DIFFUSION RESTREINTE

SPECIAL FRANCE

AVANTIX

Architecture logicielle spécifique à la CIEEMG

FLASHHAWK

C-ESM Aéroporté

AVANTIX

REFERENCE	:	DP068897DAL001
REVISION	:	11
DATE	:	04/03/2022

DIFFUSION RESTREINTE

SPECIAL FRANCE

AVANTIX

VISA

Préparé par : ATOS AVANTIX 04/03/2021

CIRCULATION

Diffusion Interne :

Diffusion Externe :

CONTACTS

Technique : Marc Houry

REGISTRE DES REVISIONS

1	14/01/2021	Tous	Création Document
2	18/01/2021		Revue avec architecte
3	03/02/2021		Revue avec chef de projet
4	13/07/2021		1ère revue avec DGA
5	11/08/2021		Maintenance (1, 5, 6.1, 8) Chiffrage embarqué (6.1, 6.2) Couplage FPGA (6.3) Outils de vérification (9)
6	28/09/2021		Couverture Wibu (6.1) Sécurisation mots de passe (6.2) Couplage hors FPGA (6.3)
7	04/10/2021		Nombre de voies inutile (4) Mots de passe (3, 5) Sécurisation bootloader (6.2) Sécurisation démarrage (6.2) Empreinte et release notes (9)
8	14/10/2021		Config. sécurisée France (3)
9	18/01/2022		Interfaces externes (4, 5, 6, 10.4) Précision génération (12)
10	01/02/2022		Chargement bibliothèque d'option (4, 5) Protection bibliothèque d'option (10.1) Protection contenu dossier .ssh (10.2) Détail chiffrement partiel calibration (14.2)
11	04/03/2022		Composants logiciels tiers (9) Sécurisation serveurs de maintenance (11.2, 11.6) Effacement logs à l'export (14) Formulation inconditionnelle (10, 11.1, 11.2, 11.4, 13, 15.5)

AVANTIX

SOMMAIRE

1 DOCUMENTS DE REFERENCE

[R1] : « Document technique à l'usage de la CIEEMG FlashHawk » réf. DV18PTE014 Ind. 10

[R2] : « FlashHawk software interface » réf. DP068897STI001 Ind. 02

[R3] : « Plan de validation CIEEMG FlashHawk » réf. DP068897PVL001 Ind. 06

2 GLOSSAIRE ET ABREVIATIONS

Dongle	Matériel amovible (USB ou carte SD) qui propose des fonctions de protection du logiciel contre la copie et la modification.
FTP	Serveur de fichiers permettant d'échanger facilement des fichiers, une fois authentifié.
gcc	Compilateur C++ utilisé pour notre cible Linux (capteur embarqué).
iss	Extension des fichiers de définition à écrire pour que le logiciel InnoSetup puisse générer nos installeurs (setup) sur Windows.
Makefile	Fichier de définition à écrire pour que le logiciel « make » puise exécuter les tâches de compilation et de génération des applications et autres éléments associés.
Splash screen	Panneau affiché brièvement pendant le démarrage de l'application du poste de commande. C'est l'occasion de présenter des éléments marketing (logo du produit) et le numéro de version du poste de commande.
Visual Studio	Plateforme de développement utilisée pour la compilation et la génération de nos applications Windows (poste de commande et serveur de calculs).

3 INTRODUCTION

Ce document décrit l'architecture logicielle spécifique à la CIEEMG du Produit FLASHHAWK C-ESM Aéroporté.

Du logiciel applicatif FLASHHAWK est présent dans les équipements suivants :

- Capteur embarqué (Linux)

- Unité serveur de calculs (Windows, à bord ou au sol en fonction de la configuration)

- Poste de commande / IHM (Windows, à bord ou au sol en fonction de la configuration)

- Poste de maintenance (Windows, limité à l'équipe de R&D et de maintenance)

Dans un cadre opérationnel, FlashHawk peut être intégré avec :

- Option A : console d'exploitation

(par exemple Airbus Fortion® SAMSARA®)

- Option B : démodulateur de flux I/Q Decodio RED

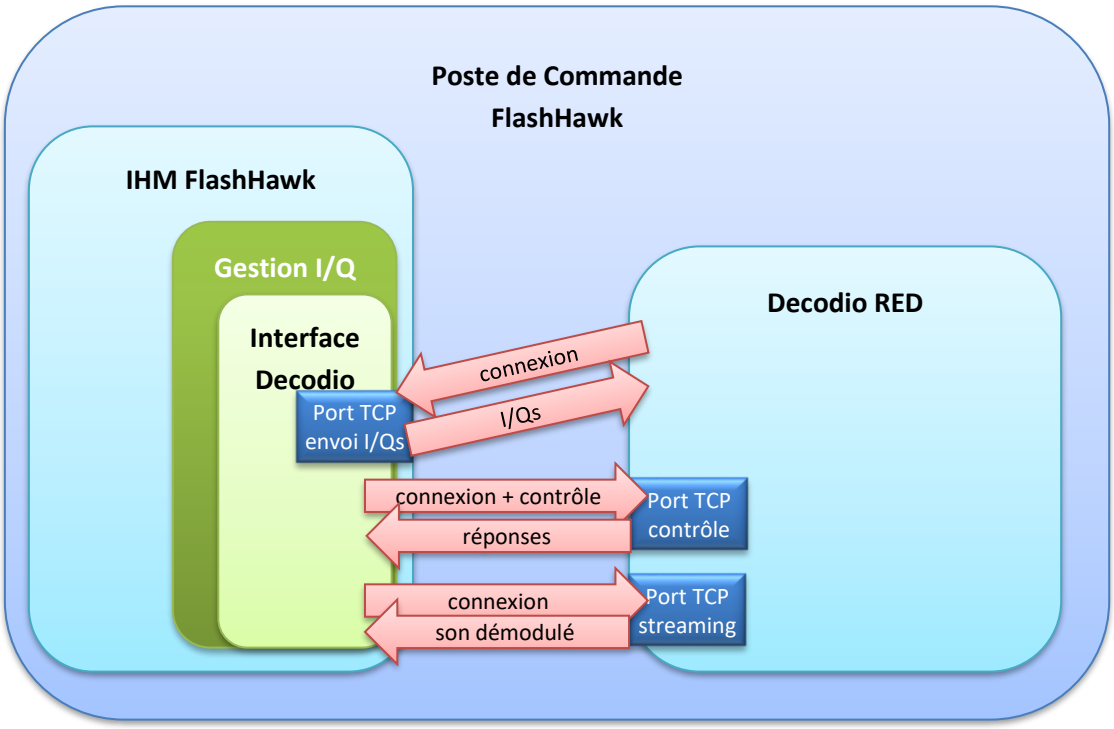
Tous les logiciels (FlashHawk, options) sont interconnectés entre eux par liaison IP, qu'elle soit en loopback sur le même équipement ou entre équipements en réseau local, sur un switch physiquement restreint à l'avion.

4 INTERFACE VERS LA CONSOLE D'EXPLOITATION

Le document [R2] décrit l'interface que le Poste de commande FlashHawk propose pour s'intégrer avec une console d'exploitation.

Cette interface est une option qui est implémentée dans une bibliothèque dynamique dédiée, qui n'est présente que dans le livrable correspondant au type de version. Cette bibliothèque est automatiquement chargée au démarrage de l'application, si elle est présente.

5 INTERFACE VERS DEMODULATEUR DE FLUX I/Q DECODIO RED



Pour la réception du flux I/Q, un « device TCP » est configuré dans Decodio en renseignant l'adresse et le port sur l'application FlashHawk. Decodio RED initie alors la connexion et FlashHawk commence à envoyer continuellement le flux I/Q sélectionné dans son IHM. La trame contient :

- (Format binaire)
- Un entête décrivant les flux I/Qs, incluant : fréquence d'échantillonnage, largeur de bande, nombre d'échantillons.
- Les séries d'échantillons I et Q, chacun au format flottant 32 bits.

Pour le contrôle de Decodio, FlashHawk connait l'adresse et le port de contrôle de Decodio RED. Il s'y connecte et transmet :

- (Format texte json)
- Protocoles de démodulation à activer.
- Choix de la sortie du son démodulé : aucune, carte son ou streaming.

En retour, FlashHawk reçoit :

- (Format texte json)
- Description de l'état de Decodio RED

Pour la réception du son démodulé, FlashHawk connait l'adresse (identique) et le port de streaming de Decodio RED. Il initie la connexion et reçoit continuellement le son démodulé. La trame contient :

- (Format binaire)
- Entête décrivant les échantillons, incluant : Fréquence d'échantillonnage, nombre d'échantillons.
- Les séries d'échantillons en son mono, en 16 bits signés.

Cette interface est une option qui est implémentée dans une bibliothèque dynamique dédiée, qui n'est présente que dans le livrable correspondant au type de version. Cette bibliothèque est automatiquement chargée au démarrage de l'application, si elle est présente.

6 LIVRABLES / TYPES DE VERSIONS

	IHM préparation, exploitation, rejeu	Export console d' exploitation	Export Decodio	Clustering (production émetteurs)	FlashHawk.proto	IHM de maintenance	Bootloader embarqué	Image noyau linux, fichiers système linux, fichiers FlashHawk
IHM	✓							
IHM interface	✓	✓	✓					
Serveur de calculs				✓				
Image embarqué							✓	✓
Mise à jour embarqué								✓
Maintenance						✓		
Toolkit (zip)					✓			

7 SECURISATION / ACCES AUX SYSTEMES

- Poste de commande Windows 10
  - Comptes utilisateurs accessibles au client.
  - Pour la maintenance, des comptes administrateurs sont présents, mais leurs mots de passes complexes ne sont pas connus du client (uniquement par le représentant d'Avantix en local).
  - La mise en place possible d'une configuration sécurisée, pour la version France, conformément à l'exigence « Le système d'exploitation du poste de commande doit être durci conformément aux guides fournis par l'Administration », n'est pas applicable à l'export, et reste à la demande du client.
- Unité de calcul Windows 10 ou Windows serveur
  - Pour la maintenance, des comptes administrateurs sont présents, mais leurs mots de passes complexes ne sont pas connus du client (uniquement par le représentant d'Avantix en local).
  - Aucun autre compte n'est présent.
- Capteur embarqué Linux
  - Pour la maintenance, des comptes administrateurs sont présents, mais leurs mots de passes complexes ne sont pas connus du client (uniquement par le représentant d'Avantix en local).
  - Aucun autre compte n'est présent.
  - Ports de connexion en mode export :
    - Service de connexion SSH
  - Ports de connexion en mode maintenance :
    - (Activable temporairement pendant une opération de maintenance)
    - Service de connexion SSH
    - Serveur Web de mise à jour / maintenance.
    - Serveur FTP.
      - Lorsqu'il est activé, permet d'échanger facilement des fichiers de logs ou de configuration pour des besoins de maintenance.

Les mots de passe « complexes » sont choisis conformément aux recommandations de l'ANSI.

Ces mots de passe sont différents entre les systèmes de niveau X0/X1/X2.

8 NIVEAU DE SPECIFICATIONS PAR ZONE GEOGRAPHIQUE

Le logiciel applique les différentes spécifications de niveau X0, X1 et X2 (suivant les zones géographiques) décrites dans le document [R2].

Le tableau ci-après décrit l'impact sur le logiciel de chacun des sous-systèmes pour chacun des éléments de spécification donnant lieu à une différenciation :

	Poste de commande	**Unité serveur de calcul**	**Capteur embarqué**
• **Vitesse de balayage** • **Largeur de bande de chaque voie**	Limitation des paramètres de programmation d'écoute saisis par l'opérateur ou calculés par le poste de commande.	N/A	Vérification de la conformité des paramètres d'écoute : non application si non conformes.
• **Sensibilité**			Suppression des détections instantanées non conformes.
• **Délai d'obtention de la position 3D de l'émetteur**	N/A	N/A	Rétention des détections instantanées pour augmenter le délai en X0.
• **Estimation par le système de la polarisation de l'émetteur**	N/A	N/A	Arrondi de la polarisation des détections instantanées pour correspondre aux valeurs autorisées.
• **Séparation des canaux**		Paramétrage de l'algorithme de corrélation émetteur.	
• **Nombre maximal d'émetteurs au sol pouvant être pistés en parallèle par le système**	N/A	Oubli des émetteurs inactifs les plus anciens. Si nombre encore supérieur, suppression des émetteurs les plus distants.	N/A

Le FPGA n'est pas impacté, car il ne traite aucune donnée concernée par les différenciations.

9 COMPOSANTS LOGICIELS TIERS

	Capteur embarqué Système d' exploitation	Capteur embarqué Application	Poste de commande / IHM Système d' exploitation	Poste de commande / IHM Application	Unité serveur de calculs Système d' exploitation	Unité serveur de calculs Application	Poste de maintenance Système d' exploitation	Poste de maintenance Application
Noyau Linux	✓							
Windows			✓		✓		✓	
Wibu / CodeMeter		✓		✓		✓		✓
OpenSSL		✓						✓
Carmenta				✓		✓		
CG				✓				
FFMPEG		✓		✓				
FreeType				✓				
FTGL				✓				
IPP		✓		✓		✓		✓
Protobuf				✓				
rapidjson		✓		✓		✓		✓
spdlog		✓		✓		✓		✓
wxWidgets		✓		✓		✓		✓

	Capteur embarqué	Système d'exploitation
btrfsprogs		✓
busybox		✓
dropbear		✓
e2fsprogs		✓
ethtools		✓
gptfdisk		✓
grub		✓
libcap		✓
logrotate		✓
lua		✓
pciutils		✓
rsyslog		✓
syslinux		✓
systemd		✓
tcpdump		✓
usb		✓
util-linux		✓

	Capteur embarqué	Maintenance	Serveurs Web et FTP de
fcgi		✓	
json-c		✓	
libconfig		✓	
libtar		✓	
libxml2		✓	
lighttpd		✓	
pcre		✓	
pureftpd		✓	
sqlite		✓	
zlib		✓	

Ces composants, bien que présents, ne sont pas utilisés lorsque le système est livré au client export.
(voir 11.6 SERVEURS WEB ET FTP DE MAINTENANCE)

10 MAINTENANCE

Dans le cadre du contrat de maintenance, si le Client en a souscrit un auprès d'AVANTIX, les composants suivants sont mis à jour annuellement, du fait de leur implication dans la sécurité de la CIEEMG :

(Leurs versions sont incluses dans le rapport de test de la CIEEMG. Voir 14 PROCEDURE D'EXPORT)

	Système d' exploitation Capteur embarqué	Application Capteur embarqué	Système d' exploitation Poste de commande / IHM	Application Poste de commande / IHM	Système d' exploitation Unité serveur de calculs	Application Unité serveur de calculs	Système d' exploitation Poste de maintenance	Application Poste de maintenance
Noyau Linux	✓							
Windows			✓		✓		✓	
Wibu / CodeMeter		✓		✓		✓		✓
OpenSSL		✓						✓

Pour le capteur embarqué Linux, la mise à jour est une nouvelle image intégrale appliquée sur le stockage, en conservant les partitions de données et de configurations.
Pour les PCs Windows, la mise à jour comprend le setup d'installation de la dernière version de Windows (appliquée comme une mise à jour) et le setup d'installation de l'application FlashHawk correspondante.

A l'occasion de la maintenance, les mots de passe peuvent être changés, si les autorités autorisant l'export le jugent nécessaire, suivant les impacts sur la sécurité depuis la précédente version, et en particulier si l'évolution est majeure.

11 PROTECTION CONTRE LE DETOURNEMENT DU SYSTEME

Les détournements possibles suivants sont adressés :

- Remplacement d'éléments du capteur embarqué, du poste de commande ou de l'unité de calcul, avec ceux d'un système de niveau supérieur.
- Retro ingénierie des programmes du capteur embarqué
- Retro ingénierie des programmes du poste de commande
- Retro ingénierie des programmes de l'unité de calcul
- Modification des paramètres de configuration transmis depuis le poste de commande vers le capteur embarqué, par le lien Ethernet, bien qu'étant au format binaire et propriétaire.
- Modification des variables du programme en mémoire pour changer le bridage.

11.1 LOGICIELS

Grâce à un dongle, les modules logiciel clés des applications FlashHawk suivantes sont protégés contre la copie et la modification, par chiffrement :

- Application du poste de commande / IHM
- Application de l'unité serveur de calcul
- Application du capteur embarqué
- Application du poste de maintenance

Pour les applications listées, au minimum, sont chiffrés tous les exécutables et bibliothèques dynamiques contenant du code gérant le niveau de spécification (voir **8 NIVEAU DE SPECIFICATIONS PAR ZONE GEOGRAPHIQUE**), gérant une protection (voir **11.2 CAPTEUR EMBARQUE LINUX**), ou implémentant une option (citée dans **3 INTRODUCTION**).

Grâce à la technologie WIBU CodeMeter, les dongles ne peuvent pas être dupliqués et les programmes chiffrés ne peuvent être ni modifiés ni générés en version non chiffrée.

La technologie WIBU CodeMeter est basée sur un dongle privé de chiffrement (clé privée) et un dongle de déchiffrement fourni avec le système.

La fiabilité de la technologie WIBU CodeMeter est sans faille connue, basée sur le dongle de chiffrement restant privé. Voir **15.1 PROTECTION AVEC TECHNOLOGIE WIBU CODEMETER**.

Pour le chiffrement des modules et la génération des dongles, des codes produits différents sont utilisés suivant le niveau X0/X1/X2 et l'option. Un dongle Xn livré avec un système Xn ne peut donc pas être utilisé avec un système Xm. De même, une option ne peut être utilisée qu'avec un dongle « à option ».

11.2 CAPTEUR EMBARQUE LINUX

Le logiciel du capteur embarqué s'exécute sur la carte CPU du capteur embarqué.

Le stockage de celle-ci contient une image bootable d'un système linux :

- A partir d'un ordinateur de développement Avantix, l'image est générée par une procédure propre à Avantix.
- Le noyau linux est directement compilé et intégré sans utilisation de distribution officielle type Debian ou Ubuntu, par exemple.
- Seuls les composants strictement nécessaires sont inclus dans l'image, avec le logiciel FlashHawk.
- Le démarrage s'effectue par l'intermédiaire d'un bootloader Grub dont le délai est réglé à 0 secondes, pour empêcher tout paramétrage du démarrage en temps réel.

Le stockage de la carte CPU contient la table de calibration propre au porteur (avion) auquel le capteur embarqué est destiné.

L'échange du capteur embarqué, de la carte CPU ou du stockage avec un système de niveau différent rendrait le système inopérant du fait de l'utilisation de l'autre table de calibration non appropriée liée à l'autre porteur.

La table de calibration est chiffrée avec une clé privée X0/X1/X2 dédiée. Le logiciel X0/X1/X2 contient la clé publique X0/X1/X2 pour déchiffrer la table. Il y a donc un couplage fort entre la table de calibration et le logiciel. Une modification du stockage embarqué en échangeant uniquement la partie logicielle, rendrait le système inopérant : un logiciel de niveau Xn ne peut pas lire (déchiffrer) une table de niveau Xm.

Le logiciel étant protégé (par chiffrement Wibu) contre la modification, il n'est pas possible de remplacer la table par son équivalent complètement déchiffré : le logiciel échouerait au chargement de celle-ci en essayant de la déchiffrer.

Le chiffrage utilisé pour la table est basé sur la technologie asymétrique RSA, avec une longueur de clé de 4096 bits. Les fonctions de chiffrement et de déchiffrement RSA de la bibliothèque OpenSSL sont utilisées. Également, une implémentation d'une optimisation possible de ce chiffrement est décrite en **15.2 OPTIMISATION CHIFFREMENT TABLE DE CALIBRATION**.

L'outil permettant le chiffrage de la table de calibration est accessible depuis l'application de maintenance, présente sur le poste de maintenance.

Au démarrage de l'application principale du capteur embarqué, celle-ci vérifie les fichiers clés de sécurité :

- /etc/passwd ne doit pas être modifié
- /etc/shadow ne doit pas être modifié
- /home/root/.ssh/ ne doit pas contenir de fichier
- /etc/init.d/rcS ne doit pas être modifié
- /etc/init.d/ ne doit contenir aucun autre fichier que rcS
- Tous les scripts appelés par rcS ne doivent pas être modifiés
- /bin/conffile ne doit pas être modifié. Utilisé par rcS, il est bridé pour ne lire que les scripts par défaut. En phase de mise au point, il permettait des ajustements sans mise à jour.
- /bin/maint ne doit pas être modifié. C'est le script lançant les serveurs FTP et HTTP. /bin/maint est appelé par /opt/maint, qui lui est au format binaire et protégé par Wibu CodeMeter (voir **11.6 SERVEURS WEB ET FTP DE MAINTENANCE**)

En cas d'échec, l'application principale s'arrête immédiatement.

Un outil du capteur embarqué permet de calculer et d'afficher la référence, en utilisant un hachage SHA-256. Celle-ci est alors saisie dans l'application de maintenance, qui produit un fichier chiffré (avec la clé RSA privée X0/X1/X2) à copier sur le capteur embarqué. L'application principale déchiffrera alors le fichier (avec la clé RSA publique X0/X1/X2) pour connaitre la référence à comparer à celle qu'il calcule au démarrage. De cette manière, le capteur embarqué est prémuni contre l'intrusion, sur le service SSH, d'une personne ne disposant pas du mot de passe, mais qui aurait remplacé, sur le stockage extrait, les fichiers clés de sécurité avec un nouveau mot de passe.

Le logiciel étant protégé (par chiffrement Wibu) contre la modification, il n'est pas possible de désactiver cette vérification des fichiers clé de sécurité, ni d'utiliser une référence non chiffrée.

11.3 COUPLAGE ENTRE LES SYSTEMES

A l'occasion de messages échangés régulièrement :

- « Changement de configuration » depuis le Poste de commande vers le Capteur embarqué
- « Configuration en cours » depuis le Capteur embarqué vers le Poste de commande
- « Configuration en cours » depuis le Poste de commande vers l'Unité de calcul
- « Identifiants émetteurs » depuis l'Unité de calcul vers le Poste de commande

Le niveau de spécification est inclus dans le message, afin de vérifier le niveau de l'interlocuteur. En cas d'incohérence, toutes les communications suivantes seront rejetées, jusqu'à la prochaine reconnexion.

Le FPGA n'est pas inclus dans le couplage, car il n'a pas d'impact sur le niveau de spécification, et que le couplage entre les différents logiciels est suffisant, puisqu'il lie tous les logiciels avec la table de calibration, et donc le porteur (avion).

11.4 PROTECTION DU PROTOCOLE APPLICATIF

Toutes les limitations spécifiques du poste de commande, qui sont traduites par l'envoi de configuration au capteur embarqué, n'ont pour finalité que de faciliter l'utilisation du capteur embarqué avec ses limitations spécifiques (de même niveau entre le poste de commande et le capteur embarqué, s'il n'y a pas d'échange de sous-systèmes, ni d'intermédiaire inséré par malveillance).

Le logiciel du capteur embarqué prend la décision au dernier moment d'appliquer ou non la configuration, suivant la conformité des paramètres d'écoute.

L'interception et la modification du protocole applicatif pour accélérer le balayage serait sans effet sur le capteur embarqué, qui considèrera non conforme le message de configuration.

Également, le logiciel embarqué vérifie la conformité numérique de tous les paramètres reçus par le protocole applicatif. Si un seul paramètre possède une valeur hors norme, il rejette immédiatement le message sans aller plus loin dans les traitements.

Concernant les données reçues suivantes :

- Par le Poste de commande depuis Capteur embarqué
- Par le Poste de commande depuis l'Unité serveur de calcul
- Par l'unité serveur de calcul depuis le Poste de commande
- Par le Poste de commande depuis Decodio RED, concernant le son démodulé

Une vérification de conformité pouvant générer des dysfonctionnements, est réalisée, afin de rejeter le message sans effectuer de traitement. Par exemple : transmission du nombre de données à transmettre dans la suite du message.

Concernant les données reçues depuis la console d'exploitation vers le Poste de commande, rejette sans dysfonctionner, tout message non conforme au protocole protobuf ou non conforme au fichier FlashHawk.proto .

Concernant les données reçues depuis Decodio RED en réponse à une requête de contrôle du Poste de commande FlashHawk, celui-ci rejette sans dysfonctionner, tout message non conforme à la norme JSON ou non conforme à l'interface de Decodio RED.

11.5 IMPLEMENTATION SPECIFICITES

Les valeurs spécifiques et portions de code logiciel spécifiques à X0/X1/X2 sont gérées par des directives de compilation dans le code C++ des sous-systèmes concernés. Le code gérant les autres spécificités n'est pas seulement du code non appelé, mais il n'est pas présent.

Ex (Limitations arbitraires dans l'exemple) :

```
#ifndef XLEVEL
#error "XLEVEL not defined"
#endif
#if XLEVEL == 0
#define MAX_SCAN_SPEED_MHZ_PER_SEC 10
#elif XLEVEL == 1
#define MAX_SCAN_SPEED_MHZ_PER_SEC 20
#elif XLEVEL == 2
#define MAX_SCAN_SPEED_MHZ_PER_SEC 1000
#endif
…
```

11.6 SERVEURS WEB ET FTP DE MAINTENANCE

Ces serveurs sont utilisés lors des phases de développement, de mise au point et de maintenance.

Ces serveurs sont démarrés de 2 manières :

- Automatiquement, par un programme protégé par Wibu et un code produit spécifique de maintenance. Seules les équipes de développement et de maintenance possèdent un dongle avec le code produit spécifique maintenance.
- A partir de la session SSH, en exécutant manuellement un unique script.

12 MARQUAGE DU TYPE DE SPECIFICATION DANS LA VERSION

Les versions du logiciel incluent le type de spécification (X0/X1/X2) avec la version du logiciel de chaque sous système, en respectant le format suivant :

A.B.CX

A, B et C désignent les numéros majeur, intermédiaire et mineur de version.

X désigne de type de spécification 0, 1 ou 2.

Ex : 1.4.171 pour une version 1.4.17 de type X1.

Le texte de version est présent dans le nom de chaque fichier livrable d'installation. Il est visible dans le « splash screen » au démarrage pour la version du poste de commande et dans le menu « A propos » présentant les versions de tous les sous-systèmes et composants.

Cela permet un contrôle clair lors des phases de tests en pré-livraison et d'attirer l'attention de tout intervenant technique ayant été formé.

13 GESTION DE CONFIGURATION LOGICIELLE

Pour empêcher toute erreur de la part du développeur en charge de générer les fichiers livrables d'installation, les éléments suivants sont mis en œuvre :

- Quelle que soit la plateforme et le sous-système, l'absence de la constante XLEVEL passée à la compilation, cause une erreur de compilation dans les fichiers sources C++ concernés par la spécialisation, et empêche ainsi le processus de génération d'aboutir.

```
#ifndef XLEVEL
#error "XLEVEL not defined"
#endif
```

- Plateforme Windows pour le poste opérateur et l'unité de calculs :
  - Mise en œuvre :
    - Compilation / génération des exécutables et bibliothèques dynamiques avec Visual Studio :
      - Configurations de génération :
        - ReleaseX0
        - ReleaseX1
        - ReleaseX2
      - Les propriétés communes de compilation de tous les projets ajoutent -*DXLEVEL=0* (ou 1 ou 2), suivant la configuration.
      - Les noms des dossiers de résultat de compilation sont formés avec le nom de la configuration, pour être sûr de ne pas garder des fichiers compilés précédemment avec un XLEVEL différent.
    - Génération des setups avec InnoSetup :
      - Le fichier iss commun récupère les exécutables et bibliothèques depuis :
        - bin_x64_X{#XLEVEL}
        - PluginsDLL_x64_X{#XLEVEL} .
      - Un fichier iss par spécialisation :
        - setup_FlashX0_MMI.iss
        - setup_FlashX1_MMI.iss
        - setup_FlashX2_MMI.iss
        - setup_FlashX0_MMI_interface.iss
        - setup_FlashX1_MMI_interface.iss

- setup_FlashX2_MMI_interface.iss
- setup_FlashX0_Computation_Server.iss
- setup_FlashX1_Computation_Server.iss
- setup_FlashX2_Computation_Server.iss
- setup_FlashX0_Maintenance.iss
- setup_FlashX1_Maintenance.iss
- setup_FlashX2_Maintenance.iss

- Chaque iss spécialisé produit un setup dont le nom contient la version :
  - FlashHawk_MMI_setup_A.B.CX.exe
  - FlashHawk_MMI_interface_setup_A.B.CX.exe
  - FlashHawk_Serveur_setup_A.B.CX.exe
  - FlashHawk_Maintenance_setup_A.B.CX.exe
  - (version A.B.C de spécialisation X, ex : 1.4.171 pour une version 1.4.17 de type X1)
- Un script général permet d'automatiser plusieurs étapes :
  - Chiffrement Wibu CodeMeter correspondant X0, X1 ou X2
  - Exécution de tous les iss InnoSetup correspondant X0, X1 ou X2
  - Archivage du FlashHawk.proto dans le zip « toolkit » versionné
- Un script est proposé pour chaque niveau :
  - setup_FlashX0.bat
  - setup_FlashX1.bat
  - setup_FlashX2.bat

- Génération des livrables :
  - Avec Visual Studio, le développeur choisit la bonne configuration avant de lancer la génération qui doit être utilisée pour le setup à livrer.
  - Lancement du script de haut niveau.
  - Récupération et archivage des setups produits.
  - (L'éventuelle omission de compilation possible par le développeur est un problème qui n'est pas spécifique à la spécialisation, et est détectée au début des tests par le contrôle de la version affichée dans le logiciel)

- Plateforme Linux pour le capteur embarqué :
  - Mise en œuvre :

- Constante XLEVEL attendue dans le Makefile de plus haut niveau. Erreur si absente.
- Appel de gcc avec -DXLEVEL=$(XLEVEL) dans chacun des Makefile des sous projets générant des exécutables et bibliothèques concernés par la spécialisation.
- Chiffrement Wibu CodeMeter correspondant X0, X1 ou X2.
- Fichiers livrables produits :
  - FlashHawk_A.B.CX.img (pour création carte compact flash)
  - FlashHawk_A.B.CX_full.tar.gz (pour mise à jour)

▪ Génération des images :

- make XLEVEL=0 *(ou 1 ou 2)*
- Récupération et archivage des images produites.

Tous les exécutables et bibliothèques compilés de FlashHawk sont installés sans les symboles de débogage. Pour des besoins de validation, la production de livrables d'installation (internes à l'équipe de développement) avec symboles de débogages est possible uniquement en mettant en œuvre une gestion de configuration séparant et marquant sans ambiguïté ces version internes : « _INT » apposé au niveau de spécification 0, 1 ou 2 dans les paramètres de compilation et d'assemblage, ainsi que dans le marquage de la version.

14 PROCEDURE D'EXPORT

Pour chaque système expédié ou mis à jour :

- Compilation et génération des livrables d'installation dans le niveau X0/X1/X2 adéquate.
- Vérification du marquage de version dans les noms de fichiers des livrables d'installation :
  - Niveau de spécification et éventuel marquage de version interne.
- Activation du mode maintenance sur le capteur embarqué.
- Installation de tous les livrables : Capteur embarqué, Poste de commande, Unité de calcul.
- Exécution et vérification du marquage des versions relevées par le poste de commande :
  - Niveau de spécification et éventuel marquage de version interne.
- Application des mots de passe export.
- Vérification des mots de passe export.
- Vérification que les services FTP et HTTP sont disponibles sur le capteur embarqué (mode maintenance).
- Activation du mode export.
- Vérification que les services FTP et HTTP ne sont plus disponibles sur le capteur embarqué.
- Effacement des fichiers de log
- Validation du système, conformément au document [R3]
- Génération des sommes d'intégrité des fichiers clés du système (outil interne Checksum), et archivage de ce fichier à fournir aux autorités autorisant l'export.
- Mise à jour du document de suivi de modifications (Release Notes) d'après l'historique des modifications du code source.
- Production d'un rapport de test, contenant également le suivi des modifications et les versions des composants logiciels tiers impactant la sécurité de la CIEEMG.

Pour chaque version modifiant les procédures de compilation, d'assemblage ou de démarrage :
(Dont la 1ère version implémentant les éléments de ce document)

- Vérification de l'absence de symboles de débogage et de logique implémentée dans les logiciels chiffrés : utilisation de objdump et Ghidra.
- Vérification de l'impossibilité d'obtenir une image déchiffrée en mémoire exploitable des modules chiffrés : utilisation de Dumpit, ProcDump de SysInternals, Volatility.
- Vérification des ports effectivement ouverts sur le capteur embarqué, avec KALI.

15 ANNEXES

15.1 PROTECTION AVEC TECHNOLOGIE WIBU CODEMETER

15.1.1 Bilan de la technologie

https://www.wibu.com/fr/concours-hackers.html

Global Hackers' Contest 2017

To test the validity and strength of the newly patented encryption method Blurry Box, integrated with the anti-debug and obfuscation methods of CodeMeter Protection Suite, we launched a new contest, open to all hackers around the globe. The underlying principle of Blurry Box is the exact opposite of “security through obscurity”; based on Kerckhoffs’ Principle, Blurry Box cryptography uses published methods that greatly increase the complexity and time required for an attack to be successful.

The contenders were delivered a game application protected with Blurry Box cryptography that came with its license stored in a CmDongle. Between May 15th and June 2nd, they were requested to hack the protected game and prove they could run it without the provided dongle and without any Internet connection to a jury consisting of IT security scientists and independent from the challenge partners (Wibu-Systems, Karlsruhe Institute of Technology, and FZI ResearchCenter for Information Technology).

None of the 315 international contendants managed to send in a full crack of the encryption scheme. The only two exploits that were received were found to be incomplete: They simulated a record playback attack that did not lead to any valid result or playable game. The two participants who submitted their partial solution received a volunteer award of €1,000 each. The remaining €48,000 of the original prize at stake will go towards further research and development.

15.1.2 Explications des protections contre la retro ingénierie

https://www.wibu.com/magazine/keynote-articles/article/detail/software-protection-from-a-hackers-perspective.html

Memory dumping

I have two approaches for encrypted software, and both of them need a license. In the first approach I start the software and wait untill it's sitting unencrypted in memory. I then do a memory dump and reconstruct the software from it.

By the way, did I mention I hate CodeMeter®? I hate it because it only lets part of the software sit unencrypted in memory. The dump is then like a puzzle but without any type of pattern. My biggest challenge is to get the software to run so that all parts are eventually decrypted. I have to use the software intensively to do this. Unfortunately, I'm not an expert user of boring geology software. And even if I were, how would I know if every function's been run at least once? The best test plans of a manufacturer only manage to test about 80% of the software. If I managed 100% I would make a fortune selling test tools. I would then be sitting under a sun umbrella on a Caribbean beach sipping cocktails every day. Or maybe I'd buy a villa in Baden Baden.

I don't really like this type of hack either. For one thing, I have to protect it somehow if I want to sell it, and then I have to repeat the hack each time a new version of the software is released. How am I supposed to get rich if I can't automate anything?

I decide to change the hack and write my own tools to automatically remove the protective encryption wrap. This means I only have to press a button when a new version is released or when I come across a piece of software with the same protection. Up till now I've seldom had to change anything to cope with new releases. These tools give me an edge over my competitors. By the way, did I mention I hate CodeMeter®? CodeMeter® inserts encrypted traps into the software. If I fall into one of them the license is deactivated. And so far I haven't managed to detect them all. I guess they must have spent of lot of time designing them. For today, I think I'll do the hacks with the other two dongles. I'll look at the CodeMeter® dongle some time next week when I have nothing else to do.

15.2 OPTIMISATION CHIFFREMENT TABLE DE CALIBRATION

La taille du fichier étant de 500 Mo, le délai de génération et de chargement serait long, avec un chiffrement intégral. Un chiffrement partiel est mis en œuvre en restant tout autant efficace :

- Fichier constitué de petites portions de descriptions (valeur de fréquence, pas angulaire par fréquence, …) et d'importantes portions de données.
- 3 groupes de données, pour chaque angle et chaque fréquence :
  - Groupes 1 et 2 constitués chacun de la réaction en I/Q de chacune des 10 antennes : soit 80 octets par groupe.
  - Groupe 3, constitué d'une matrice 2x2 I/Q, soit 32 octets.
  - Ces 3 groupes de données sont utilisés par une opération « complexe » effectuant un produit entre chaque groupe.
- Chiffrement complet des portions de description des données.
- Chiffrement des données :
  - Groupe 1 ou 2, uniquement de 1 octet sur les 80, déterminé à un emplacement variable, mais déterministe.
  - Groupe 3, uniquement de 1 octet sur les 32, déterminé à un emplacement variable, mais déterministe.
- Toutes les antennes et tous les angles sont impactés en même quantités, sur l'ensemble du fichier. Dans le nombre flottant, le 3ème octet (offset + 2 en mémoire, en « little endian ») est sélectionné, car il correspond à la partie la plus significative de la mantisse.
- Un hacker pouvant toujours maintenir 79 des 80 octets de sa table d'origine, mais sacrifiant systématiquement 1 antenne sur 10 (pas toujours la même) de chaque angle capté possible : cela causerait un angle de détection aléatoire.