD_CPS_VERSION	CPSVersion	ASIP Santé	Ne pas utiliser
CP_CARD_CPS_EFDATA	CPSEFData	ASIP Santé	Ne pas utiliser
CP_CARD_CPS_EFLANG	CPSEFLang	ASIP Santé	Ne pas utiliser
CP_CARD_CPS_PERSO_VERSION	CPSPersoVersion	ASIP Santé	Ne pas utiliser
CP_CARD_CPS_CPLC	CPSCPLC	ASIP Santé	Ne pas utiliser

Tableau 26 : Données métier via le Minidriver CPS: Propriétés exposées par le Minidriver CPS

10Annexe - Gestion du minidriver en ligne de commande

10.1 Installation

Dans "Rechercher" > entrer "cmd.exe" > clic-droit sur le résultat "cmd.exe" > choisir "run as admin" > entrer la commande "Pnputil.exe -i -a [PATH_TO_INF]" et l'exécuter.

10.2 Désinstallation

- Dans "Rechercher" > entrer "cmd.exe" > clic-droit sur le résultat "cmd.exe" > choisir "run as admin" > entrer la commande "pnputil.exe -e"
- Recherche Oem#.inf où # est le numéro du Minidriver CPS signé par l'ASIP Santé
- Entrer la commande "Pnputil.exe -d Oem#.inf" où # est le numéro du Minidriver CPS signé par l'ASIP Santé

11Annexe - Message d'erreur suite à une insertion de carte CPS dans un lecteur PC/SC en l'absence de Minidriver ou de CSP

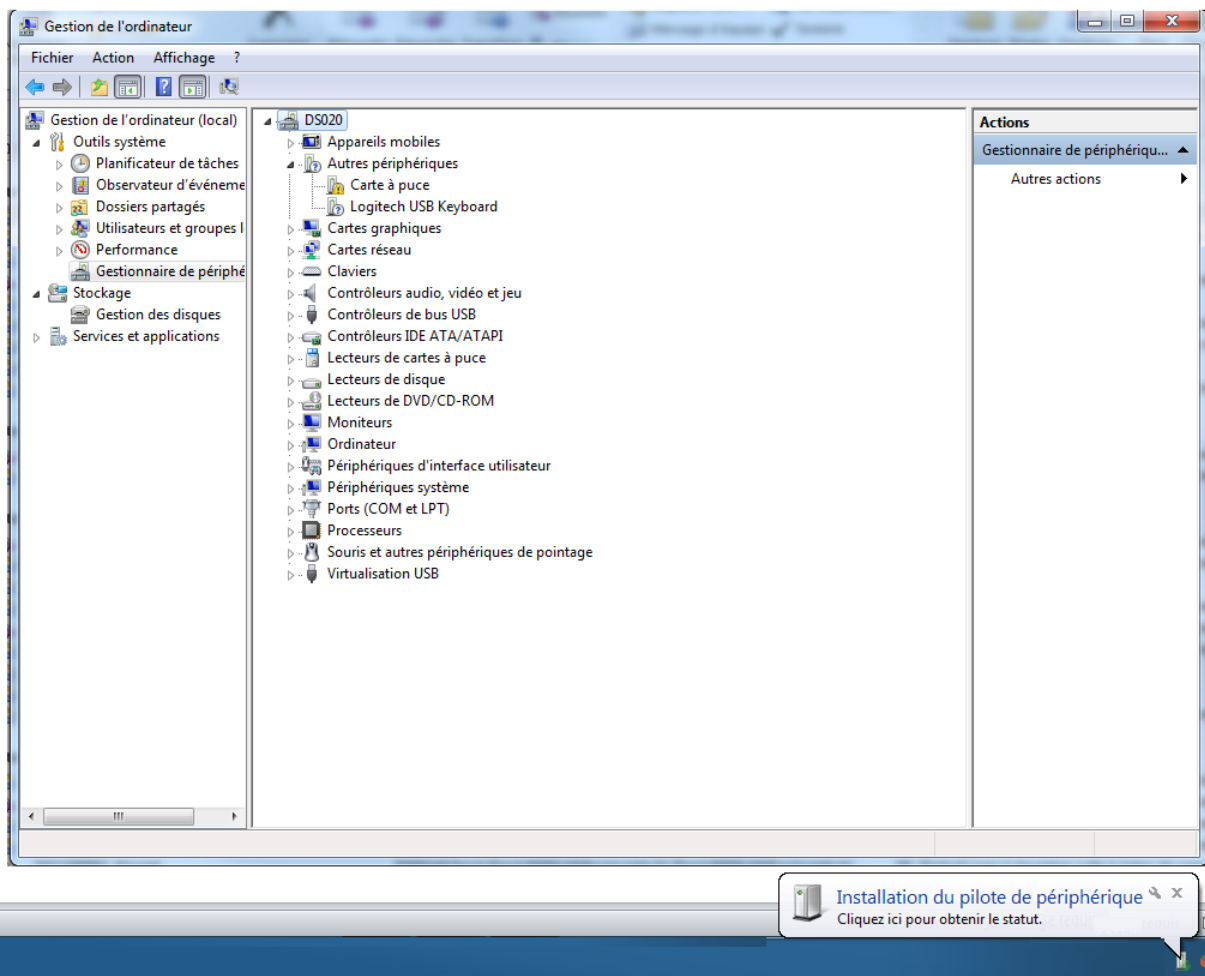


Figure 31: Insertion d'une carte à puce non reconnue par Windows dans un lecteur PC/SC

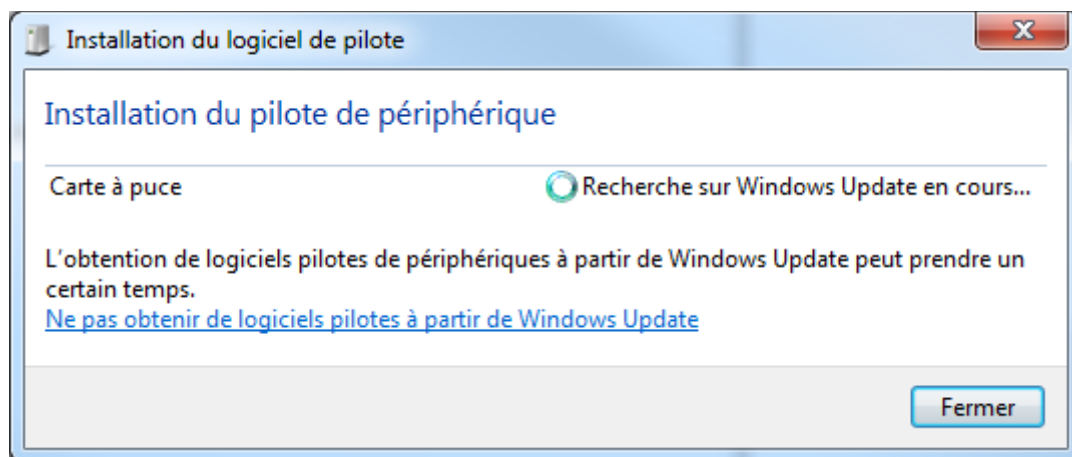


Figure 32: Insertion d'une carte à puce non reconnue par Windows dans un lecteur PC/SC: message de recherche de pilote pour la carte à puce

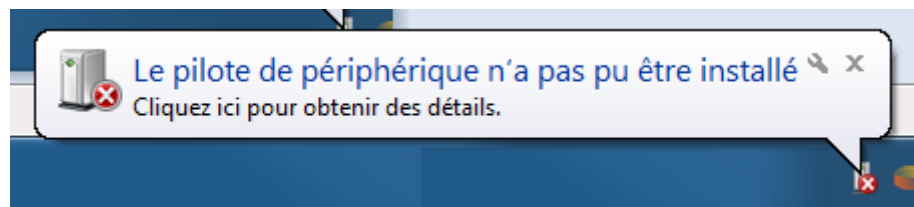


Figure 33: Insertion d'une carte à puce non reconnue par Windows dans un lecteur PC/SC: message d'erreur



Figure 34: Insertion d'une carte à puce non reconnue par Windows dans un lecteur PC/SC: message d'erreur

12Annexe – Problèmes connus

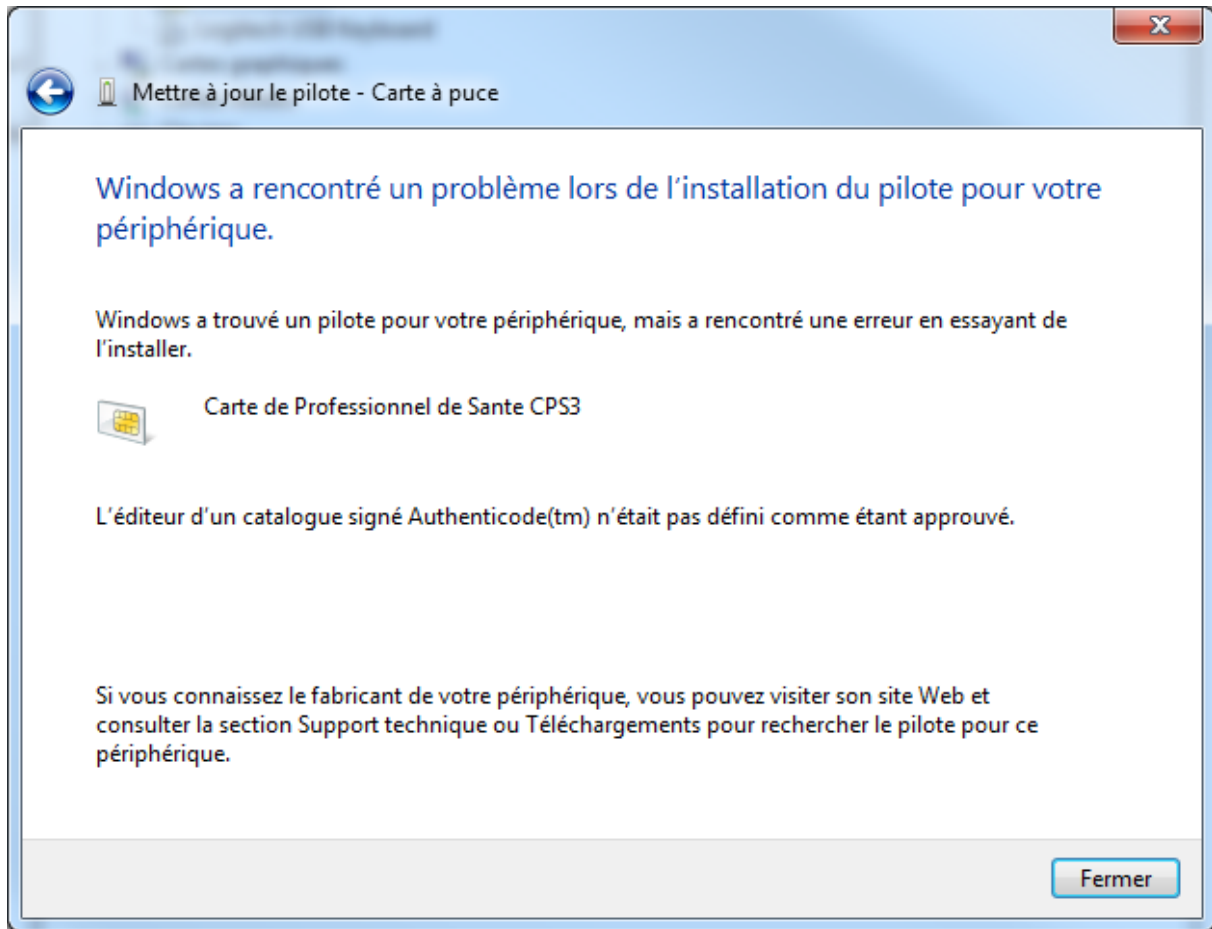


Figure 35: Problème d'installation de Minidriver CPS

Dans ce cas de figure, désinstaller le Minidriver CPS encore présent dans le système:

- Rechercher > cmd.exe > clic droit > "run as admin" > pnputil.exe -e
- Pnputil.exe -d Oem#.inf où # est le numéro du Minidriver CPS signé par l'ASIP Santé
- Ré-insérer la carte CPS et ré-installer le Minidriver CPS

13Annexe – Liste des figures

Figure 1: Architecture classique de la Cryptolib CPS sous Windows	10
Figure 2: Architecture de la Cryptolib CPS sous Windows depuis Windows Vista	11
Figure 3: Différences d'appels applicatifs depuis l'introduction de l'architecture CNG par Microsoft	12
Figure 4: Architecture de la Minidriver CPS cible.....	13
Figure 5: Déploiement: Principe général de déploiement de "pilotes" ("drivers")	15
Figure 6: Déploiement: Répétition des installations de pilotes pour chaque périphérique.....	16
Figure 7: Déploiement: Déploiement de "pilotes" ("drivers") via l'OS	16
Figure 8: Déploiement: Déploiement de "pilotes" ("drivers") via Windows Update.....	17
Figure 9: Déploiement: Déploiement de "pilotes" ("drivers") via Windows Update appliqué à la CPS	18
Figure 10: Windows 7: Installation de lecteur: message affiché lors du branchement du lecteur	19
Figure 11: Windows 7: Installation de lecteur: message affiché pendant l'installation des pilotes du lecteur	19
Figure 12: Windows 7: Installation de lecteur: icône affiché pendant l'installation des pilotes du lecteur....	19
Figure 13: Windows 7: Installation de lecteur: message affiché en fin d'installation des pilotes du lecteur..	19
Figure 14: Windows 10: Installation de la carte CPS: message affiché lors de l'insertion de la carte	20
Figure 15: Windows 10: Installation de la carte CPS: message affiché en fin d'installation du Minidriver CPS	20
Figure 16: Windows Catalog: Page d'accueil.....	21
Figure 17: Windows Catalog: Résultat d'une recherche sur "asip".....	22
Figure 18: Windows Catalog: "afficher le panier"	22
Figure 19: Windows Catalog: Page d'accueil.....	24
Figure 20: Windows Catalog: Résultat d'une recherche sur "asip".....	25
Figure 21: Windows Catalog: "afficher le panier"	25
Figure 22: Windows Catalog: Importation des mini drivers vers le WSUS	26
Figure 23: Minidriver: Windows 7: Initier manuellement une mise à jour de Minidriver CPS	26
Figure 24: Minidriver: Windows 7: cliquer sur "rechercher automatiquement un pilote mis à jour".....	27
Figure 25: Minidriver: Windows 7: Vérification de bon fonctionnement du Minidriver CPS en consultant le magasin de certificats.....	28
Figure 26: Minidriver: Windows 10: Vérification de bon fonctionnement du Minidriver CPS avec TestSSL...29	
Figure 27: Windows Store: Modèle de déploiement d'applications.....	30
Figure 28: Visual Studio: Conception d'application Universelle.....	31
Figure 29: Visual Studio: Spécifications des capacités de l'application: cocher "Certificats utilisateur"	31
Figure 30: Visual Studio: Signature numérique avec la carte CPS dans une application Universelle avec le Minidriver CPS.....	32
Figure 31: Insertion d'une carte à puce non reconnue par Windows dans un lecteur PC/SC	40

Figure 32: Insertion d'une carte à puce non reconnue par Windows dans un lecteur PC/SC: message de recherche de pilote pour la carte à puce	41
Figure 33: Insertion d'une carte à puce non reconnue par Windows dans un lecteur PC/SC: message d'erreur.....	41
Figure 34: Insertion d'une carte à puce non reconnue par Windows dans un lecteur PC/SC: message d'erreur.....	41
Figure 35: Problème d'installation de Minidriver CPS.....	42

14Annexe – Liste des tableaux

Tableau 1 : Références	3
Tableau 2 : contact accompagnement ASIP Santé.....	5
Tableau 3 : Glossaire	7
Tableau 4 : Entreprises citées.....	8
Tableau 5 : Avertissements	9
Tableau 6 : Sous Windows, les cartes à puce sont des périphériques parmi tant d'autres.....	10
Tableau 7 : Impact 00: Minidriver uniquement sous Windows	11
Tableau 8 : Impact 01: Changement d'API pour accéder aux capacités de la carte CPS.....	12
Tableau 9 : Impact 02: nécessité de s'adapter à l'architecture Minidriver	12
Tableau 10 : Impact 03: DLL cps3_pkcs11_wXX.dll et Minidriver CPS.....	13
Tableau 11 : Impact 04: Intérêt du Minidriver CPS pour Santé&Social	14
Tableau 12 : Impact 05: Pas de GALSS / PSS avec le Minidriver CPS.....	14
Tableau 13 : Impact 06: Mozilla Firefox ne fonctionne pas avec le Minidriver CPS	14
Tableau 14 : Sous Windows, 2 périphériques et leurs drivers à installer: lecteur de carte et CPS.....	15
Tableau 15 : Résumé des cas d'usage permis par le Minidriver CPS.	20
Tableau 16 : Windows Catalog: Choix du répertoire de destination	23
Tableau 17 : Windows Catalog: Progression du téléchargement	23
Tableau 18: Minidriver CPS: Paramétrage avancé	29
Tableau 19 : Paramétrage du CSP en architecture Minidriver.....	29
Tableau 20 : Impact 07: Impact du passage au Minidriver CPS: changement de ContainerName.....	29
Tableau 21 : Sideload d'applications Windows Store	30
Tableau 22 : Impact 08: Recherche de certificat CPS sur le poste	34
Tableau 23 : Nouvelles API .NET.....	37
Tableau 24 : Recommandations d'usage du Minidriver CPS.....	38
Tableau 25 : Données métier via le Minidriver CPS: LoadLibrary, GetProcAddress et CardGetProperty.....	38
Tableau 26 : Données métier via le Minidriver CPS: Propriétés exposées par le Minidriver CPS.....	39

15Notes

[fin du document]



Agence des systèmes d'information partagés de santé
9, rue Georges Pitard - 75015 Paris
Tel : 01 58 45 32 50
esante.gouv.fr