

Certification et Développement Sécurisé

Damien Sauveron

XLIM (UMR CNRS 7252 – Université de Limoges)

Équipe CRYPTIS

<http://damien.sauveron.fr/>



CDS - Limoges - 2019/2020 - D. Sauveron

1

Damien Sauveron

- Maître de conférences Habilité à Diriger des Recherches
 - Thèse au LaBRI/SERMA Technologies
 - Post doctorat Information Security Group du Royal Holloway University of London
- Membre de l'équipe CRYPTIS du laboratoire XLIM
 - 14 permanents + 14 doctorants
 - 1 salle dédiée à la sécurisation des objets connectés
- Enseignant dans les deux Masters CRYPTIS
 - Première formation BAC+5 en France en sécurité et cryptologie créée en 1986
 - Plus de 750 diplômés
- Président du WG11.2 de l'IFIP « Pervasive Systems Security »
- Membre du DC de l'IFIP « IoT » en charge des aspects sécurité



Université
de Limoges



CDS - Limoges - 2019/2020 - D. Sauveron

2

Plan

- La sécurité ?
 - Introduction à la sécurité
 - Quelques définitions
 - Quelques propriétés
 - Quelques « outils » pour mettre en œuvre ces propriétés
 - Que veut-on protéger ?
 - Comment ?
 - Comment vérifier la sécurité ?
 - UML
 - Méthodes formelles & Test
 - Évaluation sécuritaire et certification
 - Open source vs propriétaire
 - Outils d'analyse logiciel
 - Analyse de sécurité
 - Procédures d'attaques
- Les sécurités
 - Au niveau physique
 - Au niveau logicielle
 - Au niveau de l'environnement de production

Qu'est ce que la sécurité ?

La sécurité

- La sécurité est une chaîne dont la résistance est celle de son maillon le plus faible !



- On essaye pour voir ?
 - La sécurité physique est souvent négligée !

Qu'est ce que la sécurité ?

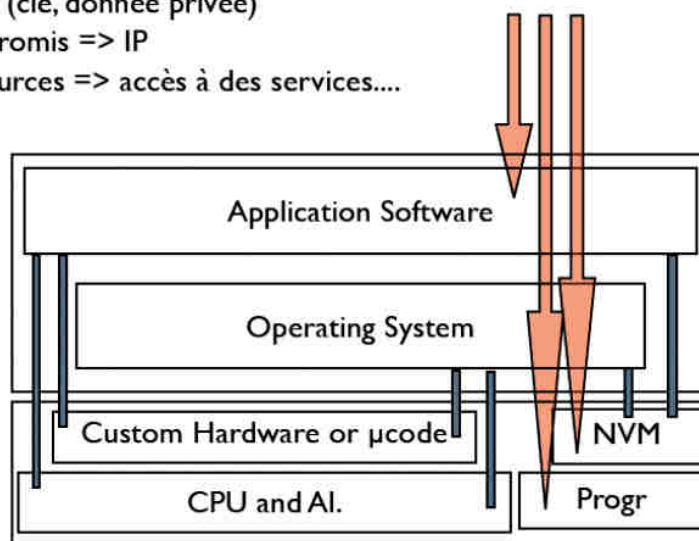
- Comme vous venez de le voir définir la sécurité n'est pas aussi simple qu'on le pense
- L'approche la plus courante concernant la sécurité informatique se résume sous le terme **CIA**
 - **Confidentialité** (Confidentiality)
 - **Intégrité** (Integrity)
 - **Disponibilité** (Availability)
- Pour d'autres types de sécurité (des personnes par exemple), nous devons la définir en relation avec les risques

Confidentialité

- La confidentialité consiste à empêcher **des utilisateurs non autorisés de lire** des informations pour lesquelles ils n'ont pas le droit.
- C'est en quelque sorte la propriété de « secret » attachée aux informations
- Traditionnellement, pour la plupart des gens la sécurité est assimilé à la confidentialité

La confidentialité

- Data compromise (clé, donnée privée)
- Programme compromis => IP
- Accès à des ressources => accès à des services....



Confidentialité

- Comment peut on assurer cette propriété ?
 - **par chiffrement** (mécanisme cryptographique) pour assurer la confidentialité lors de transmission d'information, i.e. pour les communications, mais aussi pour le stockage
 - **par l'utilisation de mécanismes de contrôle d'accès**
 - Pour un SEM, on peut utiliser des contrôles d'accès aux informations ou aux ressources (problème de granularité des droits d'accès)
 - Contrôle d'accès au SEM
 - Contrôle d'accès aux ressources internes au SEM
 - Contrôle d'accès logique à l'intérieur du SEM
- Attention : Il existe une propriété proche de la confidentialité qui est l'**anonymat**. Elle consiste à assurer la confidentialité d'une donnée bien particulière : l'**identité** (on peut la voir comme un sous ensemble)

Intégrité

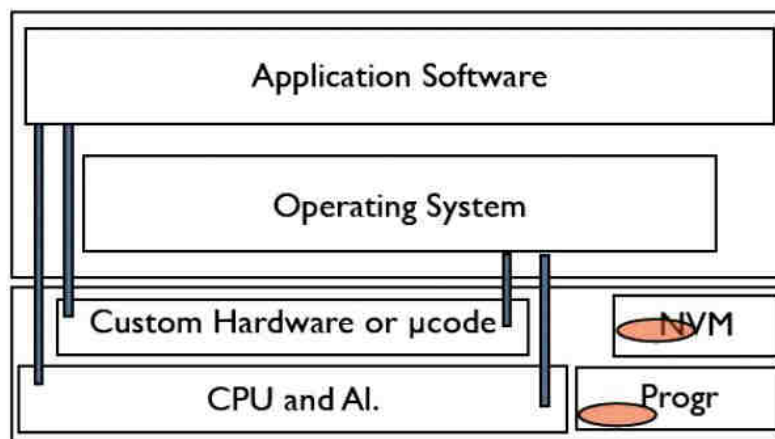
- L'intégrité consiste à s'assurer que les choses seront faites comme elles devraient l'être
- Dans le *contexte de l'informatique*, l'intégrité consiste à empêcher des utilisateurs **non autorisés d'écrire** des informations pour lesquelles ils n'ont pas le droit.
- Dans un *contexte général*, l'intégrité consiste à s'assurer que l'état d'un système n'a pas été modifié par des personnes qui n'ont pas le droit de le faire.

Intégrité

- Dans le contexte des *communications de données*
 - l'intégrité se restreint souvent à détecter les modifications de ces informations.
 - Une propriété proche consiste à identifier de qui l'information provient. C'est l'**authentification** des données (peut être vu comme une généralisation de l'authentification d'entité qui porte sur l'authentification d'une donnée bien particulière : l'**identité**)
- Comment peut on assurer cette propriété ?
 - Par du contrôle d'accès
 - Par des mécanismes de MAC ou de signature numérique (cryptographie asymétrique)
 - Par des mécanismes de CRC ou Checksum (somme de contrôle) ou de redondance
 - Mécanismes assurés par du matériel ou du logiciel

L'intégrité

- Données et programmes changés



Intégrité

- Informations concernées par l'intégrité :
 - Les données
 - Les programmes (i.e. le code)
 - Peut avoir un impact sur les données des applications, sur les données du système lui-même (autorisation de nouveau service, modification de table d'accès, etc.)
 - L'altération peut être volontaire (e.g. une attaque) mais également le fruit d'une erreur de programmation (i.e. bug)
 - L'exécution d'un programme ? **Comment garantir cela ?**

Disponibilité

- La disponibilité est la propriété d'un système d'**être accessible à la demande** par un utilisateur autorisé à y accéder.
- Cette notion va bien au delà de la sécurité et concerne aussi la tolérance aux fautes.
- D'un point de vue sécurité, on s'intéresse principalement à **éviter les attaques par déni de service** d'utilisateurs non autorisés.
 - Par exemple, dans le cas d'un serveur web, la disponibilité sera le fait que les requêtes sont satisfaites et que l'attaque par déni de service sera sans effet.
- On peut associer à la notion de disponibilité, celle de **QoS** (Quality of Service – Qualité de Service). Cette propriété **fonctionnelle** englobe plusieurs paramètres comme par exemple la capacité à répondre dans un délai prévu.

D'autres propriétés

- Attention, il y a d'autres propriétés de sécurité en plus de CIA.
- **Responsabilité** (Accountability)
 - En pratique, on ne peut pas empêcher des actions malveillantes de se produire.
 - => Les utilisateurs doivent être rendus **responsables** de leurs actions, y compris les mauvais usages du système
 - Cela se traduit souvent par la mise en place d'un mécanisme d'authentification puis un traçage et un enregistrement des actions ayant trait à la sécurité dans un log d'audit.
 - Cette notion permet de faire de l'**imputation** et de la **non répudiation**

D'autres propriétés

- En lien avec la sécurité :
 - **Fiabilité** (Reliability)
 - Pour les systèmes qui doivent fonctionner correctement dans des conditions adverses, la sécurité est clairement proche de la fiabilité (reliability) et de la sûreté (safety)
 - **Dépendabilité** (Dependability)
 - Terme parfois utilisé pour englober la sûreté et la fiabilité.
- Si les objectifs de la sécurité et de la dépendabilité sont proches, les méthodes utilisées pour évaluer les systèmes sont souvent similaires

Spécificités des Systèmes Embarqués et Mobiles (IoT/CaP)

- Le dispositif est entre toutes les mains
 - Porte ouverte aux intrusions, ...
 - En opposition à un data center qui est une unité de lieu bien délimité
 - Toutefois les grilles de calculs distribués ouvertes connaissent un peu la même problématique
 - Dispose souvent d'une connectivité
 - Porte d'entrée d'une possible attaque
 - Plate-forme parfois ouverte (type Java)
 - Facilite le time-to-market et l'écosystème mais aussi les attaques de l'intérieur
- Les menaces sont celles des systèmes informatiques classiques
 - Mais :
 - Le **contexte d'utilisation** de l'objet peut être **spécifique**
 - PayTV
 - Les **aspects système** peuvent être **spécifiques**
 - Attaques par déclenchement des chauffages en forçant la température sur les capteurs de chauffage
 - Les **enjeux**
 - Carte bancaire = argent

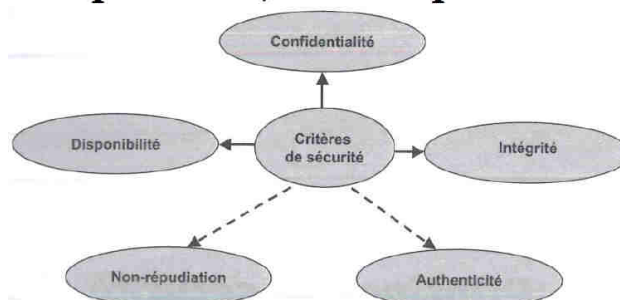
CDS - Limoges - 2019/2020 - D. Sauveron

17

Qu'est ce
qu'on
veut
sécuriser
?

RAPPEL

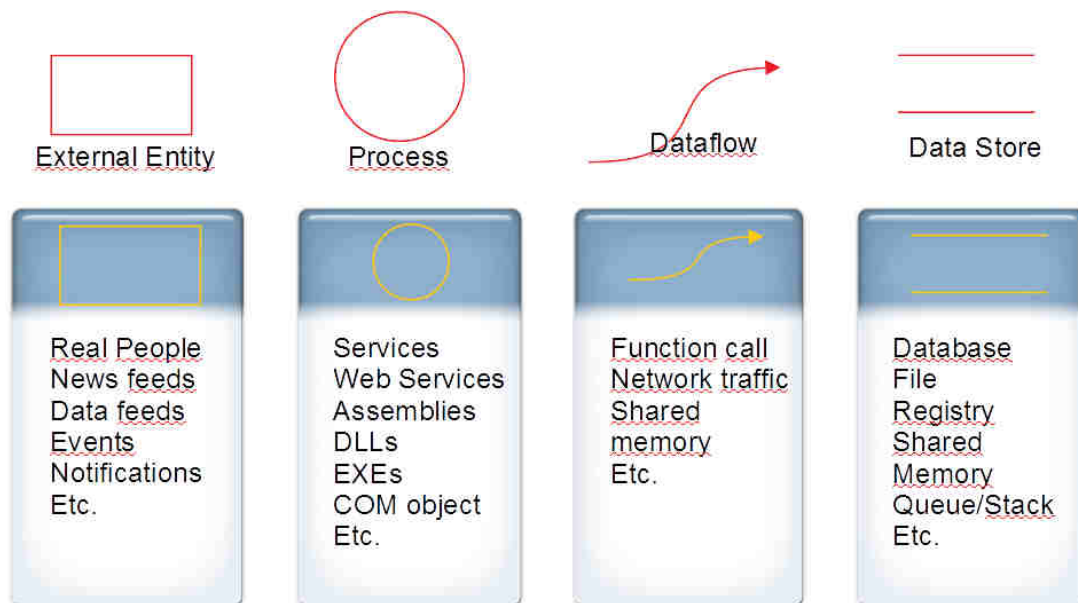
- **Sécurité**
- **Dépendabilité : Fiabilité & Sureté**
- **Confidentialité et Anonymat**
- **Intégrité et « authentification »**
- **Disponibilité**
- **Responsabilité**
=> **Imputation/Non Répudiation**



CDS - Limoges - 2019/2

Figure 1.1 – Critères de sécurité.

Asset (les biens !)



CDS - Limoges - 2019/2020 - D. Sauveron

19

Une autre définition de la sécurité

- Une autre façon de définir la sécurité, en particulier dans le cas de système non informatique, consiste à faire tout d'abord une **analyse des menaces**. La sécurité est alors définie comme la lutte contre les menaces perçues.
- Toutes les menaces ne sont pas toujours toutes intéressantes à contrer.
 - Il faut donc faire une analyse coût/bénéfice.
- L'**analyse de risques** consiste à évaluer l'importance de chaque menace pour décider si elle doit ou non être combattue.
 - En effet, il peut être moins coûteux de vivre avec certaines menaces que de les empêcher.

CDS - Limoges - 2019/2020 - D. Sauveron

20

Intégration de la sécurité

- De façon générale, on peut se demander comment la sécurité peut être intégrée dans un système ?
- Quels sont **les grands types de contrôles de sécurité** et **où sont-ils intégrés** ?
- La sécurité intégrée dans une application peut être contournée par une attaque sur le système d'exploitation !
- La sécurité intégrée dans le système d'exploitation peut quand à elle être contournée par une attaque au niveau du matériel !
- Aussi penser l'intégration de la sécurité dans un système **depuis sa conception** est bien meilleure que l'ajouter après coup.

Quels grands types de contrôles ?

- Lorsque l'on protège des données dans un système informatique, on utilise souvent des règles sur le comportement du système :
- Ces **règles** peuvent être :
 - Limiter les façons pour manipuler une donnée
 - Limiter les opérations qui peuvent être réalisées sur des données
 - Limiter les utilisateurs qui peuvent réaliser certains types d'actions
- Les contrôles de sécurité peuvent cibler les données, les opérations ou les utilisateurs (ou des combinaisons).
- Les règles ne sont pas les seuls types de contrôles de sécurité
 - Il ne faut pas oublier **la sécurité physique**.



Localisation des contrôles

- Un système IT classique peut se modéliser comme 5 couches :
 - Applications/Programmes
 - Services : e.g. fourni par un SGBD ou un système de fichier distribué
 - Le système d'exploitation : gestion des fichiers, des impressions, etc.
 - Le noyau de l'OS (Kernel) : intermédiaire pour l'accès au processeur et à la mémoire
 - Le matériel : le processeur et la mémoire
- Les contrôles de sécurité peuvent être localisés dans n'importe laquelle de ces couches.
- Les **mécanismes bas niveaux** (près du matériel) seront plus **génériques** et orienté vers la sécurité du système alors que plus les **mécanismes haut niveaux** (près des applications) seront plus **orientés utilisateur**.

Remarque : Identification/Authentification

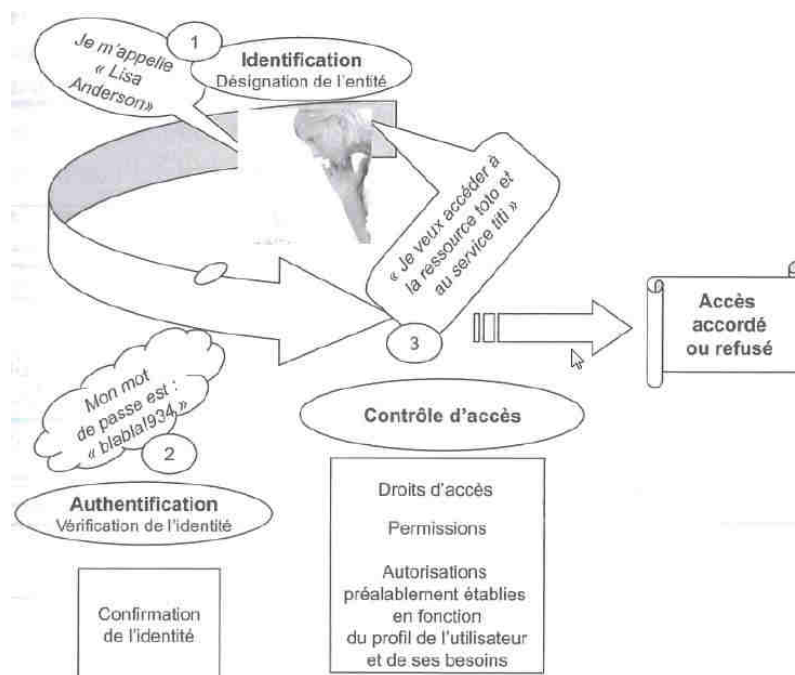
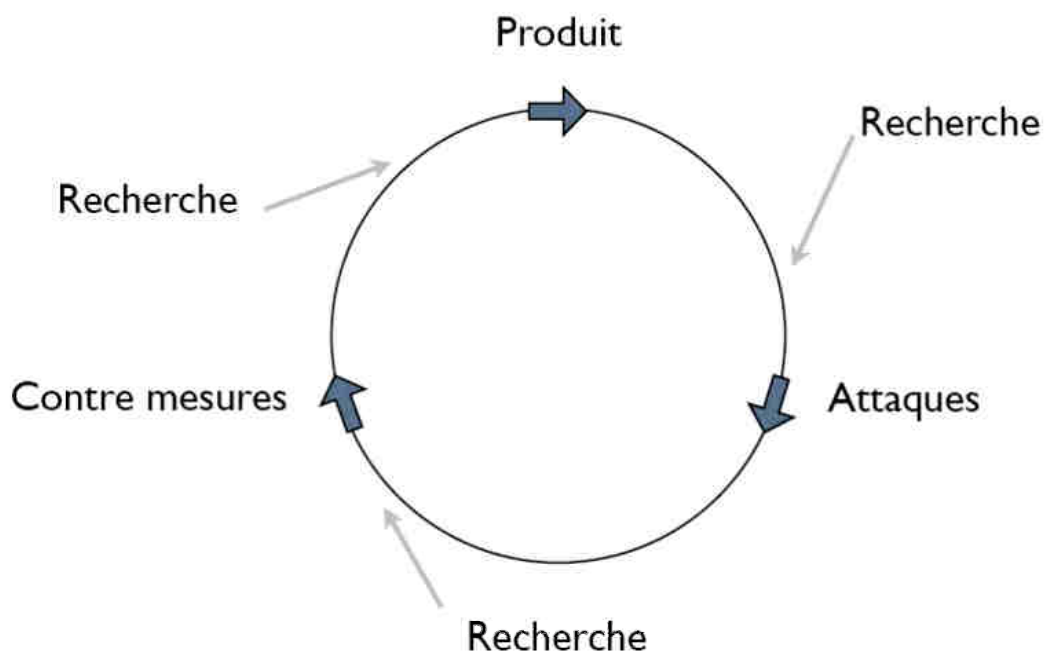


Figure 1.2 – Identification et authentification.

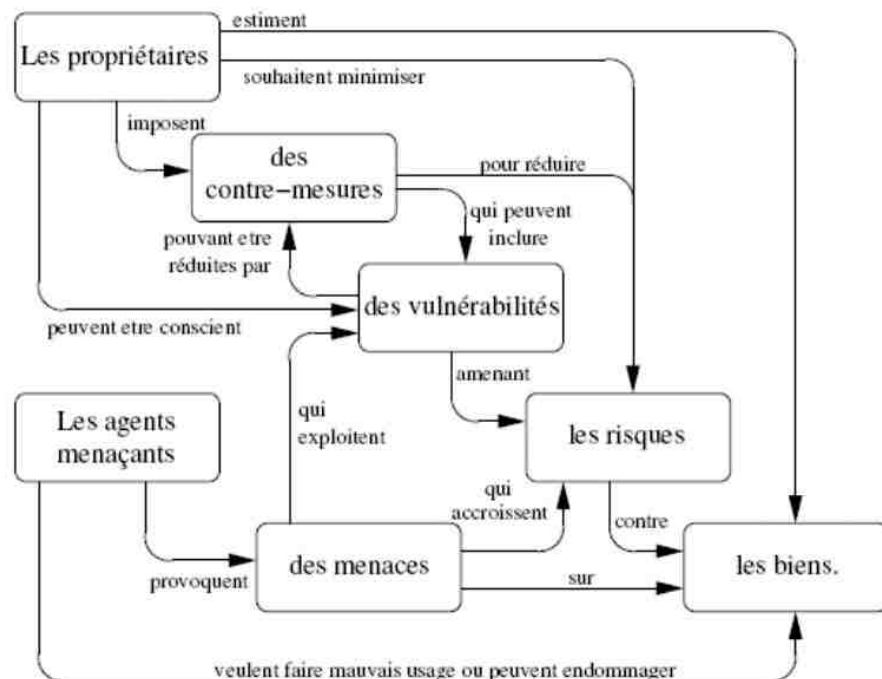
Quelques remarques

- Pour la sécurité des communications, le principal outil est la cryptographie
- Cette différence technologique est à la base du développement séparé de :
 - Comsec : Communication Security
 - Compusec : Computer Security

Un cycle infernal!

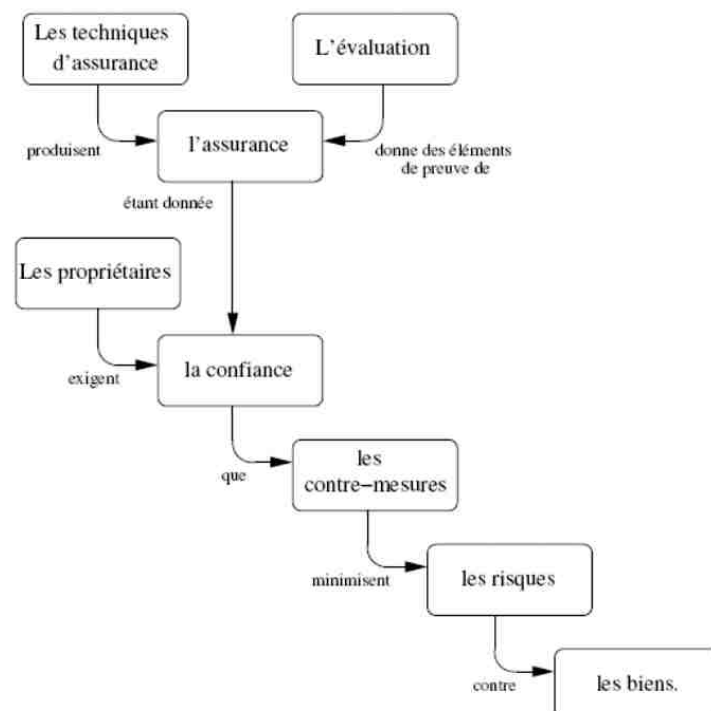


Concepts de sécurité et relations



27

La sécurité c'est bien, mais la confiance, c'est mieux !



28

Functionality versus Assurance

In assessing secure systems, two different aspects need considering

- ▶ **Functionality**, i.e. what security facilities are provided.
- ▶ **Assurance**, i.e. what guarantees are offered that the security functionality performs as claimed.

These two aspects are often reflected in **Security Evaluation Criteria**, from the **Orange Book** onwards.

Assurance versus Complexity

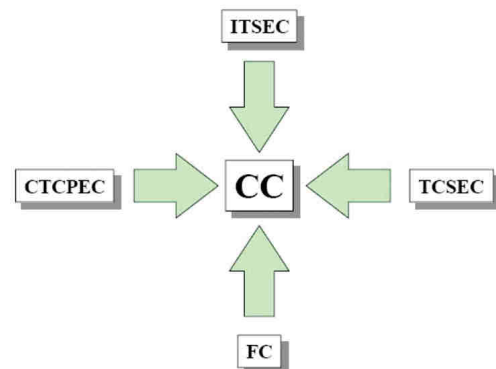
The work involved in providing a high level of assurance in security features is proportional to the complexity of those features.

Hence, if a high level of assurance is required it makes sense to minimise complexity.

Normes pour l'évaluation

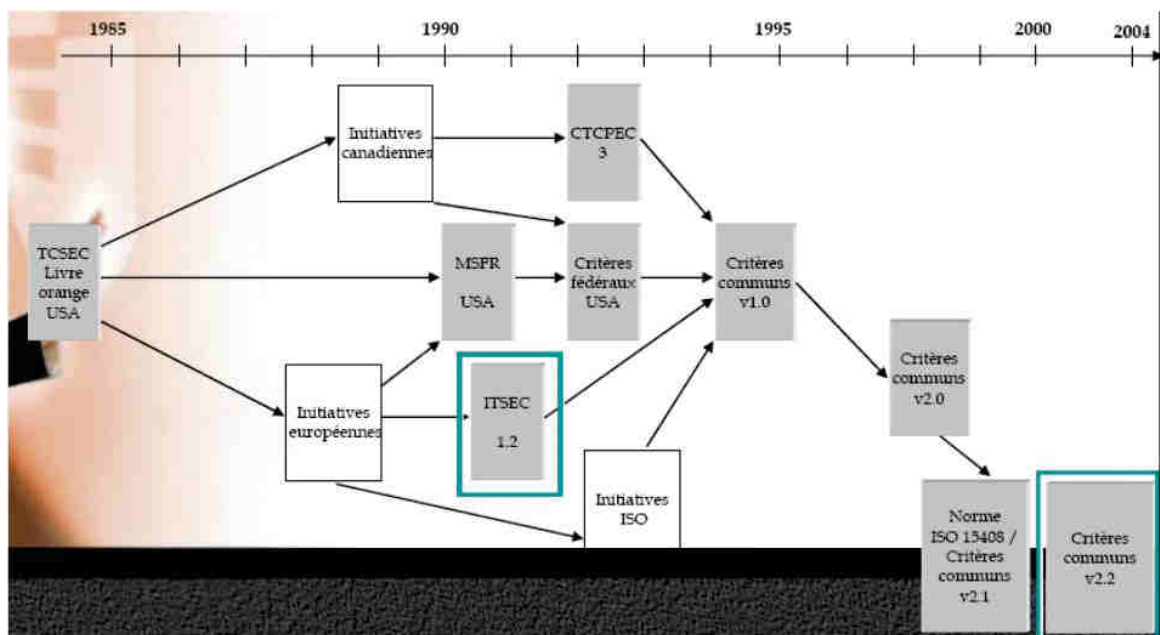
- Avant l'apparition des CC, les pays utilisaient des critères d'évaluation différents :
- l'Orange Book pour les États Unis ;
- les ITSEC pour la France, l'Allemagne, le Royaume-Uni et les Pays-Bas ;
- le CTCPEC pour le Canada.

Harmonisation des critères existants



CDS - Limoges - 2019/2020 - D. Sauveron

31



CDS - Limoges - 2019/2020 - D. Sauveron

32

Critères Communs

- Les Critères Communs (CC) sont une norme qui a pour vocation d'être utilisée comme base pour l'évaluation des propriétés de sécurité des produits et systèmes des Technologies de l'Information (TI). Ils sont définis par le Common Criteria Interpretations Management Board (CCIMB) et sont depuis quelques années une norme ISO définie comme ISO/IEC 15408

- **CC : Common Criteria v2.1, August 1999**

(<http://www.scssi.gouv.fr/documents/docs/CC/cc21.html>)

- Part 1 : Introduction and general model
August 1999 Version 2.1
- Part 2 : Security functional requirements
August 1999 Version 2.1
- Part 3 : Security assurance requirements
August 1999 Version 2.1

- **CEM : Common Methodology v1.0, August 1999**

(<http://www.scssi.gouv.fr/documents/docs/CEM/cem.html>)

- Evaluation methodology for PP and ST
Version 1.0
- Evaluation Methodology for levels EAL1 to EAL4
Version 1.0

33

Une norme internationale

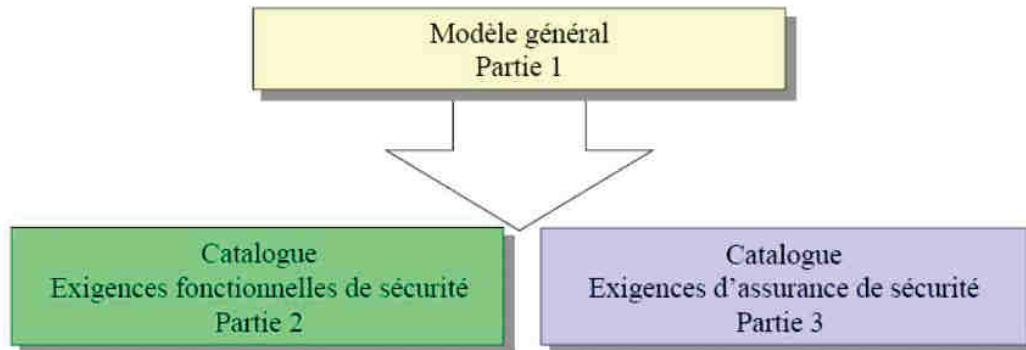
ISO/IEC JTC 1/SC27
Technologies de l'information - Techniques de sécurité



Juin 1999 Normalisation des Critères Communs

ISO/IEC IS 15408-1	Partie 1 Introduction et modèle général
ISO/IEC IS 15408-2	Partie 2 Exigences de sécurité fonctionnelles
ISO/IEC IS 15408-3	Partie 3 Exigences de sécurité d'assurance

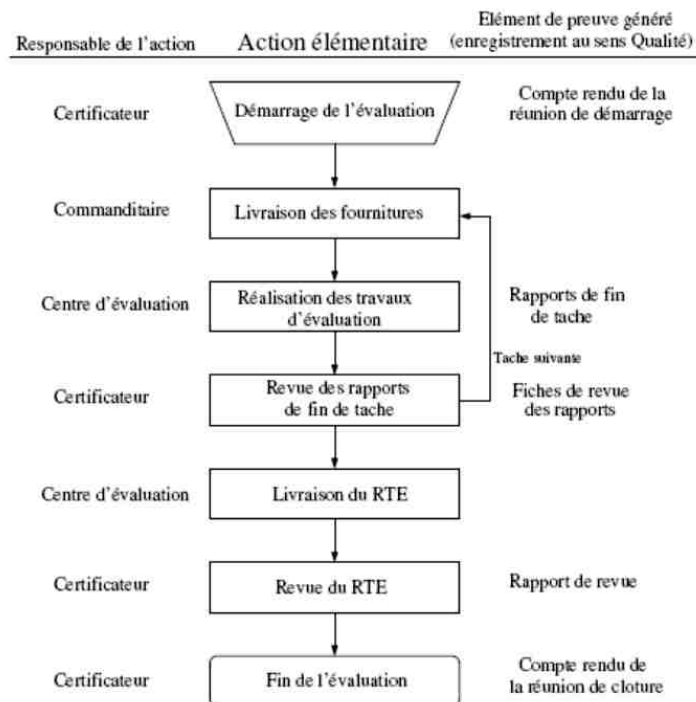
Expression des exigences



Public

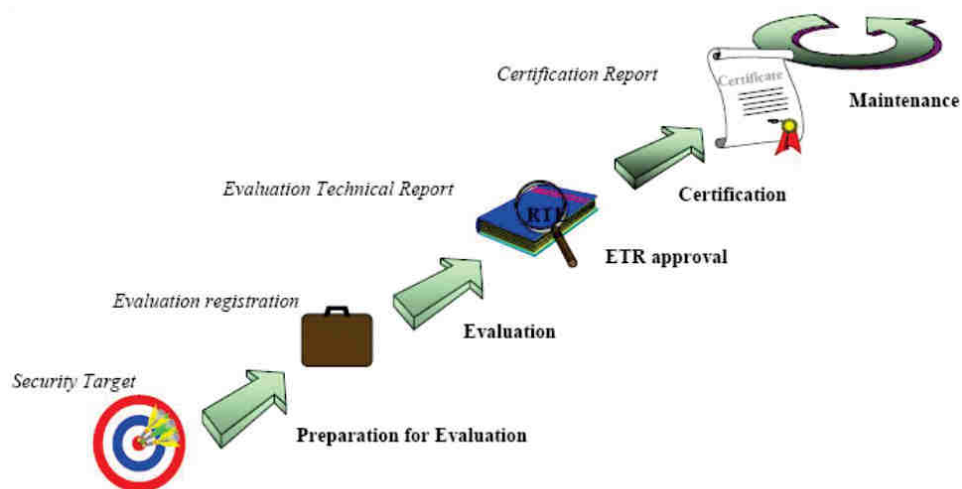
	Consumers	Developers	Evaluators
Part 1	Use for background information and reference purposes. Guidance structure for PPs.	Use for background information and reference for the development of requirements and formulating security specifications for TOEs.	Use for background information and reference purposes. Guidance structure for PPs and STs.
Part 2	Use for guidance and reference when formulating statements of requirements for security functions.	Use for reference when interpreting statements of functional requirements and formulating functional specifications for TOEs.	Use as mandatory statement of evaluation criteria when determining whether a TOE effectively meets claimed security functions.
Part 3	Use for guidance when determining required levels of assurance.	Use for reference when interpreting statements of assurance requirements and determining assurance approaches of TOEs.	Use as mandatory statement of evaluation criteria when determining the assurance of TOEs and when evaluating PPs and STs.

Déroulement de l'évaluation



37

Processus d'évaluation : les principales étapes



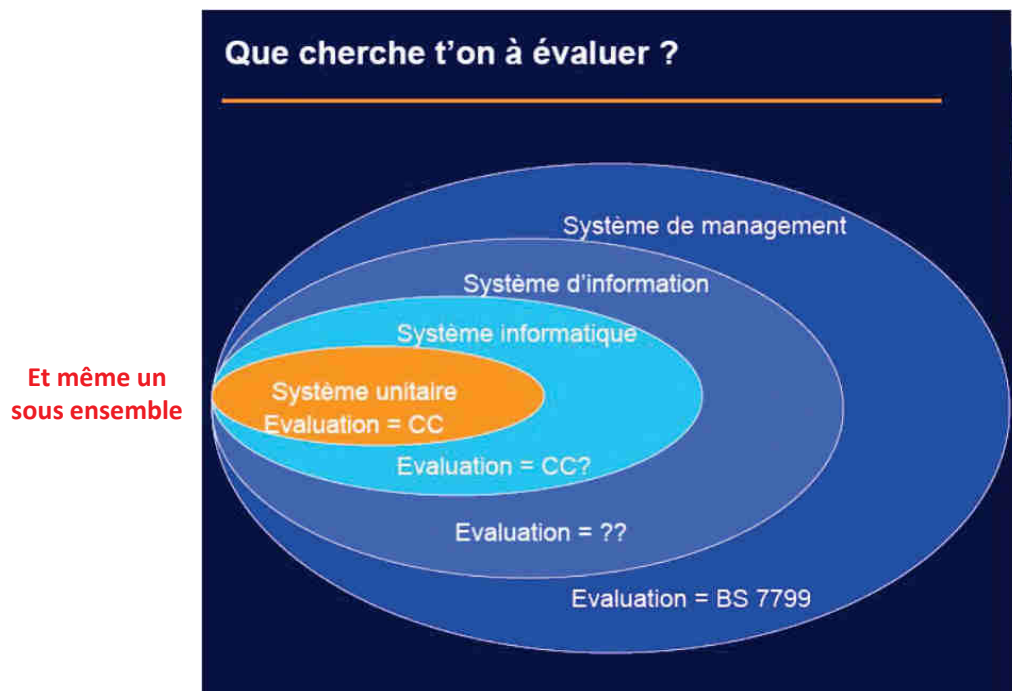
Quelques définitions

- TOE (Target Of Evaluation) pour définir la cible d'évaluation, i.e. le produit ou le système TI à évaluer + ses guides administrateurs et utilisateurs
- ST (Security Target) pour désigner une cible de sécurité, i.e. un document rassemblant l'ensemble des exigences de sécurité et des spécifications à utiliser comme base pour l'évaluation d'une TOE identifiée
- PP (Protection Profil) pour désigner un profil de protection, i.e. un document rassemblant l'ensemble des exigences de sécurité valables pour une catégorie de TOE qui satisfait des besoins spécifiques d'utilisateurs, indépendamment de son implantation (e.g. il existe des PPs pour l'évaluation de puce nue – i.e. sans applicatif –, pour le domaine des applications bancaires)

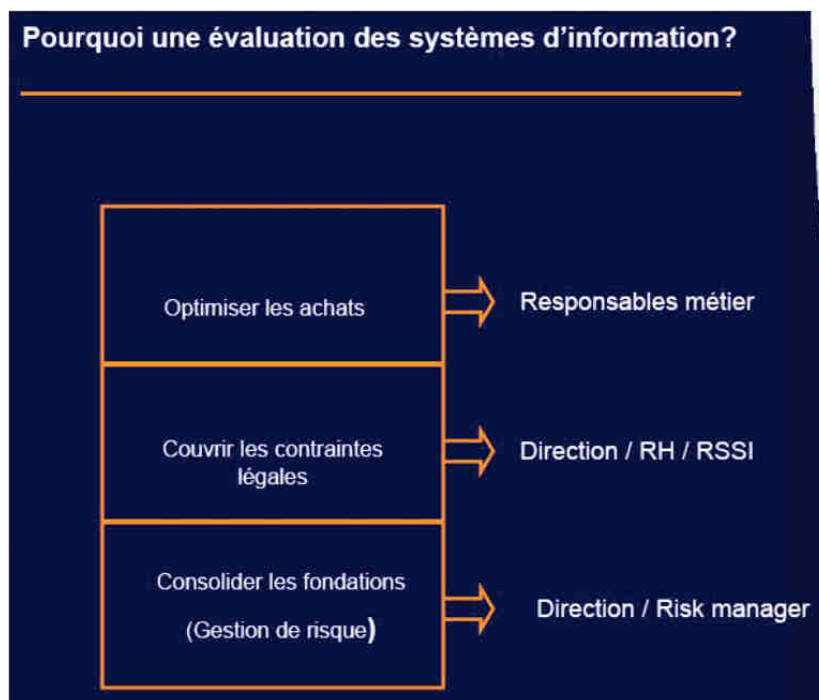
- Target of evaluation (TOE) : an IT product or system and its associated administrator and user guidance documentation that is the subject of an evaluation. It defines assets to protect.
- TOE Security Functions (TSF) : A set consisting of all hardware, software, and firmware of the TOE that must be relied upon for the correct enforcement of the TSP.
- TOE Security Policy (TSP) : A set of rules that regulate how assets are managed, protected, and distributed within a TOE.
- Security Target (ST) : Defines the target of evaluation, the environment, the threats, assets to protect, security objectives, assumptions.

39

Périmètre



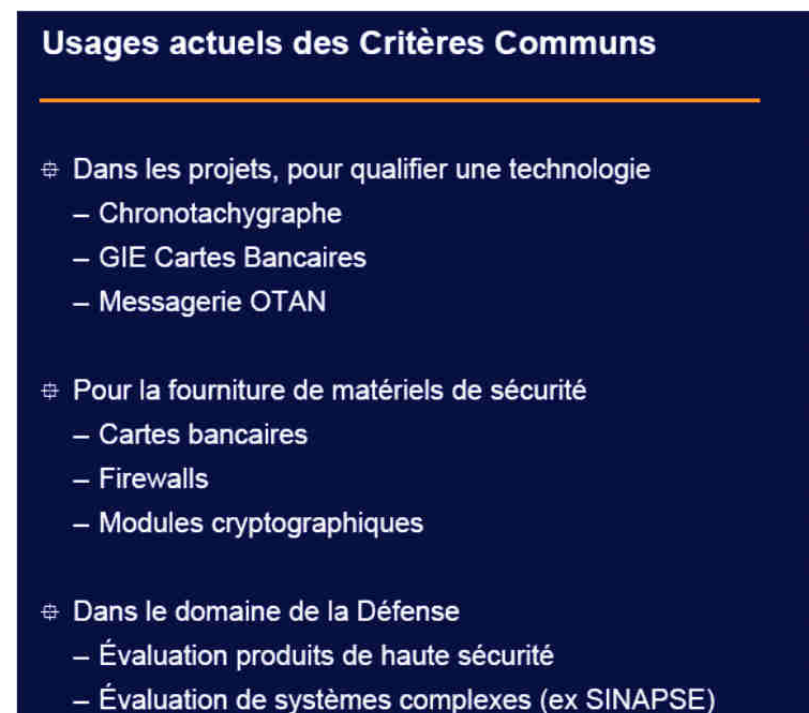
Etablissement de la confiance !



CDS - Limoges - 2019/2020 - D. Sauveron

41

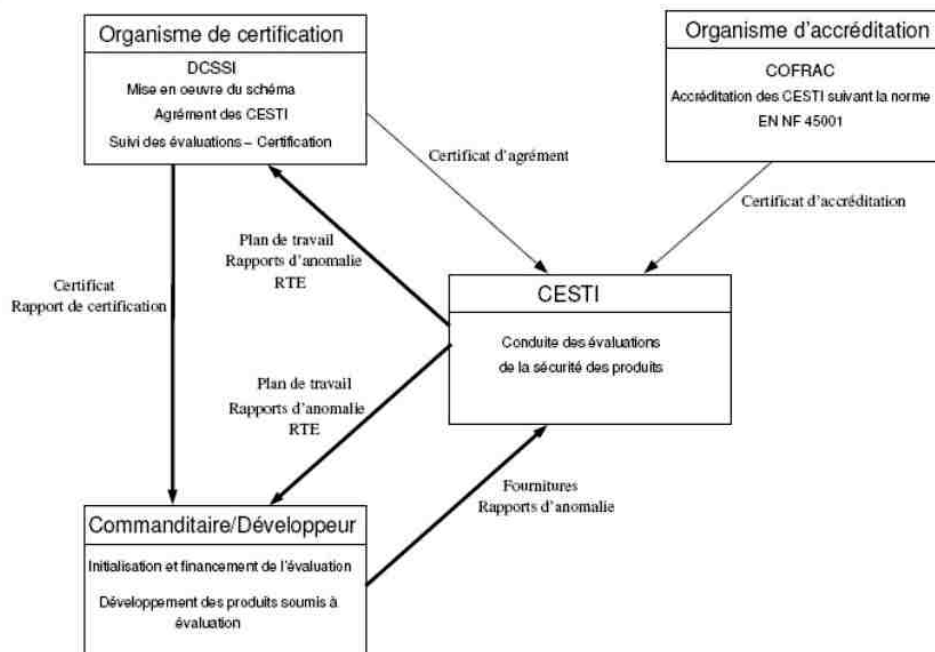
Etablissement de la confiance !



CDS - Limoges - 2019/2020 - D. Sauveron

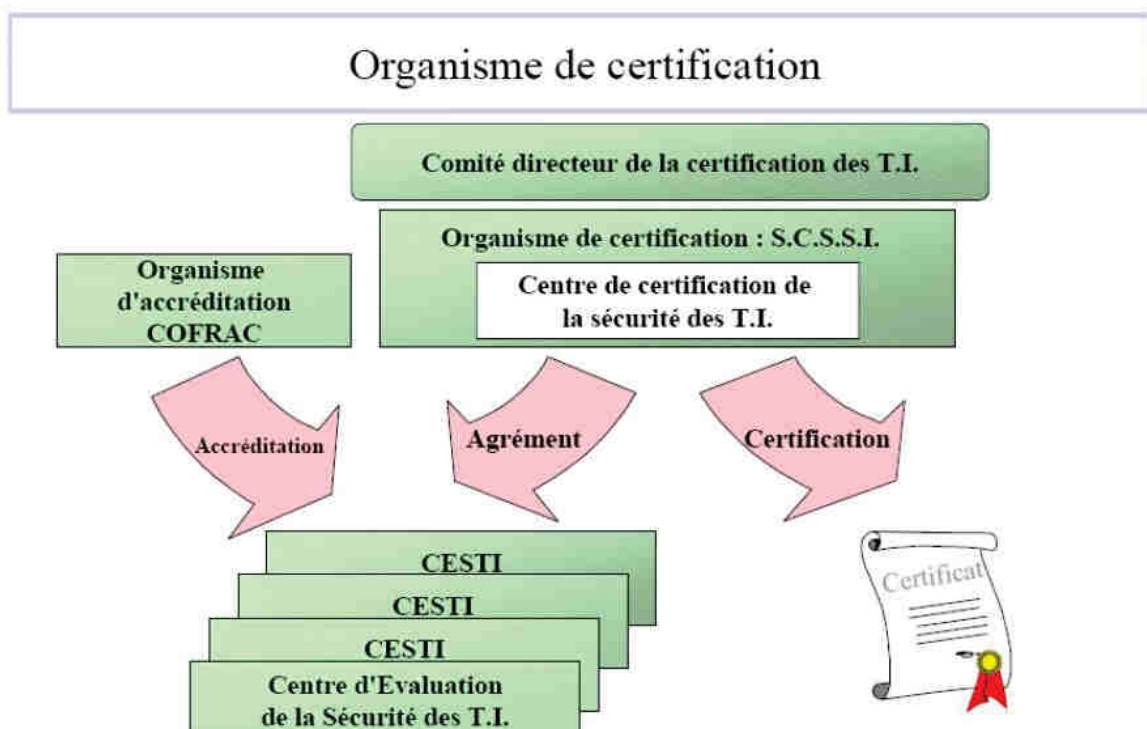
42

Le schéma Français



CDS - Limoges - 2019/2020 - D. Sauveron

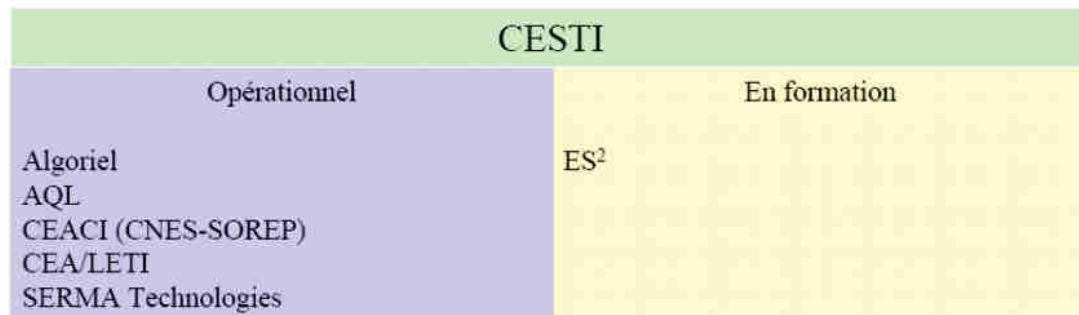
43



CDS - Limoges - 2019/2020 - D. Sauveron

44

Centres d'évaluation



Agréés pour le domaine de la carte à puce :
CEACI,
CEA/LETI,
SERMA Technologies.

CDS - Limoges - 2019/2020 - D. Sauveron

45

Un environnement international...

Organismes de certification:

BSI	Allemagne
CESG	Royaume Uni
DSD	Australie
CSE	Canada
NIST et NSA	États-Unis
SCSSI	France

CDS - Limoges - 2019/2020 - D. Sauveron

46

Reconnaissance mutuelle



Mars 1998

Accord SOGIS

Senior Officials Group for Information Security of the European Commission

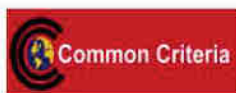


- Accord entre 12 pays européens :
Allemagne, Espagne, Finlande, France, Grèce, Italie, Pays-Bas, Portugal, Norvège, Royaume Uni, Suède et Suisse
- Organismes de certification qualifiés
Allemagne, France et Royaume Uni
- Certificat ITSEC E1 à E6 et CC EAL1 à EAL7

CDS - Limoges - 2019/2020 - D. Sauveron

47

Reconnaissance mutuelle



Mai 2000

Arrangement harmonisé

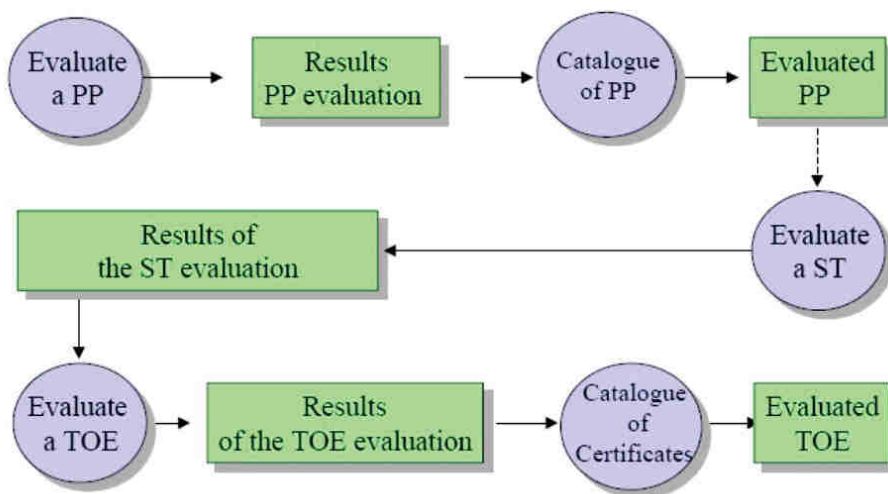


- Accord entre 13 pays :
Allemagne, Australie&Nouvelle Zélande, Canada, Espagne, États-Unis, Finlande, France, Grèce, Italie, Pays-Bas, Norvège, Royaume Uni et Suède
- Organismes de certification qualifiés
BSI (Allemagne), DSD (Australie), CSE (Canada), NIST/NSA (États-Unis), SCSSI (France) et CESG (Royaume Uni)
- Organismes de certification commerciaux
- Certificat CC EAL1 à EAL4

CDS - Limoges - 2019/2020 - D. Sauveron

48

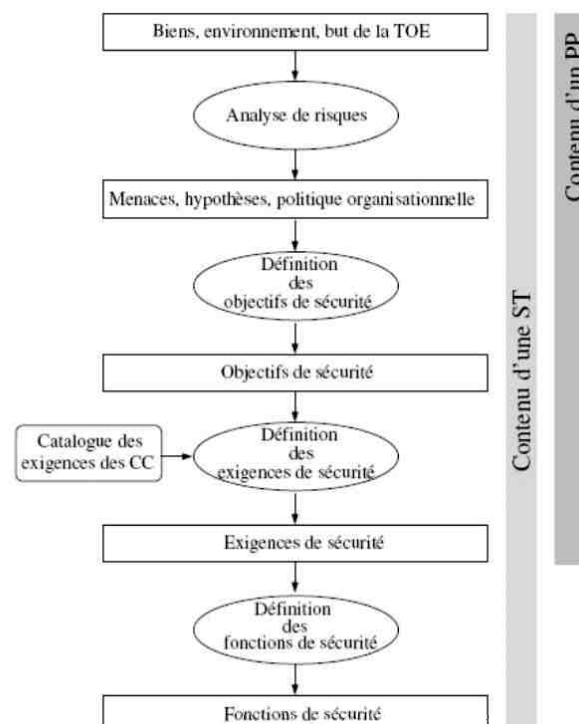
CC evaluations and their results



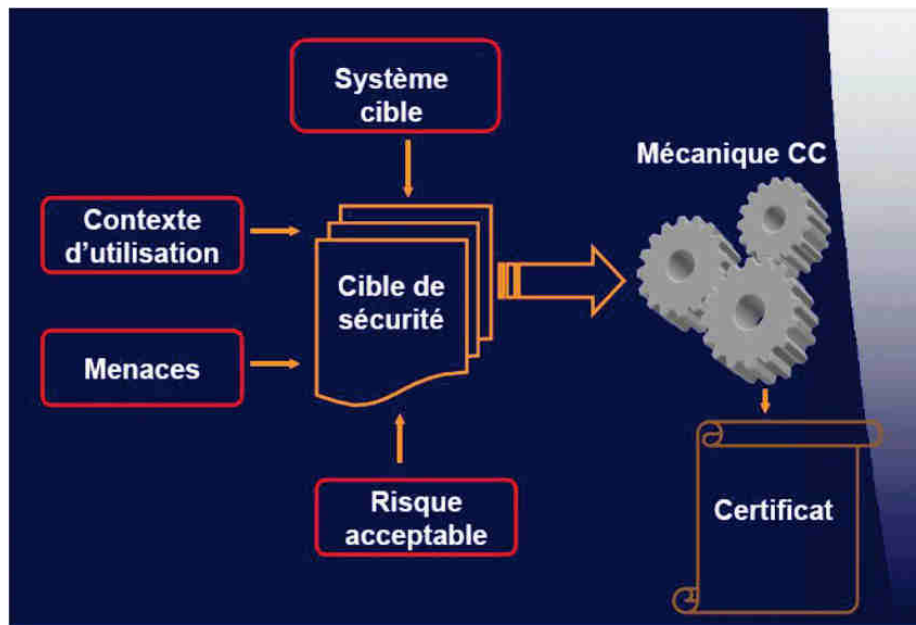
CDS - Limoges - 2019/2020 - D. Sauveron

49

Contenu d'une ST et d'un PP



50



Un certificat



Quelques chiffres !

Retour d'expérience sur l'évaluation d'un VPN

⊕ Contexte :

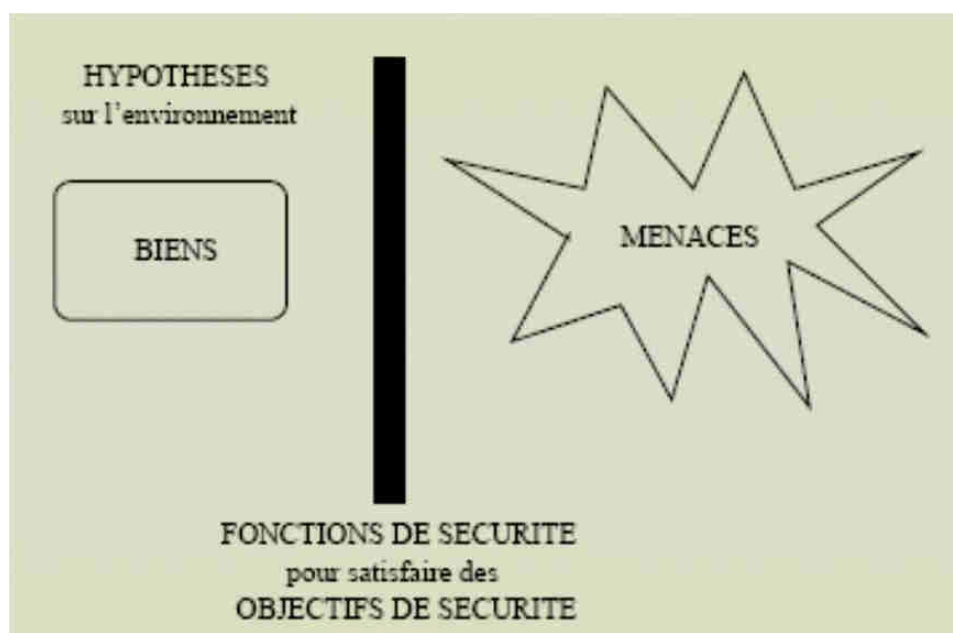
- Sollicitation pour les besoins gouvernementaux
- Évaluation CC EAL2+ (Qualification standard)
- Périmètre le plus englobant possible
- Certification body : DCSSI
- Centre d'évaluation : AQL
- Obtention du certificat en décembre 2002

⊕ Caractéristiques du projet

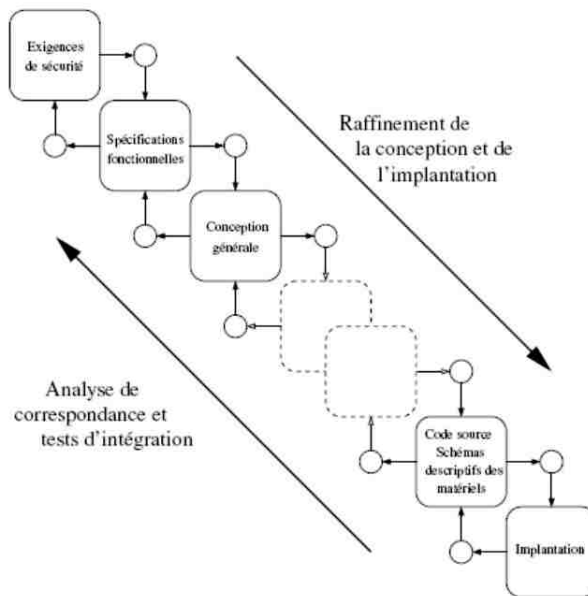
- Durée : 7 mois
- Coût : 200K€

Les bases des critères communs

- Critères Communs (ISO15408)



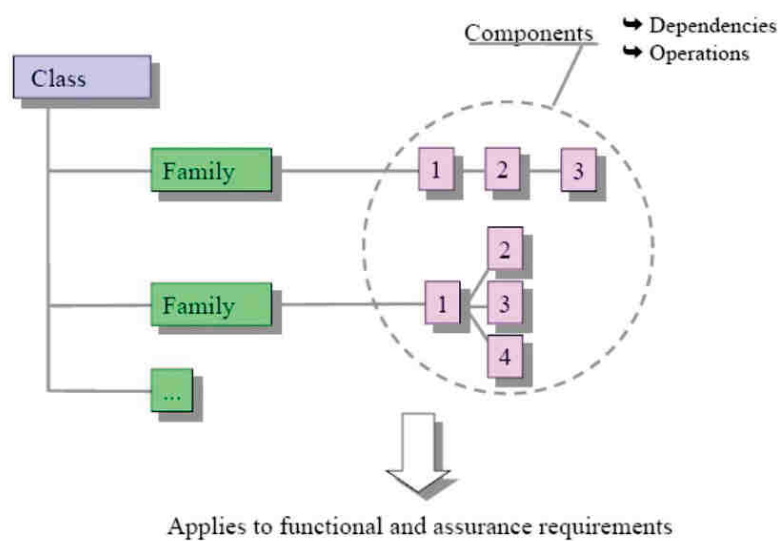
Modèle de développement d'une TOE



CDS - Limoges - 2019/2020 - D. Sauveron

55

Requirements : hierarchical structure



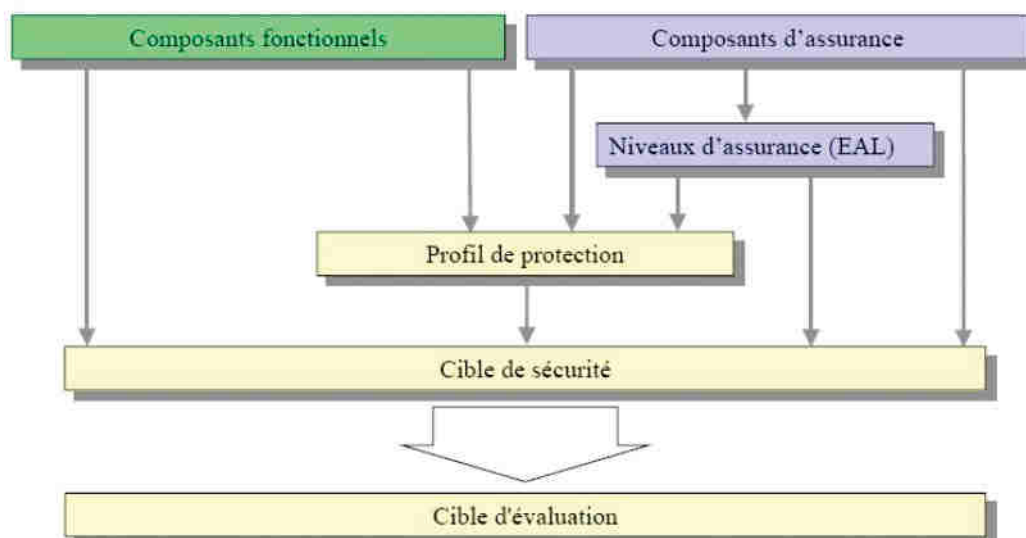
CDS - Limoges - 2019/2020 - D. Sauveron

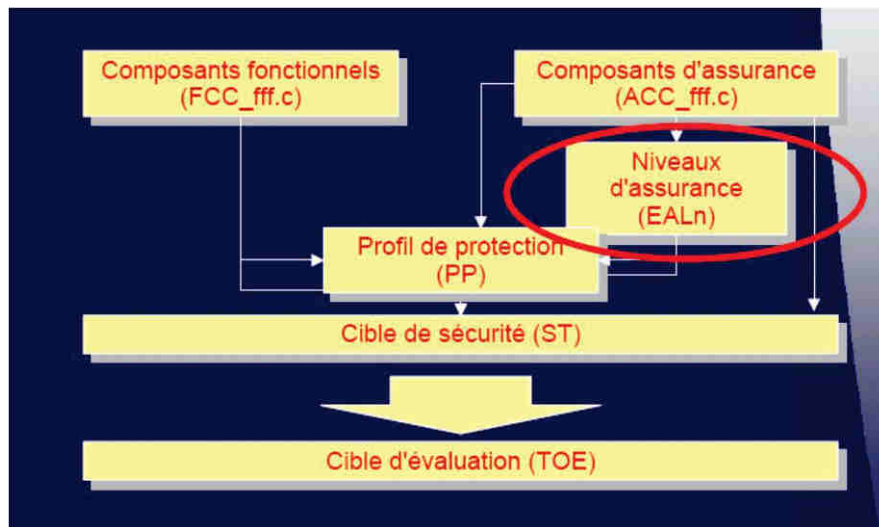
56

CC Catalogue : definitions

- **Class** : A grouping of families that share a common focus.
- **Family** : A grouping of components that share security objectives but may differ in emphasis or rigour.
- **Component** : The smallest selectable set of elements that may be included in a PP, an ST, or a package.
- **Package** : A reusable set of either functional or assurance components (eg. an EAL), combined together to satisfy a set of identified security objectives.

Utilisation des catalogues



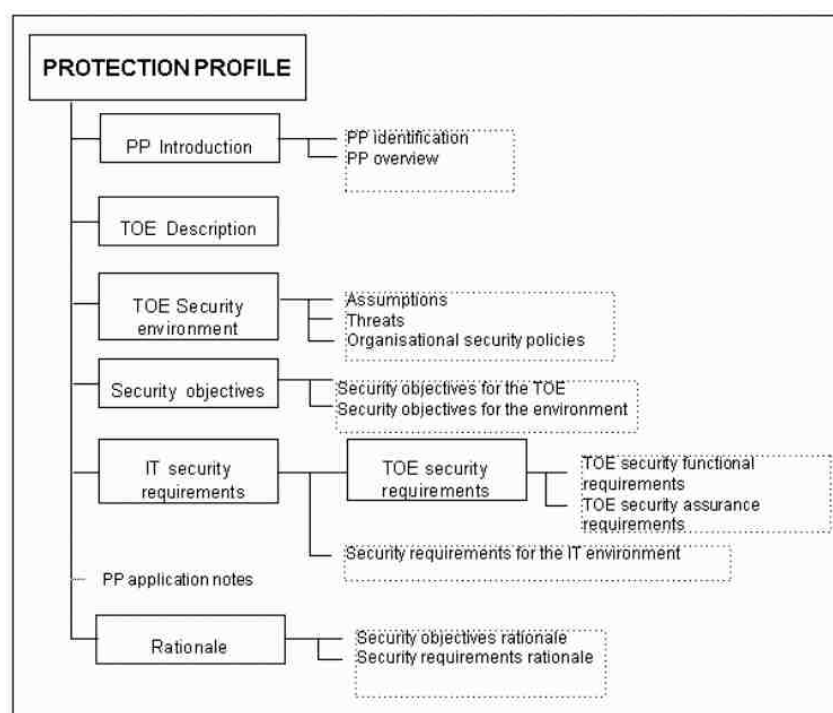
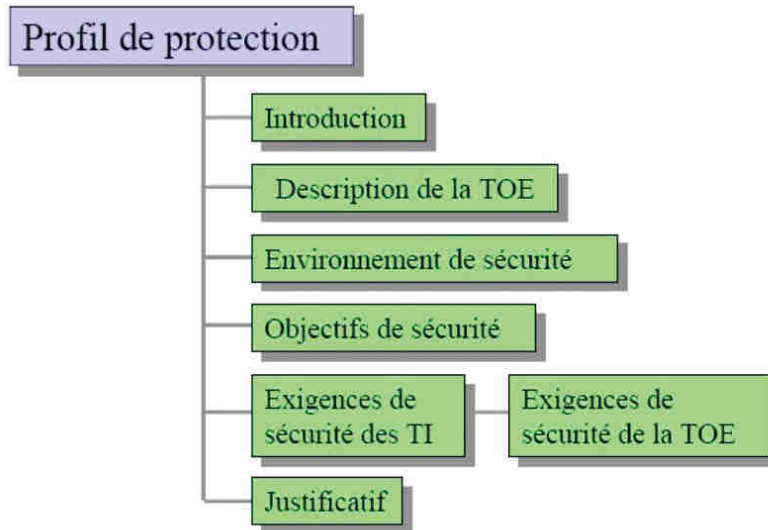


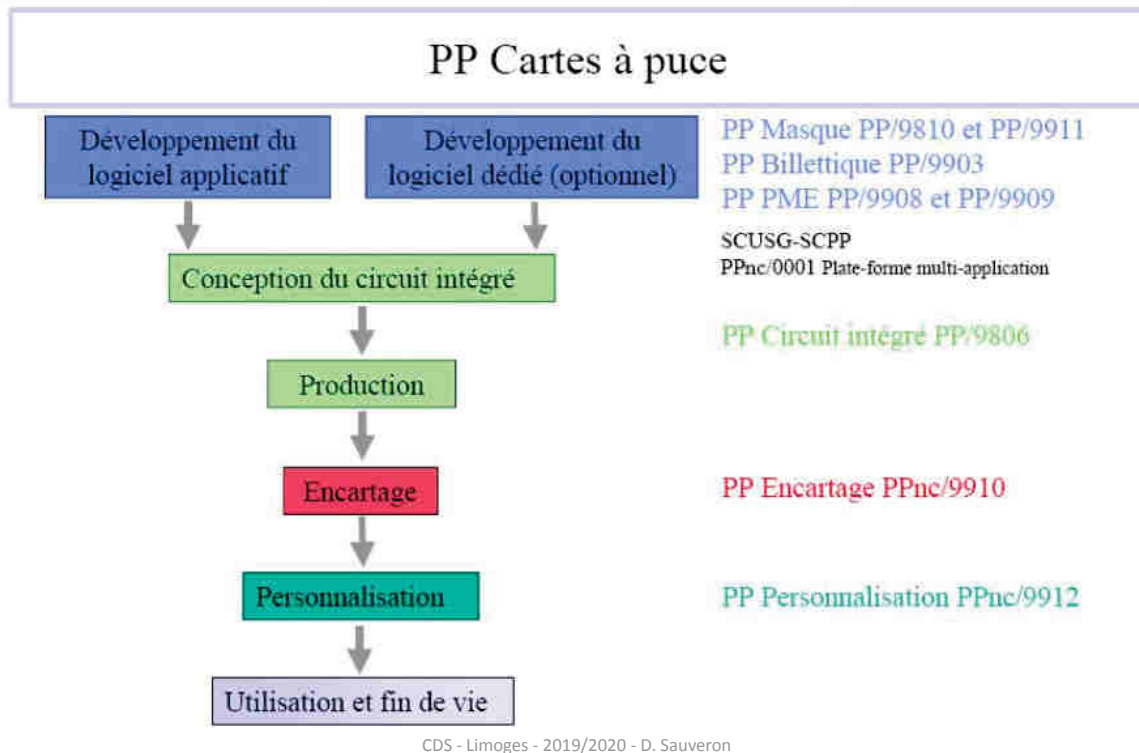
La cible de sécurité

⊕ La clé de voûte de l'évaluation : la cible de sécurité (ST)

⊕ Architecture de la ST

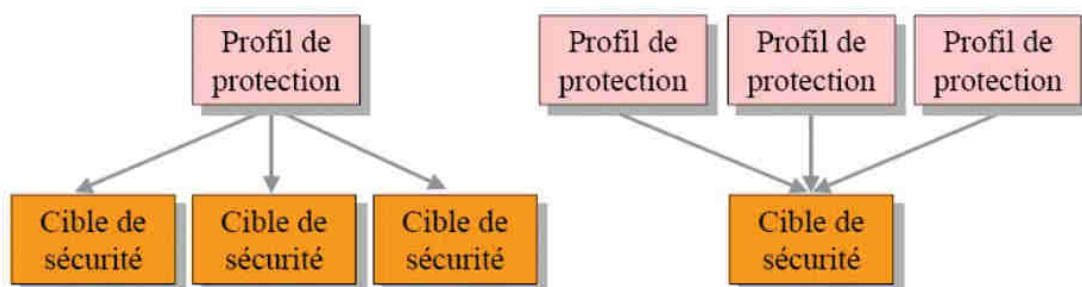
- Niveau d'assurance : définit la profondeur de l'évaluation
- Périmètre du système évalué (TOE)
- Environnement de sécurité
 - Hypothèses d'utilisation
 - Menaces prises en compte
- Objectifs de sécurité : doivent couvrir toutes les menaces
- Fonctions de sécurité : doivent couvrir tous les objectifs
- Référence à une cible type : le Profil de Protection (PP)





63

Utilisation des profils de protection



64

PP Masque

PP/9810: Smartcard embedded software

EAL4+ (ADV_IMP.2, ALC_DVS.2, AVA_VLA.4)

Schlumberger



PP/9911: Smartcard integrated circuit with embedded software

EAL4+ (ADV_IMP.2, ALC_DVS.2, AVA_VLA.4)

EuroSmart



PP Billetique

PP/9903: Carte à puce billettique avec et sans contact

EAL4+ (ADV_IMP.2, AVA_VLA.4)

SNCF, RATP



PP PME

PP/9908: Intersector Electronic Purse and Purchase Device
Version for Pilot Schemes only

EAL1+ (AVA_VLA.2)

Banque de France, EuroSmart, GIE Cartes Bancaires CB



PP/9909: Intersector Electronic Purse and Purchase Device

EAL4+ (ADV_IMP.2, ALC_DVS.2, AVA_VLA.4)

Banque de France, EuroSmart, GIE Cartes Bancaires CB



CDS - Limoges - 2019/2020 - D. Sauveron

67

PP Composant pour carte a puce

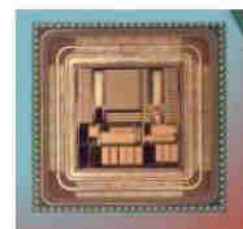
PP/9806: Smartcard integrated circuit

EAL4+ (ADV_IMP.2, ALC_DVS.2, AVA_VLA.4)

EuroSmart,

Motorola, Philips, Siemens, STMicroelectronics,

Texas-Instruments



CDS - Limoges - 2019/2020 - D. Sauveron

68

PP Encartage et personnalisation

PPnc/9910: SmartCard embelling Sites

Paquet d'assurance spécifique

AFPC (Association des Fabricants et des Personnaliseurs de Carte)



PPnc/9912: SmartCard personalization Sites

Paquet d'assurance spécifique

GIP CPS



CDS - Limoges - 2019/2020 - D. Sauveron

69

PP Terminaux



PP/9907: Automates bancaires

EAL4+ (AVA_VLA.3)

Bull

Dassault A.T.

IBM

NCR

Siemens Nixdorf

Wang Global

PP/0002: Lecteur transactionnel de cartes à puce

EAL4+ (ADV_IMP.2, AVA_VLA.3)

Cyber-COMM

CDS - Limoges - 2019/2020 - D. Sauveron

70

PP Echanges de données informatisées

PP/9802: Transactions portant sur des données non confidentielles

EAL3+ (ADV_IMP.1, ADV_LLD.1, ALC.TAT.1)

Ministère de l'Economie, des Finances et de l'Industrie

PP/9803: Transactions portant sur des données confidentielles

EAL3+ (ADV_IMP.1, ADV_LLD.1, ALC.TAT.1)

Ministère de l'Economie, des Finances et de l'Industrie



CDS - Limoges - 2019/2020 - D. Sauveron

71

PP Firewalls

PP/9904: Firewall à exigences réduites

EAL4+ (ADV_IMP.2, AVA_CCA.1, AVA_VLA.3)

DGA

PP/9905: Firewall à exigences élevées

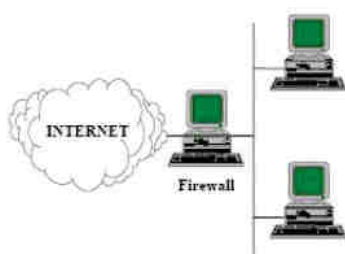
EAL5+ (ALC_FLR.2, AVA_VLA.4)

DGA

PP/9906: Passerelle filtrante de sécurité configurable

EAL5

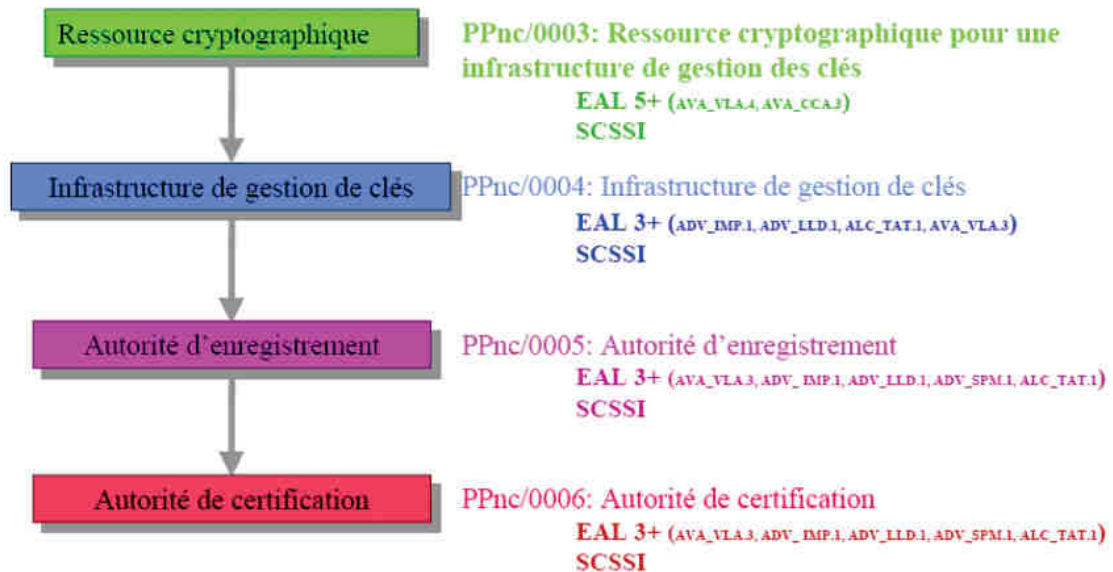
DGA



CDS - Limoges - 2019/2020 - D. Sauveron

72

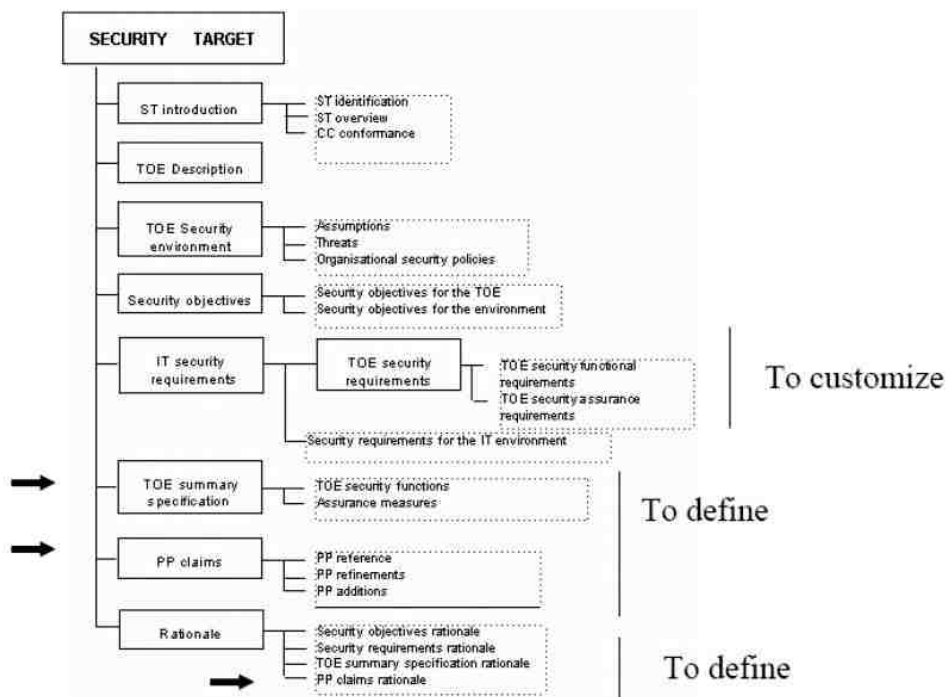
PP Infrastructure de Gestion de Clés



CDS - Limoges - 2019/2020 - D. Sauveron

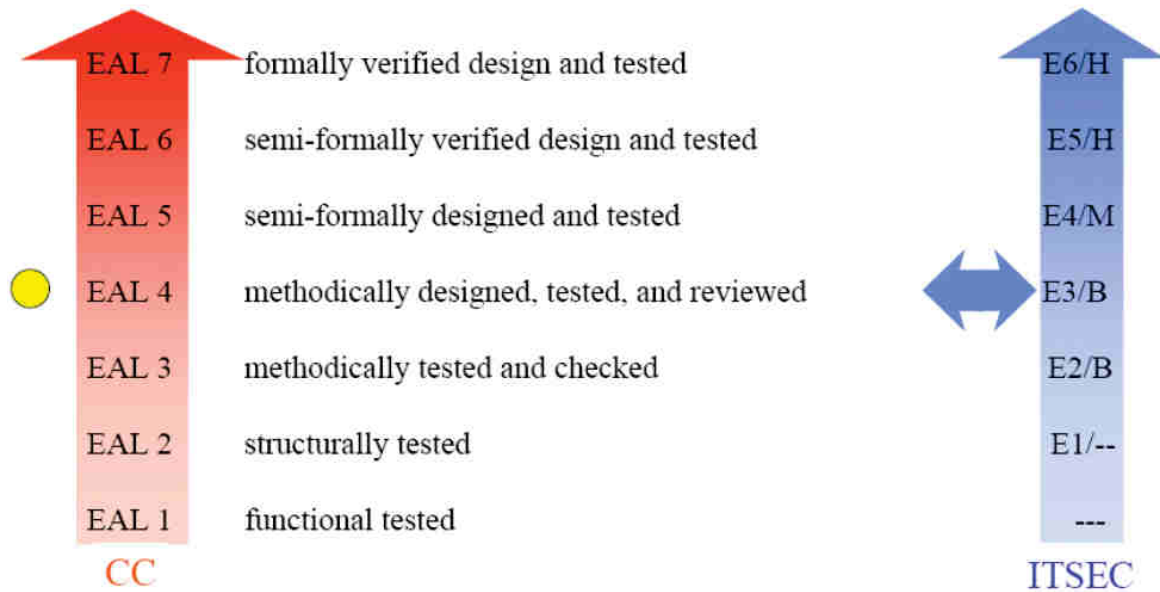
73

Cible de sécurité

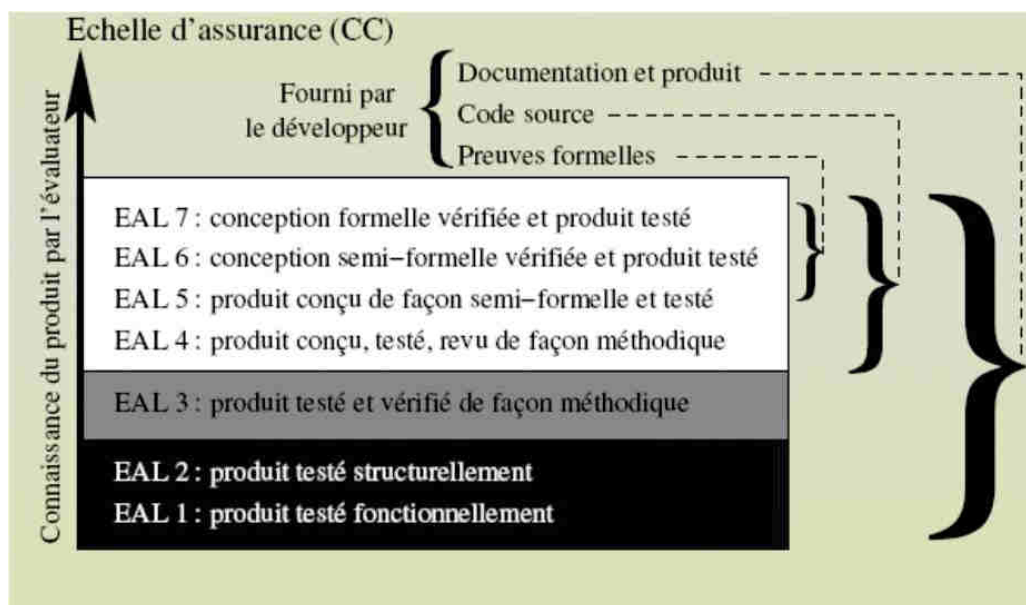


CDS - Limoges - 2019/2020 - D. Sauveron

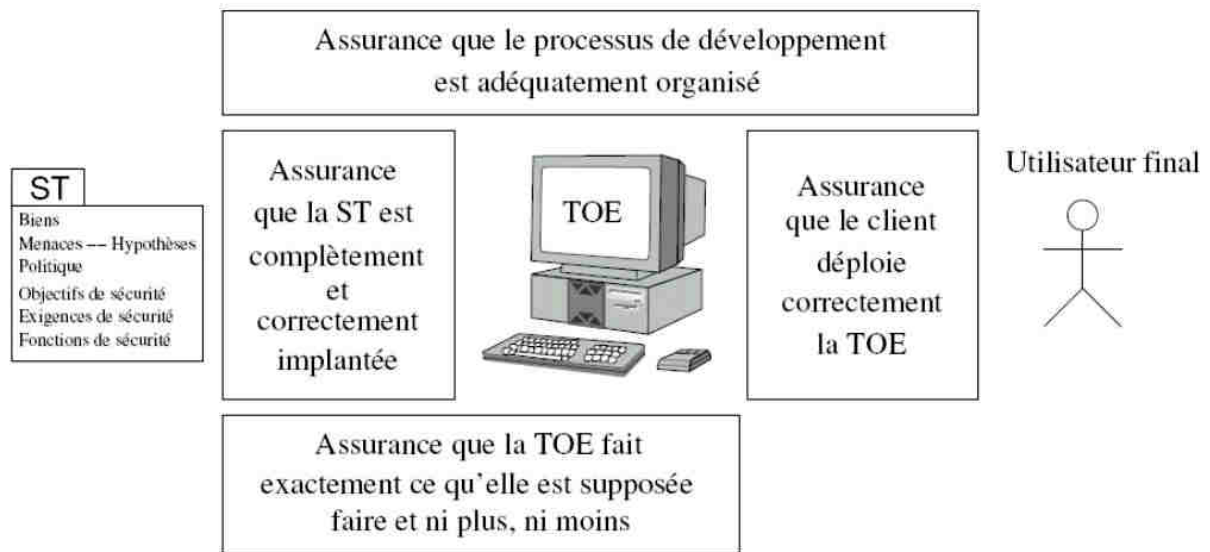
74



L'échelle d'assurance



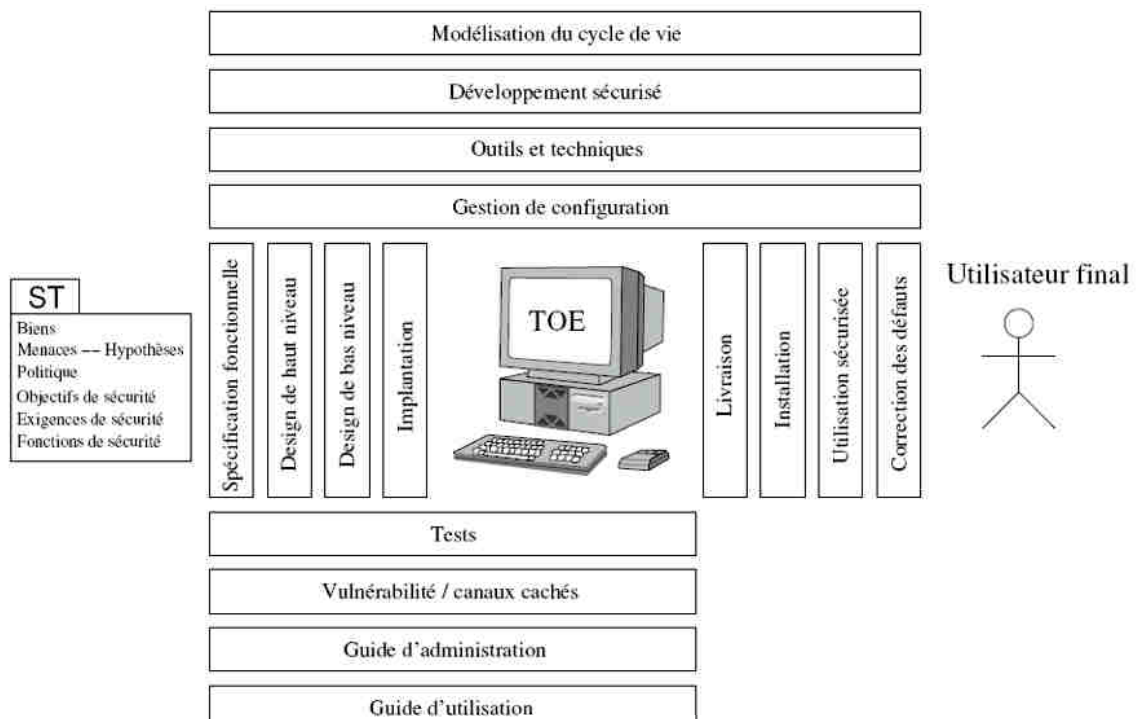
L'assurance des Critères Communs



CDS - Limoges - 2019/2020 - D. Sauveron

77

Les exigences d'assurance des Critères Communs

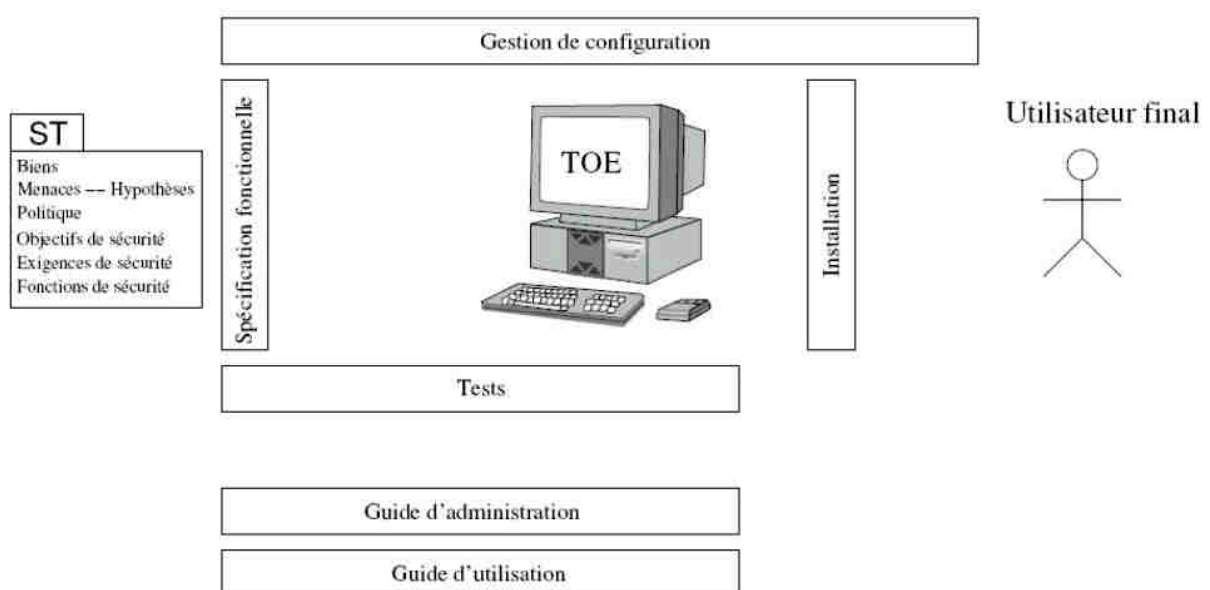


78

Résumé des EALs

Assurance class	Assurance Family	Assurance Components by						
		Evaluation Assurance Level						
		EAL1	EAL2	EAL3	EAL4	EAL5	EAL6	EAL7
Configuration management	ACM AUT				1	1	2	2
	ACM CAP	1	2	3	4	4	5	5
	ACM SCP			1	2	3	3	3
Delivery and operation	ADO DEL		1	1	2	2	2	3
	ADO IGS	1	1	1	1	1	1	1
Development	ADV FSP	1	1	1	2	3	3	4
	ADV HLD		1	2	2	3	4	5
	ADV IMP				1	2	3	3
	ADV INT					1	2	3
	ADV LLD				1	1	2	2
	ADV RCR	1	1	1	1	2	2	3
	ADV SPM				1	3	3	3
	ADV SPM				1	3	3	3
Guidance documents	AGD ADM	1	1	1	1	1	1	1
	AGD USR	1	1	1	1	1	1	1
Life cycle support	ALC DVS			1	1	1	2	2
	ALC FLR							
	ALC LCD				1	2	2	3
	ALC TAT				1	2	3	3
Tests	ATE COV		1	2	2	2	3	3
	ATE DPT			1	1	2	2	3
	ATE FUN		1	1	1	1	2	2
	ATE IND	1	2	2	2	2	2	3
Vulnerability assessment	AVA CCA					1	2	2
	AVA MSU			1	2	2	3	3
	AVA SOF		1	1	1	1	1	1
	AVA VLA		1	1	2	3	4	4

Les exigences d'assurance des Critères Communs pour le niveau EAL1



EAL 1

- *Could be used for an evaluation without the developer*

- TOE security functions analysis
- TOE functional specifications and interfaces
- Security functions independent testing

EAL 2

- *Low-level independent evaluation*

- TOE security functions analysis
- TOE functional specification and interfaces
- TOE sub-systems high-level design
- Review of security functions black-box tests done by the developer
- Obvious vulnerability assessment

81

EAL 3

- *Moderate level evaluation*

- Grey-box testings
- Independent confirmation of a selected sample of developer tests results
- Search for vulnerabilities justified by the developer
- Development environment control
- TOE configuration management

EAL 4

- ***Complete white-box evaluation***

- TOE modules low-level design
- Subset of implementation representation
- Independent search for vulnerabilities
- Conformance of Development process against a life-cycle model
- Tools identification
- Automated configuration management

EAL 5

- ***High level of assurance obtained through a rigorous method of development***

- All implementation analyses
- Formal model and semi-formel presentation of functional specification and high-level design
- Semi-formel Demonstration of correspondence
- Modular design
- Search for vulnerabilities and resistance to moderate potential attacks
- Covert-channel analysis

EAL 6

- *High-level of risks product evaluation*

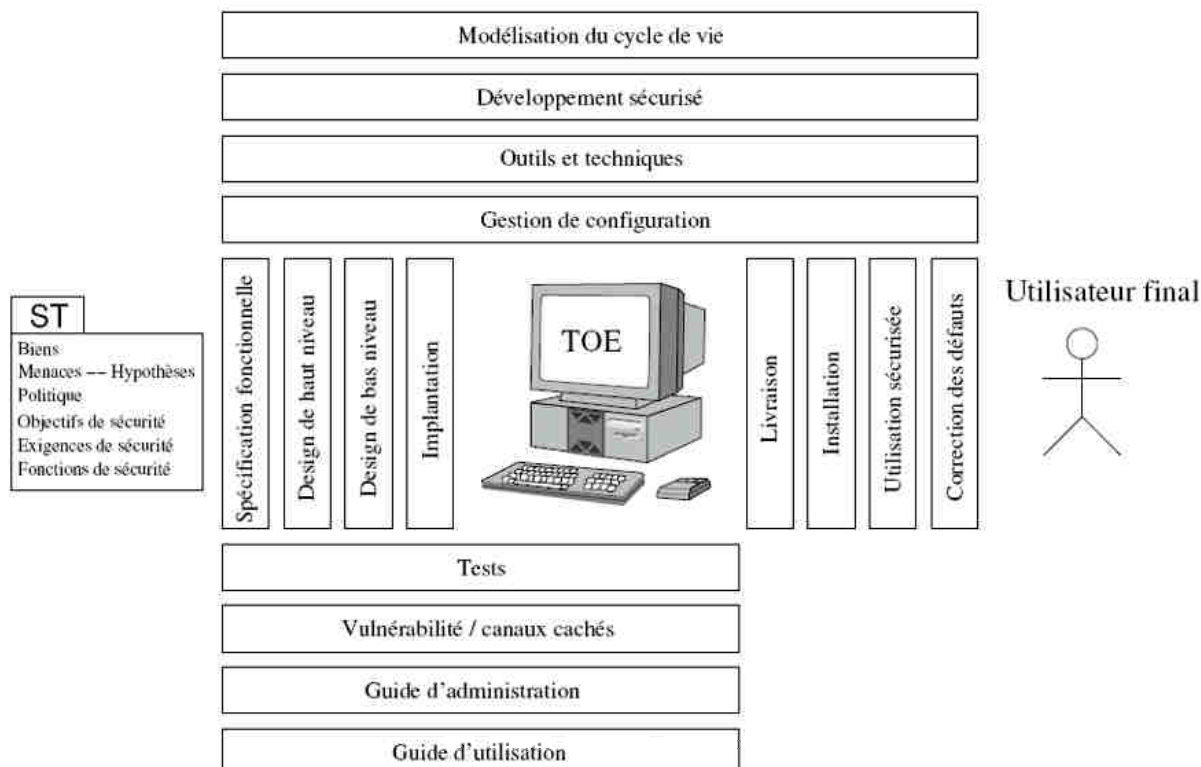
- Modular design and design by successive refinements
- Structured implementation of the TSF
- High-controlled development environment and advanced configuration management
- Systematic search for vulnerabilities and resistance to high potential attacks

EAL 7

- *Very-high level of risks product evaluation*

- Formal presentation of functional specification and high level design
- White-box developer Comprehensive testing
- Complete and independent confirmation of developer tests results
- Minimisation of design complexity

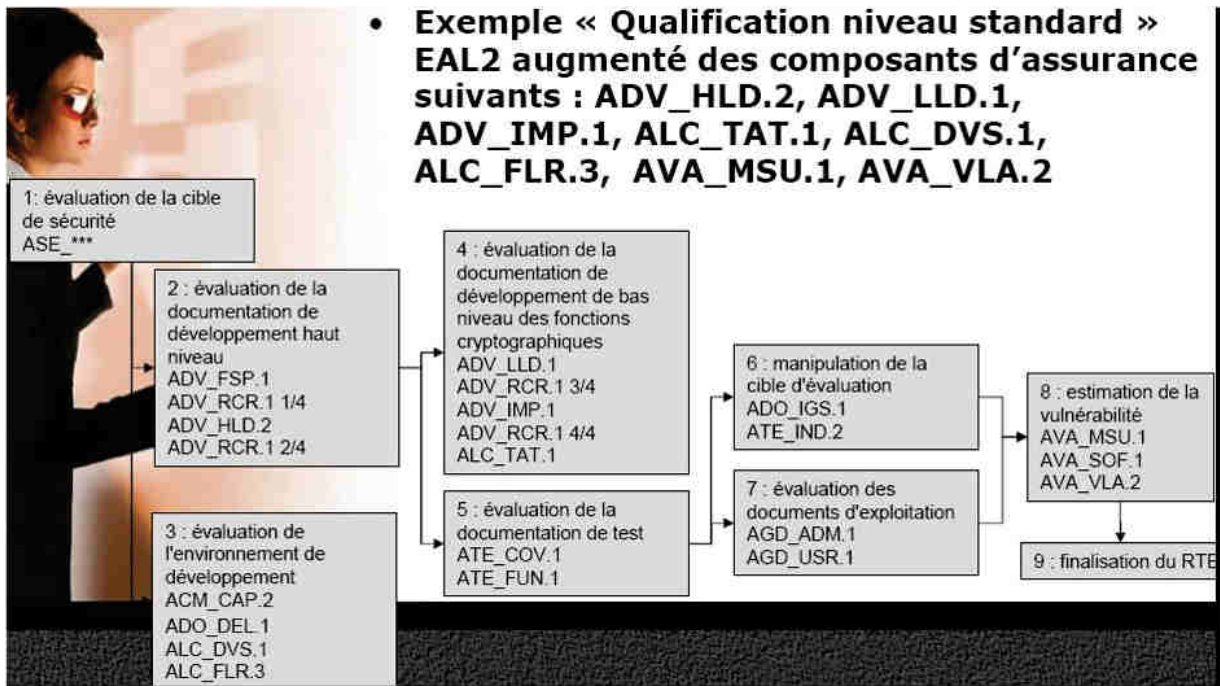
Les exigences d'assurance des Critères Communs pour le niveau EAL7



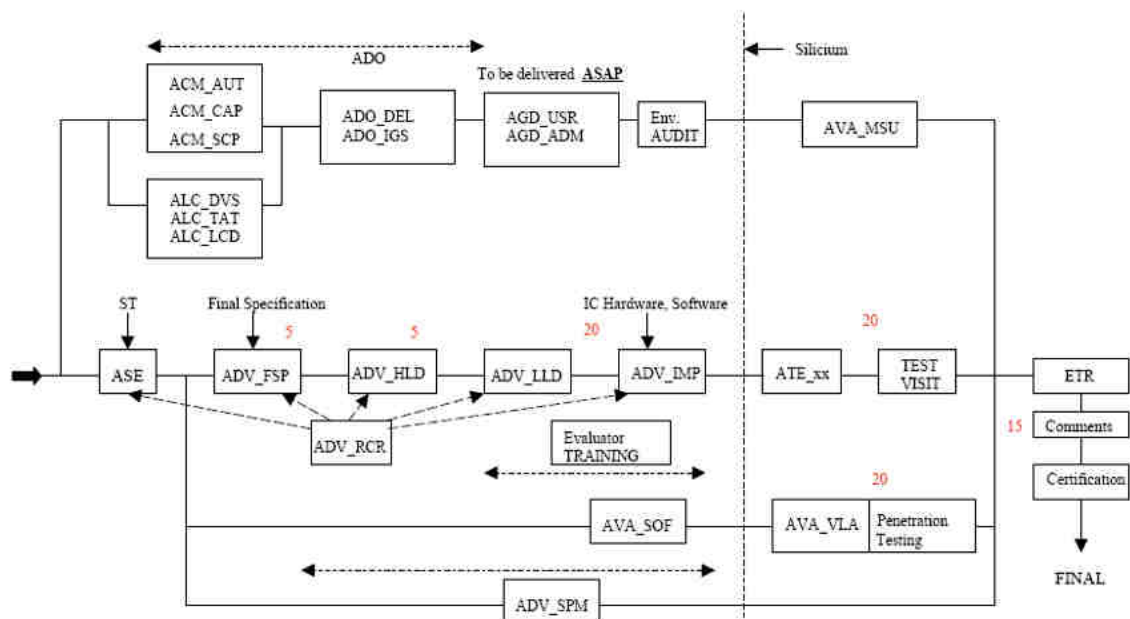
87

Les niveaux de confiance

Classe d'assurance	Famille d'assurance	Composants d'assurance par niveau d'assurance de réévaluation							niveau retenu pour le paquet EAL2+ de qualification au niveau
		EAL 1	EAL 2	EAL 3	EAL 4	EAL 5	EAL 6	EAL 7	
Gestion de configuration	ACM_AUT				1	1	2	2	
	ACM_CAP	1	2	3	4	4	5	5	2
	ACM_SCP			1	2	3	3	3	
	ADO_DEL		1	1	2	2	2	3	1
Livraison et exploitation	ADO_IGS	1	1	1	1	1	1	1	1
	ADV_FSP	1	1	1	2	3	3	4	1
Développement	ADV_HLD		1	2	2	3	4	5	2
	ADV_IMP				1	2	3	3	1
	ADV_INT					1	2	3	
	ADV_LLD				1	1	2	2	1
	ADV_RCR	1	1	1	1	2	2	3	1
	ADV_SPM				1	3	3	3	
	AGD_ADM	1	1	1	1	1	1	1	1
	AGD_USR	1	1	1	1	1	1	1	1
Guides	ALC_DVS			1	1	1	2	2	1
	ALC_FLR								3
Support au Cycle de vie	ALC_LCD				1	2	2	3	
	ALC_TAT				1	2	3	3	1
	ATE_COV		1	2	2	2	3	3	1
	ATE_OPT			1	1	2	2	3	
Tests	ATE_FUN		1	1	1	1	2	2	1
	ATE_IND	1	2	2	2	2	2	3	2
	AVA_CCA					1	2	2	
	AVA_MSU			1	2	2	3	3	1
Estimation des vulnérabilités	AVA_SOF		1	1	1	1	1	1	1
	AVA_VLA		1	1	2	3	4	4	2

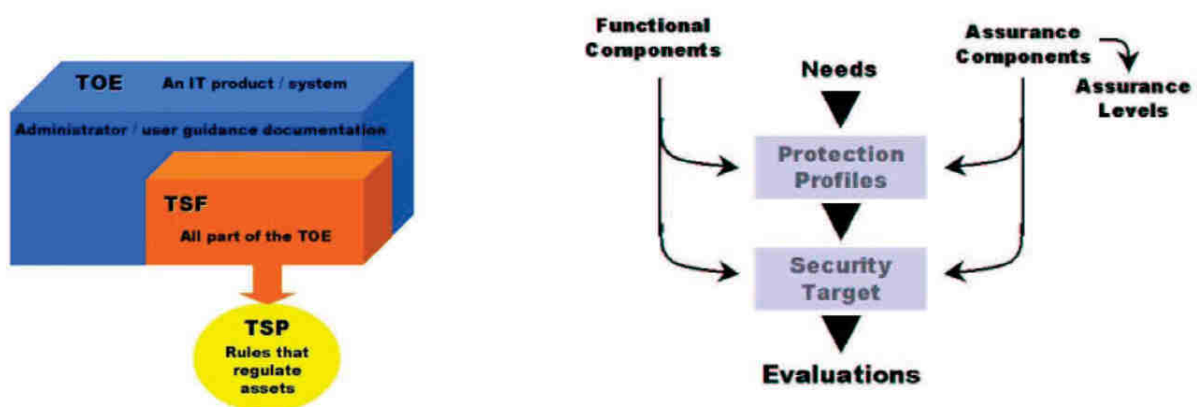


Evaluation Flow



- ON SWAPPE SUR MERLE.PDF p24

Résumé



Offre de produits et PP certifiés

Offre de produits certifiés

Systèmes d'exploitation (19) : *Windows NT 4 SP3, Trusted Solaris, AIX, HP-UX, SCO, ...*

Cartes à puces (18) : *Cartes à mémoire, cartes multiapplicatives (Javacard, Multos), ...*

Lecteurs de cartes (28) : *Certification obligatoire en Allemagne*

Firewall (16) : *Nerwall, Firewall-1, PLX, Gauntlet, Lucent, ...*

Bases de données (7) : *Oracle, Informix, Ingres, ...*

Utilitaires PC (19) : *Contrôle d'accès, utilitaires disques, ...*

Divers (11) : *Chiffrement, PKI, ...*

➔ www.scssi.gouv.fr

Source : Certificats reconnus par le schéma français au 1er Jan. 2000

CDS - Limoges - 2019/2020 - D. Sauveron

93

Certificats : Domaines couverts

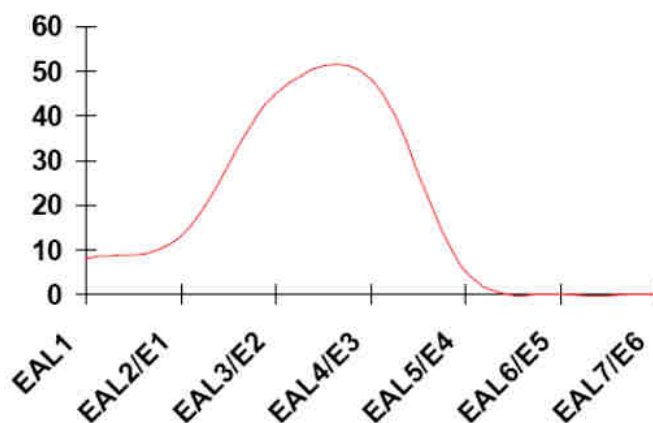


Source : Certificats reconnus par le schéma français au 1er Jan. 2000

CDS - Limoges - 2019/2020 - D. Sauveron

94

Certificats : Répartition par niveau d'assurance

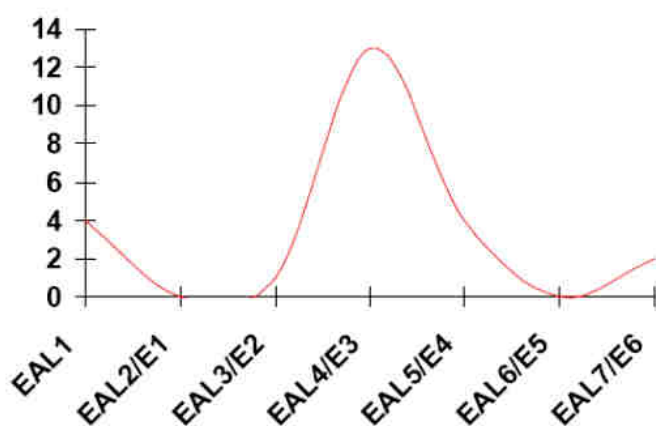


Source : Certificats reconnus par le schéma français au 1er Jan. 2000

CDS - Limoges - 2019/2020 - D. Sauveron

95

Cartes à puce certifiées



Source : Certificats reconnus par le schéma français au 1er Jan. 2000

CDS - Limoges - 2019/2020 - D. Sauveron

96

Offre de Profils de Protection

Cartes à puces : *Composants, applications, sites de production, ...*

Firewalls : *DGA, NIST/NSA*

Messagerie électronique : *SCSSI, MEFI*

Bases de données : *Oracle*

Divers : *Contrôle d'accès, DAB, lecteurs de cartes, ...*

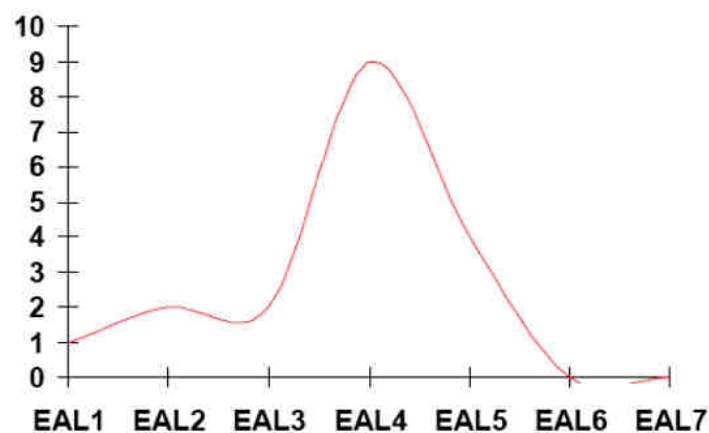
➔ www.scssi.gouv.fr

Source : Certificats reconnus par le schéma français au 1er Jan. 2000

CDS - Limoges - 2019/2020 - D. Sauveron

97

PP : Répartition par niveau d'assurance



Source : Certificats reconnus par le schéma français au 1er Jan. 2000

CDS - Limoges - 2019/2020 - D. Sauveron

98

Un exercice de style : la rédaction de la Cible de sécurité



- ⊕ Choix du périmètre : commercial, réaliste ou rigoriste?
 - Frontière physique de la TOE
 - Hypothèses d'utilisation
 - « Les administrateurs sont de confiance et ont été formés »
 - « Les mots de passe sont aléatoires et sont formés de + de 7 caractères »
 - Menaces contrées
 - « Un individu peut effectuer une attaque en confidentialité sans connaissance des secrets »
- ⊕ Niveau d'assurance :
 - Garder en tête les fondamentaux : EAL2 = Qualité, EAL4 = Sécurité
 - Chercher ce qui se cache derrière le « + » de « EALx+ »
- ⊕ L'avenir : la conformité aux Profils de Protection (PP)

CDS - Limoges - 2019/2020 - D. Sauveron

99

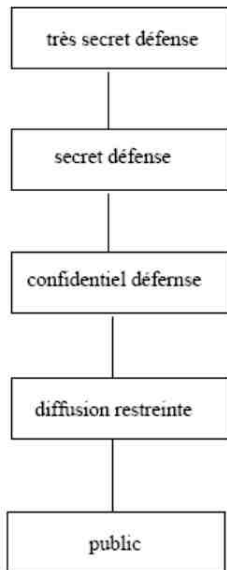
Les limites du processus Critères Communs



- ⊕ La complexité du processus : trop long et trop coûteux
- ⊕ La difficile maintenance des certificats
 - Les procédures de surveillance sont inadaptées aux produits logiciels
 - Import de la notion américaine « assurance continuity »?
- ⊕ La complexité de l'interprétation des résultats d'une évaluation
- ⊕ L'impossibilité de composer les évaluations CC
- ⊕ L'impossibilité d'évaluer les systèmes complexes à un niveau décent

CDS - Limoges - 2019/2020 - D. Sauveron

100



Usage particulier en France des CC

- ⊕ Fondement de la **PRIS** (Politique de Référencement Intersectorielle de Sécurité)
- ⊕ Objectif : fournir à l'administration et au secteur privé un cadre opérationnel cohérent pour le traitement des informations sensibles et classifiées
- ⊕ Méthode : bâtir un catalogue de solutions qualifiées, sur la base de leur évaluation suivant les CC

Sensibilité de l'information	Niveau de qualification requis	Niveau CC requis
Sensible non classifié	Niveau Standard	EAL2+
Classifié Confidentiel Défense	Niveau Renforcé	EAL4+

Critères Communs : les bonnes questions



- Qu : Quelles parties du produit ont été analysées?
R : La TOE est décrite dans la Cible de Sécurité
- Qu : Quelle confiance accorder aux vérifications effectuées?
R : Une totale confiance en terme de qualité (agrément COFRAC)
- Qu : Quels sont les documents publics associés aux certifications?
R : Le rapport de certification et la cible de sécurité
- Qu : Où trouve t-on ces documents?
R : Sur www.commoncriteriaportal.org ou www.ssi.gouv.fr

Classification des attaques

- 🕒 Les attaques se font par rapport aux propriétés non fonctionnelles :
 - 🕒 Confidentialité :
 - 🕒 L'objectif est d'obtenir des informations.
 - 🕒 Intégrité :
 - 🕒 L'objectif est de changer, supprimer ou ajouter des informations.
 - 🕒 Disponibilité :
 - 🕒 L'objectif est de rendre le dispositif inutilisable par suite de très nombreuses requêtes.

103

Les types d'attaques

- 🕒 Les attaques physiques :
 - 🕒 Microprobing
 - 🕒 Electromagnétisme,...
- 🕒 Les canaux cachés :
 - 🕒 Observations précises du comportement en mode en fonction du dispositif,
 - 🕒 Les observations peuvent porter sur le temps, la consommation,...
- 🕒 Les attaques par injection de fautes,
- 🕒 Les attaques logicielles :
 - 🕒 Virus, Chevaux de Troie,...

104

Les attaques

Deux grandes catégories

- Non invasives
 - Le produit reste fonctionnel
- Invasives
 - Le produit est partiellement endommagé

Quelques attaques non invasives

Des attaques non invasives

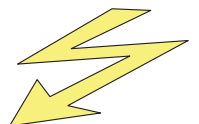
- modification des conditions opérationnelles (Vcc, F)



- Attack on VPP
- Using nail polish
- Card not debited...

- modification de la température
- modification des rayonnements lumineux (UV, rayon X, lumière blanche, IR, ...)

- injection de fautes (glitches, rayonnements lumineux)
 - (injection de fautes sur la JVM : <http://www.cs.princeton.edu/~sudhakar/papers/>)

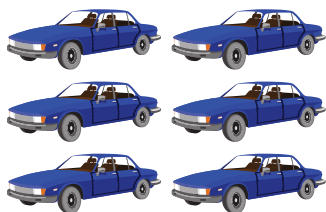


- attaques sur les **canaux cachés** :
 - Temps
 - Consommation électrique
 - Émissions électromagnétiques

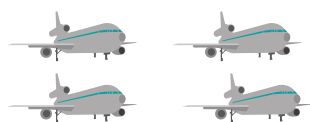
Injection de fautes

Êtes vous prêt ?

- Politique : Les jouets cassés ne sont pas payés

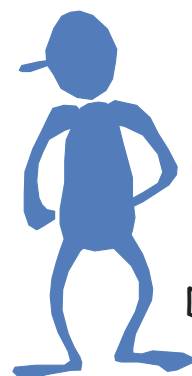


voiture = \$3



avion = \$5

Jack

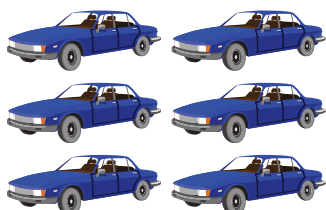


Dino

Dino achète des jouets à Jack

Êtes vous prêt ?

• Politique : Les jouets cassés ne sont pas payés



voiture = \$3

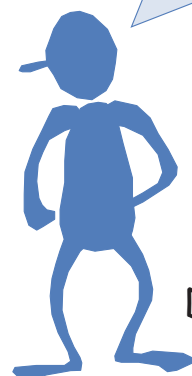


avion = \$5

Jack



Je voudrais acheter
3 avions



Dino

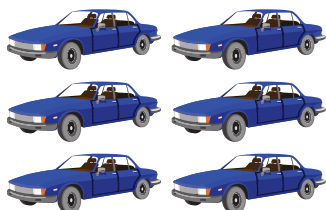
Dino achète des jouets à Jack

CDS - Limoges - 2019/2020 - D. Sauveron

111

Êtes vous prêt ?

• Politique : Les jouets cassés ne sont pas payés

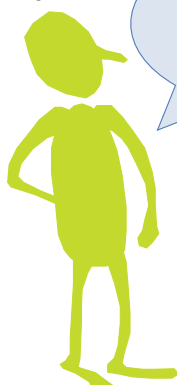


voiture = \$3



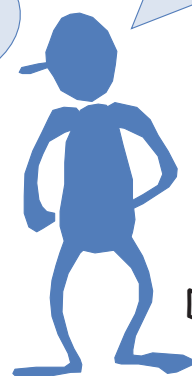
avion = \$5

Jack



Ça fera \$15

Je voudrais acheter
3 avions



Dino

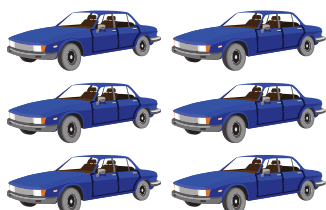
Dino achète des jouets à Jack

CDS - Limoges - 2019/2020 - D. Sauveron

112

Êtes vous prêt ?

• Politique : Les jouets cassés ne sont pas payés



voiture = \$3



avion = \$5

Jack

Ça fera \$15

OK, s'il vous plaît
envoyez le par DHL

Dino

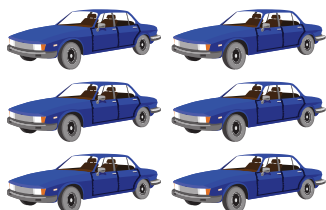
Dino achète des jouets à Jack

CDS - Limoges - 2019/2020 - D. Sauveron

113

Êtes vous prêt ?

• Politique : Les jouets cassés ne sont pas payés



voiture = \$3



avion = \$5

Jack

Comment payez vous ?

OK, s'il vous plaît
envoyez le par DHL

Dino

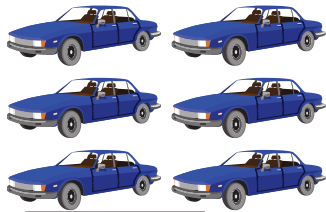
Dino achète des jouets à Jack

CDS - Limoges - 2019/2020 - D. Sauveron

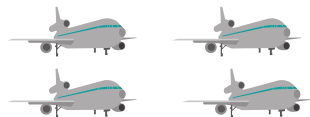
114

Êtes vous prêt ?

• Politique : Les jouets cassés ne sont pas payés



voiture = \$3



avion = \$5

Jack

Comment payez vous ?

J'enverrai \$15
en mandat postal

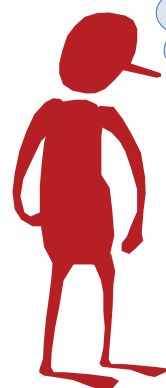
Dino

Dino achète des jouets à Jack

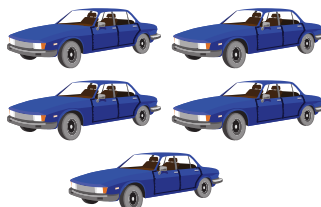
CDS - Limoges - 2019/2020 - D. Sauveron

115

Le facteur veut savoir ce que Dino a acheté pour \$15

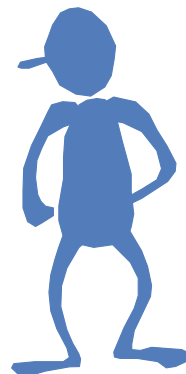


Qu'est ce qu'il a
acheté pour \$15 ?



Méchant facteur

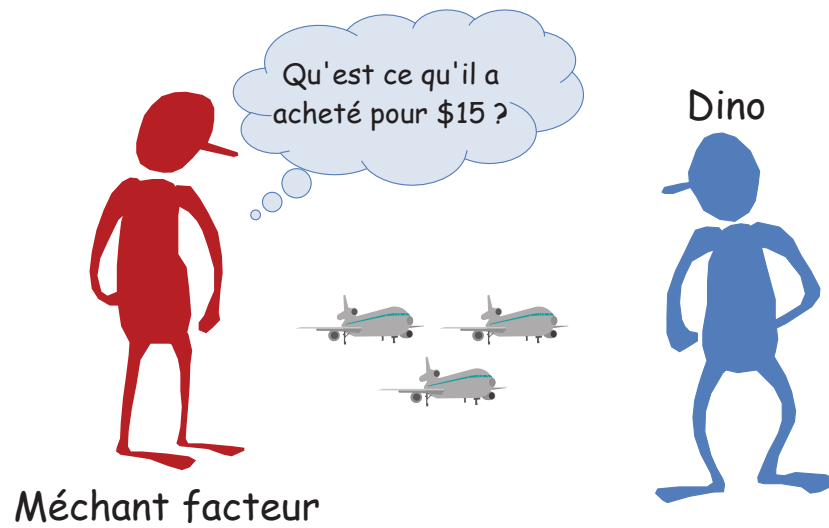
Dino



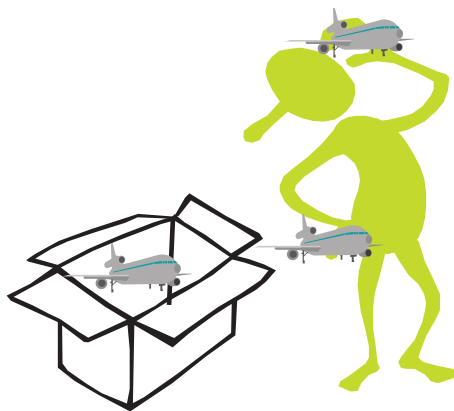
CDS - Limoges - 2019/2020 - D. Sauveron

116

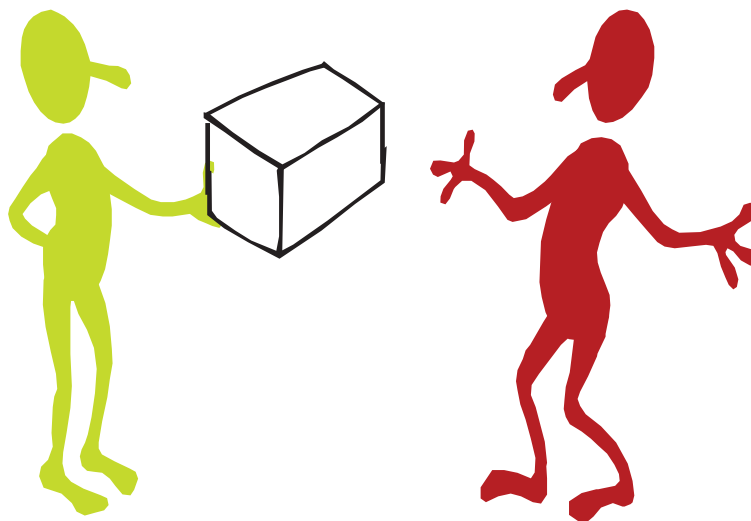
Le facteur veut savoir ce que Dino a acheté pour \$15



Pendant ce temps, Jack prépare le colis DHL



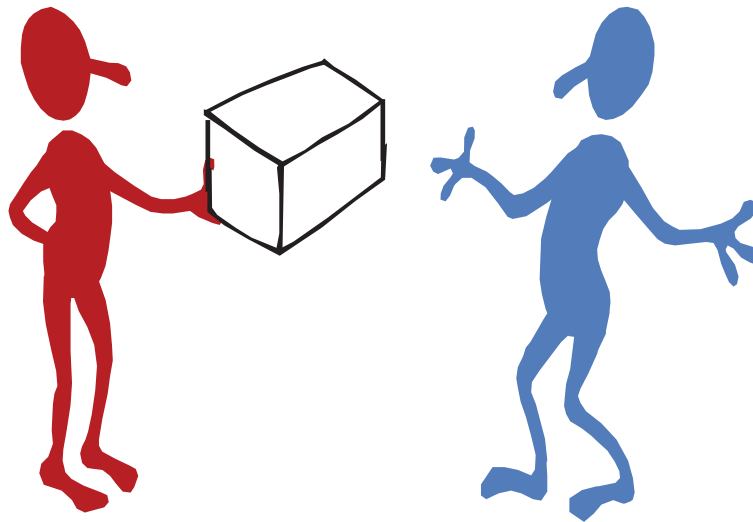
Et le donne au facteur



Qui tape dedans assez fort pour casser **un** jouet



Et le donne à Dino



CDS - Limoges - 2019/2020 - D. Sauveron

121

Une semaine plus tard, il surveille le mandat postal de Dino...



- La leçon apprise : Les attaques en faute peuvent aussi permettre d'extraire des secrets de tokens!

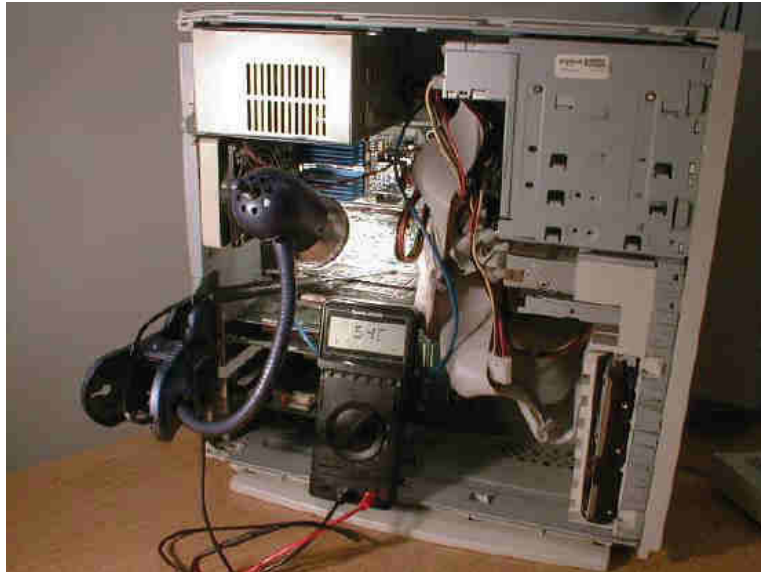
- Les fautes matérielles peuvent être venir de diverses sources :
 - Glitches de tension, faisceau lumineux, faisceau laser ...
- Utilisable sur la signature RSA, le chiffrement DES, ...
- Il y a des détecteurs lasers embarqués sur les puces récentes !
 - (voir thèse d'Alexandre SARAFIANOS)

CDS - Limoges - 2019/2020 - D. Sauveron

122

Attaques combinées (software/hardware)

- Pour forger des pointeurs (réaliser une confusion de types) et ensuite ...



CDS - Limoges - 2019/2020 - D. Sauveron

123

Les canaux cachés

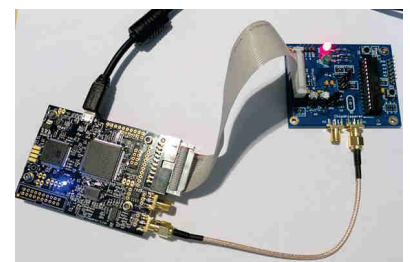
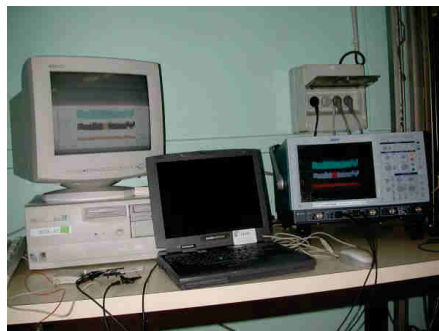
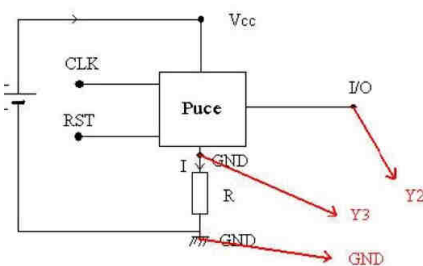
On parle aussi de canaux auxiliaires

Canaux cachés

- le temps d'exécution
 - nombre de cycles d'une instruction ou d'un algorithme
- la consommation de courant
 - Les modifications rapides de la tension et de l'intensité du courant au sein du même composant sont à la base des émissions du circuit car ils conduisent des courants RF à l'intérieur et à l'extérieur du chip.
- les émissions électromagnétiques
 - Les courants RF entraînent un rayonnement électromagnétique.
- le son
- les émissions de photons

Le matériel pour les attaques par canaux cachés

- **Mesure :**
 - du temps d'exécution
 - de la consommation électrique
 - des émissions électromagnétiques
 - du son
- **Équipements**
 - un oscilloscope numérique,
 - un pc équipé de cartes d'acquisition et de logiciels mathématiques pour le traitement des données



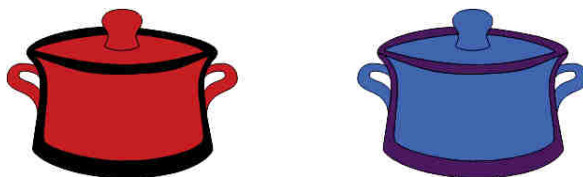
La timing attack

La timing attack

- Cette attaque consiste à mesurer le temps d'exécution d'un algorithme.
 - Elle révèle des informations sur les opérations et/ou les opérandes.
 - Elle nécessite souvent :
 - un grand nombre d'exécution à messages choisis,
 - un traitement statistique des résultats obtenus.

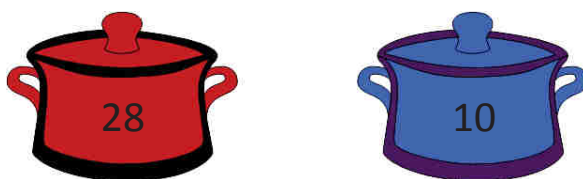
Un jeu

- Mettez 28 dans un des pots et 10 dans l'autre :



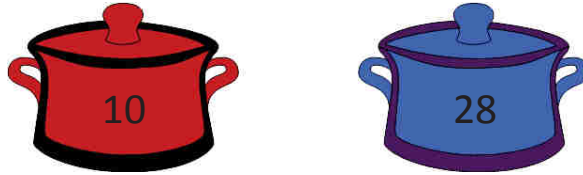
Un jeu

- Mettez 28 dans un des pots et 10 dans l'autre :



Un jeu

- Mettez 28 dans un des pots et 10 dans l'autre :



Un jeu

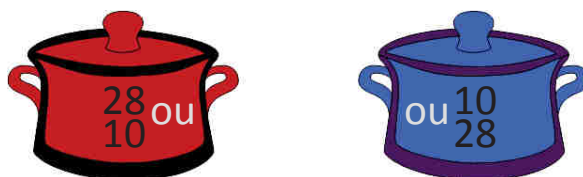
- Mettez 28 dans un des pots et 10 dans l'autre :



- Je vous demande de :
 - multiplier le contenu du pot rouge **par 7**
 - multiplier le contenu du pot bleu **par 10**
 - d'**additionner** les deux résultats
 - de me dire si la somme est **paire** ou **impaire**.

Un jeu

- Mettez 28 dans un des pots et 10 dans l'autre :



- Je vous demande de :
 - multiplier le contenu du pot rouge par 7
 - multiplier le contenu du pot bleu par 10
 - d'additionner les deux résultats
 - de me dire si la somme est paire ou impaire.
- Est-ce que votre réponse est suffisante pour révéler le contenu de chaque pot ?

Est-ce que ce jeu peut m'aider?

- Et bien, normalement non :

$$28 \times 7 + 10 \times 10 = 296$$

est un nombre pair

et

$$10 \times 7 + 28 \times 10 = 350$$

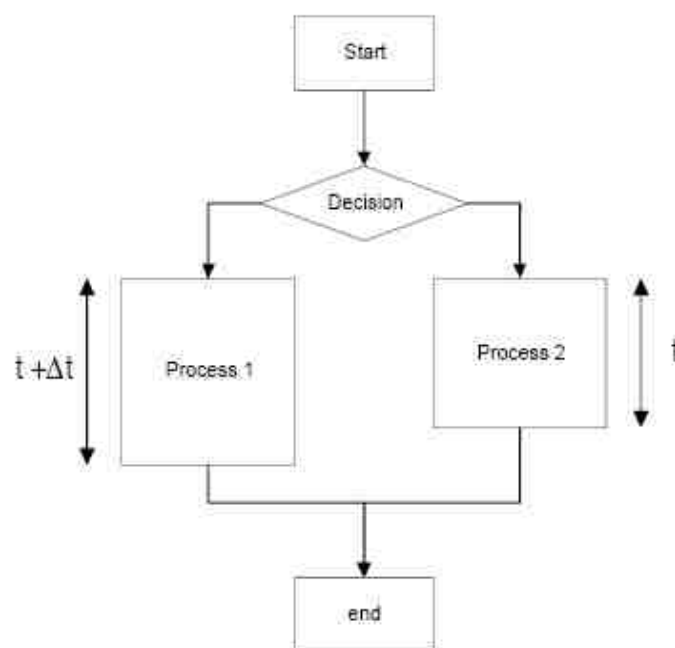
est aussi un nombre pair...

- Pourtant, juste en observant le temps pris pour donner la réponse (le calcul mental conduisant à 296 est plus compliqué que celui conduisant à 350), on peut dire quelles valeurs étaient dans quels pots !

Conclusion

- Une observation externe du temps de calcul d'une carte peut conduire à la fuite d'informations secrètes vers l'extérieur (par exemple des clés, des PINs, etc.).
- Les **timing attacks** sont apparues au début des années 1990.
- La leçon apprise : Les logiciels actuels pour cartes s'exécute en temps constant.

La « timing attack »



La “timing attack” sur le PIN (1/2)

```
for ( i = 0 ; i <= 7; i++)
    if ( pinCarte [ i ] != pinPresente [ i ] )
        return false ;
return true ;
```

 pinCarte
1,3,2,4,250,7,9,134

- Présentons un PIN quelconque et déroulons le programme :

- cas du premier octet FAUX

```
t
i = 0
i <= 7 ? OUI
( pinCarte [ i ] != pinPresente [ i ] ) ? OUI
return false
```

pinPresente
0,0,0,0,0,0,0,0

- Cas du premier octet VRAI

```

i = 0
i <= 7 ? OUI
( pinCarte [ i ] != pinPresente [ i ] ) ? NON
i++
i <= 7 ? OUI
( pinCarte [ i ] != pinPresente [ i ] ) ? OUI
return false
Δt
```

pinPresente
1,0,0,0,0,0,0,0

La “timing attack” sur le PIN (2/2)

```
for ( i = 0 ; i <= 7; i++)
    if ( pinCarte [ i ] != pinPresente [ i ] )
        return false ;
return true ;
```

 pinCarte
1,3,2,4,250,7,9,134

- Présenter les n valeurs possibles de pinPresente[0] (256 valeurs) (n,0,0,0,0,0,0,0).

- Mesurer la durée d'exécution de la commande T pour les n valeurs.

- Calculer T[n0] le maximum des T
soit T[n0] = max(T[n]); n = 0; ... ; 255
n0 est la solution pour pinCarte[0]

- Itérer sur tous les pinPresente[i]

pinPresente
0,0,0,0,0,0,0,0
1,0,0,0,0,0,0,0
...
255,0,0,0,0,0,0,0

- Nombre d'essais : $8 * 256 = 2048$ (contre 256^8 en force brute)

- On peut même s'arrêter avant n=255 !

Exemples de protection (1/2)

```
alea = random ( 0 , 7 ) ; // alea situe dans l'intervalle [0 ; 7]
for ( i = 0 ; i <= 7; i++) {
    octet = ( alea + i ) mod 8 ;
    if ( pinCarte [ octet ] != pinPresente [ octet ] )
        return false ;
}
return true ;
```

- Nombre d'essais **en moyenne** : $8 * 2048$ (contre 256^8 en force brute)
- Possibilité d'attaques de type MasterMind.

Exemples de protection (2/2)

```
boolean test = true ;
for ( i = 0 ; i <= 7; i++) {
    if ( pinCarte [ i ] != pinPresente [ i ] )
        test = test && false ;
    else
        test = test && true ;
}
return test ;
```

- Équivaut à :

```
boolean test = true ;
for ( i = 0 ; i <= 7; i++) {
    test = ( pinCarte [ i ] == pinPresente [ i ] ) && test ;
}
return test ;
```

- Et pas :

```
boolean test = true ;
for ( i = 0 ; i <= 7; i++) {
    test = test && ( pinCarte [ i ] == pinPresente [ i ] ) ;
}
return test ;
```

Des attaques en temps existent sur d'autres supports (e.g. PC).

Voir par exemple : *Cache-timing attacks on AES* ou *CACHE MISSING FOR FUN AND PROFIT*

Security Rules

And what about security ?

- Write **securely** a function that checks that an array is equal to another with less than 3 trials.

```
boolean verify (byte[] buffer, short ofs, byte len)
{...}
```

- Need a constant `maxTries` initialized at 3,
- A field which memorize the trial number, says `triesLeft`

```

boolean verify (byte[] buffer, short ofs, byte len)
{
    // No comparison if PIN is blocked
    if (triesLeft <= 0)
        return false ;

    // Main comparison
    for(short i=0; i < len; i++)
        if (buffer[ofs+i] != pin[i])
        {
            triesLeft-- ;
            authenticated[0] = false ;
            return false ;
        }

    // Comparison is successful
    triesLeft = maxTries ;
    authenticated[0] = true ;
    return true ;
}

```

Called by a friendly
programmer who has
checked `len` before ;-)

```

boolean verify (byte[] buffer, short ofs, byte len)
{
    // No comparison if PIN is blocked
    if (triesLeft <= 0)
        return false ;

    // First decrements the number of remaining tries
    triesLeft-- ;

    // Main comparison
    boolean equal = true ;
    for(short i=0; i < len; i++)
        equal = (equal && (buffer[ofs+i] != pin[i])) ;// cst time

    if (!equal) {
        // Comparison failed
        authenticated[0] = false ;
        return false ;
    }
    else {
        // Comparison is successful
        triesLeft = maxTries ;
        authenticated[0] = true ;
        return true ;
    }
}

```

```

boolean verify (byte[] buffer, short ofs, byte len)
{
    // No comparison if PIN is blocked
    if (triesLeft <= 0)
        return false ;

    // First decrements the number of remaining tries
    triesLeft-- ;

    // Main comparison
    boolean equal = true ;
    for(short i=0; i < len; i++)
        equal = (equal && (buffer[ofs+i] != pin[i])) ;// cst time

    if (!equal) {
        // Comparison failed
        authenticated[0] = false ;
        return false ;
    }
    else {
        // Comparison is successful
        triesLeft = maxTries ;
        authenticated[0] = true ;
        return true ;
    }
}

```

No!

```

boolean verify (byte[] buffer, short ofs, byte len)
{
    // No comparison if PIN is blocked
    if (triesLeft <= 0)
        return false ;

    // First decrements the number of remaining tries
    triesLeft-- ;

    // Main comparison
    boolean equal = true ;
    for(short i=0; i < len; i++)
        equal = ((buffer[ofs+i] != pin[i]) && equal) ;// cst time

    if (!equal) {
        // Comparison failed
        authenticated[0] = false ;
        return false ;
    }
    else {
        // Comparison is successful
        triesLeft = maxTries ;
        authenticated[0] = true ;
        return true ;
    }
}

```

Yes!

```

public final static short BOOL_TRUE = (short)0x5a5a ;
public final static short BOOL_FALSE = (short)0xa5a5 ;
short equal = BOOL_TRUE ;

...

// Main comparison
for(short i=0; i < len; i++)
    equal = (short)(equal &
        ((buffer[ofs+i] != pin[i])? BOOL_FALSE:  BOOL_TRUE));

if (equal == BOOL_TRUE) {
    // Comparison is successful
    triesLeft = maxTries ;
    authenticated[0] = true ;
    return true ;
}
else {

...

```

```

// First checks the integrity of the variable
if (triesLeft != triesLeftBackup)
    takeCountermeasure() ;

// No comparison if PIN is blocked
if (triesLeft < 0)
    return false ;

```

Correct?

```
// First checks the integrity of the variable
if (triesLeft != triesLeftBackup)
    takeCountermeasure() ;

// No comparison if PIN is blocked
if (triesLeft < 0)
    return false ;
```

- It protects only against a writing between the last and the current one. If the attack is during the evaluation...
- RAM is safer than EEPROM

```
// Transfer in a local and check the integrity of the variable
byte t1 = triesLeft ;

if (t1 != triesLeftBackup)
    takeCountermeasure() ;

// No comparison if PIN is blocked
if (t1 < 0)
    return false ;
```

CDS - Limoges - 2019/2020 - D. Sauveron

149

```
boolean verify (byte[] buffer, short ofs, byte len) {
    byte t1 = triesLeft ;
    if (t1 != (short)(~triesLeftBackup)) takeCountermeasure() ;
    if (t1 < 0) return false;
    JCSystem.beginTransaction() ;
        triesLeft = --t1 ;
        triesLeftBackup++ ;
    JCSystem.commitTransaction() ;
    if (triesLeft != (short)(~triesLeftBackup)) takeCountermeasure() ;
    short equal = BOOL_TRUE ;
    for(short i=0; i < len; i++)
        equal = (short)(equal &
            ((buffer[ofs+i]!=pin[i])?BOOL_FALSE:BOOL_TRUE));

    if (equal == BOOL_TRUE) {
        JCSystem.beginTransaction() ;
            triesLeft = maxTries ;
            triesLeftBackup = (byte)(~maxTries) ;
        JCSystem.commitTransaction() ;
        // Verifies the new value
        if (triesLeft!=(short)(~triesLeftBackup))
            takeCountermeasure() ;
        authenticated[0] = true ;
        return true ;
    } else {
        authenticated[0] = false ;
        return false ;
    }
}
```

CDS - Limoges - 2019/2020 - D. Sauveron

150

```

boolean verify(byte[] buffer, short ofs, byte len)
{
    // Initializes the step counter
    short stepCounter = INITIAL_COUNTER ;

    // First checks the integrity of the variable
    byte tl = triesLeft ;
    stepCounter++ ;
    if (tl != (short)(~triesLeftBackup)) takeCountermeasure() ;
    stepCounter++ ;
    // No comparison if PIN is blocked
    if (tl < 0) return false ;
    stepCounter++ ;
    // First decrements the number of remaining tries
    JCSysSystem.beginTransaction() ;
        triesLeft = --tl ;
        stepCounter++ ;
        triesLeftBackup++ ;
    JCSysSystem.commitTransaction() ;
    stepCounter++ ;
    // Verifies the new value
    if (triesLeft != (short)(~triesLeftBackup)) takeCountermeasure() ;

```

```

    stepCounter++ ;
    short equal = BOOL_TRUE ; // Main comparison
    stepCounter++ ;
    for(short i=0;i<len;i++)
        equal = (short)(equal &
            ((buffer[ofs+i]!=pin[i])?BOOL_FALSE:BOOL_TRUE));
    stepCounter++ ;
    if (equal == BOOL_TRUE) { // Comparison is successful
        //Reset the tries to the max
        stepCounter++ ;
        JCSysSystem.beginTransaction() ;
            triesLeft = maxTries ;
            triesLeftBackup = (byte)(~maxTries);
        JCSysSystem.commitTransaction() ;
        stepCounter++ ;
        if (triesLeft!=(short)(~triesLeftBackup))
            takeCountermeasure() ;
        stepCounter++ ;
        authenticated[0] = true ;
        if (stepCounter == (short)(INITIAL_VALUE+11) )
            return true ;
    } else { // Comparison failed
        stepCounter++ ;
        authenticated[0] = false ;
        if (stepCounter == (short)(INITIAL_VALUE+9))
            return false ;
    }
    takeCountermeasure() ; // Should have returned at this point
}

```

Other well known rules ...

```
if (pin.isValidated() == true)
{
    // sensitive operations
}
else
{
    ISOException.throwIt((short) (0x6300+pin.getTriesRemaining()));
}
```

Other well known rules ...

```
if (pin.isValidated() == true)
{
    if (pin.isValidated() == false)
    {
        ISOException.throwIt((short) (0x6300+pin.getTriesRemaining()));
    }
    else
    {
        // sensitive operations
    }
}
else
{
    ISOException.throwIt((short) (0x6300+pin.getTriesRemaining()));
}
```

- Références :

<http://javacard.vetilles.com/2008/05/15/jc101-12c-defending-against-attacks/>

Ronald De Keulenaer, Jonas Maebe, Koen De Bosschere, Bjorn De Sutter. Link-time smart card code hardening. International Journal of Information Security, 2015

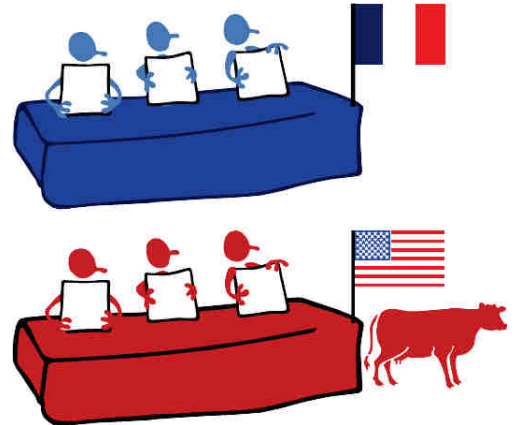
La consommation de courant

La consommation de courant

- Elle est surtout utilisée dans le domaine de la cryptographie.
- Il existe différentes attaques :
 - la SPA (Simple Power Analysis)
 - la DPA (Differential Power Analysis)
 - la HODPA (High Order Differential Power Analysis)

IMPORTATION DU BOEUF ?

- Seattle, 1999.
- Les représentants Français et Américains négocient sous quelles conditions le bœuf pourrait être importé en France.

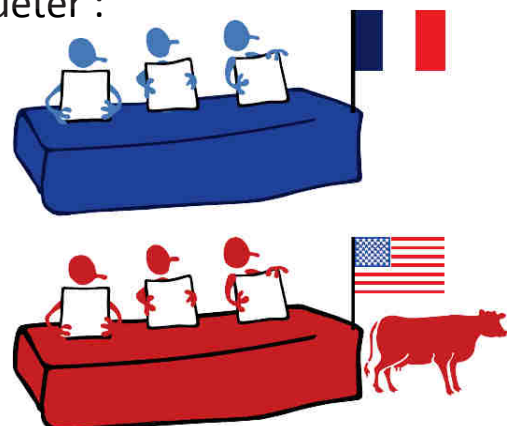


CDS - Limoges - 2019/2020 - D. Sauveron

157

IMPORTATION DU BOEUF ?

- Seattle, 1999.
- Les représentants Français et Américains négocient sous quelles conditions le bœuf pourrait être importé en France.
- «The Sun » envoie un journaliste pour enquêter :

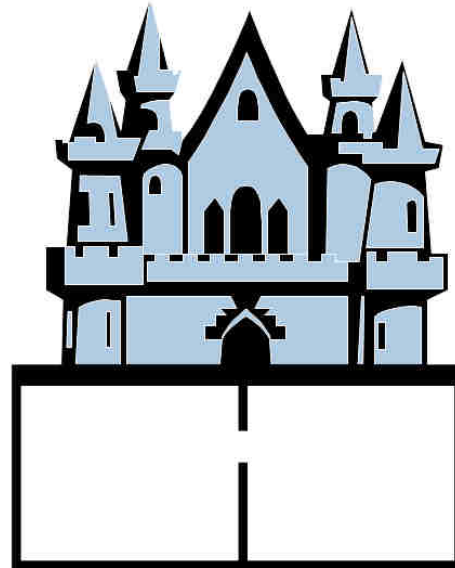


CDS - Limoges - 2019/2020 - D. Sauveron

158

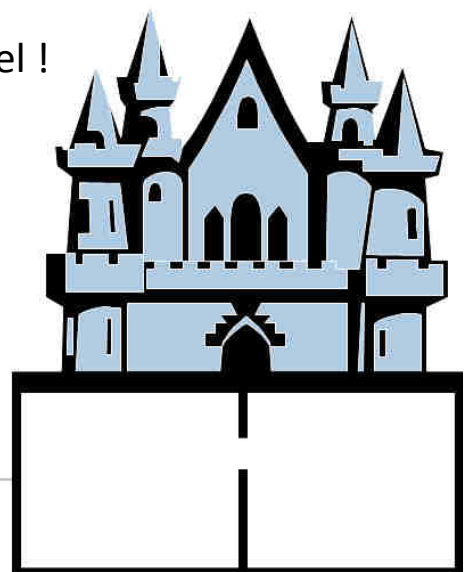
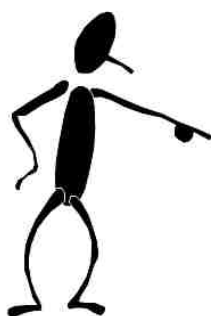
IMPORTATION DU BOEUF ?

- Mais il y a un problème technique : les négociations se déroulent dans un hôtel aux vitres opaques



Attaque sur la consommation

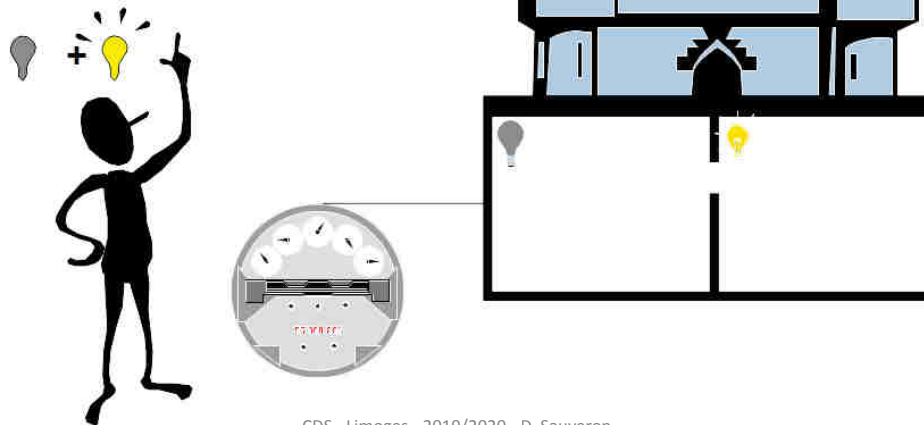
- Idée : Regarder le compteur électrique de l'hôtel !



Attaque sur la consommation

- Le disque tourne lentement:

DEAL CONCLUDED



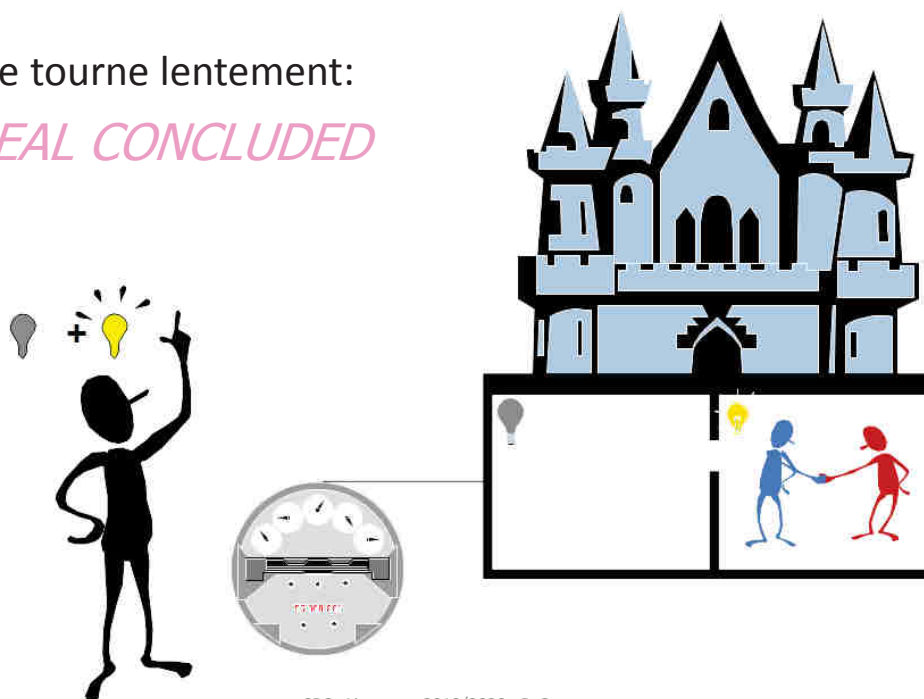
CDS - Limoges - 2019/2020 - D. Sauveron

161

Attaque sur la consommation

- Le disque tourne lentement:

DEAL CONCLUDED

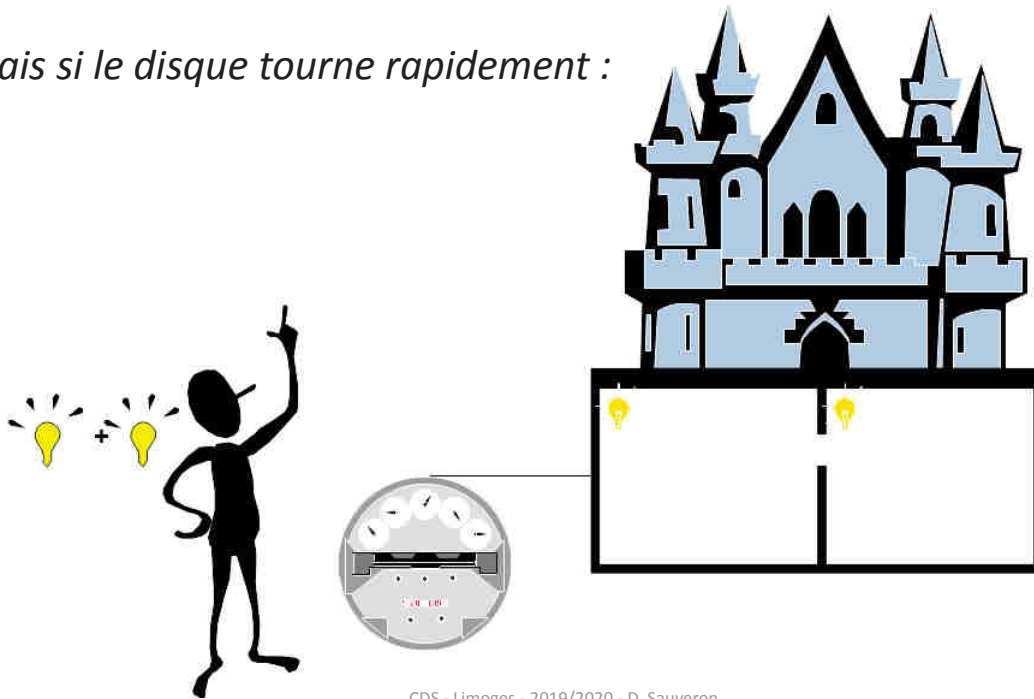


CDS - Limoges - 2019/2020 - D. Sauveron

162

Attaque sur la consommation

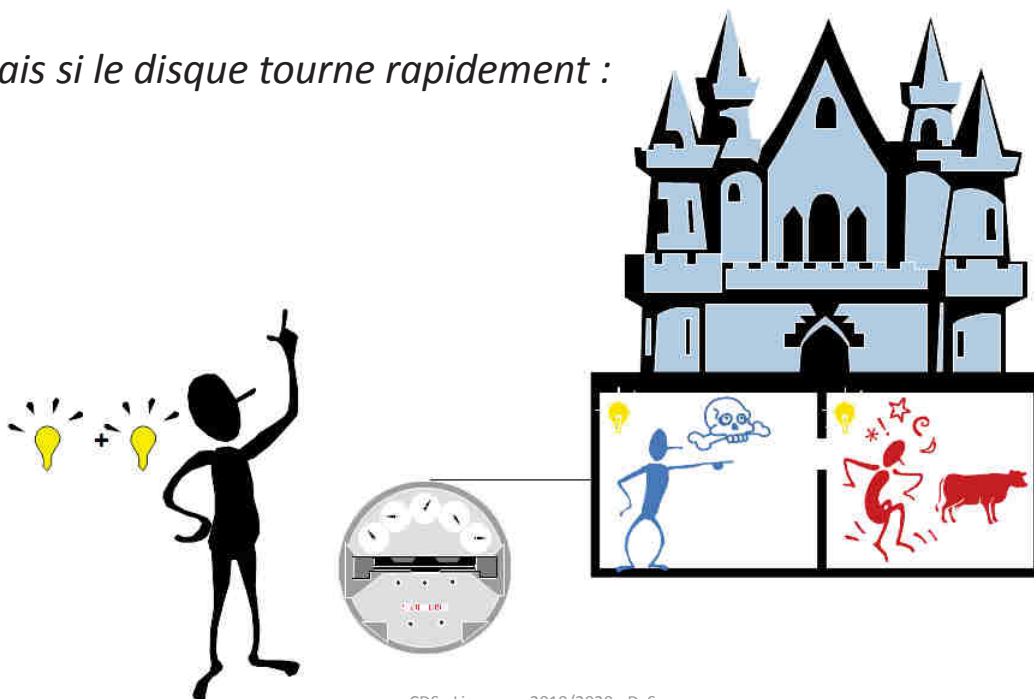
- *Mais si le disque tourne rapidement :*



163

Attaque sur la consommation

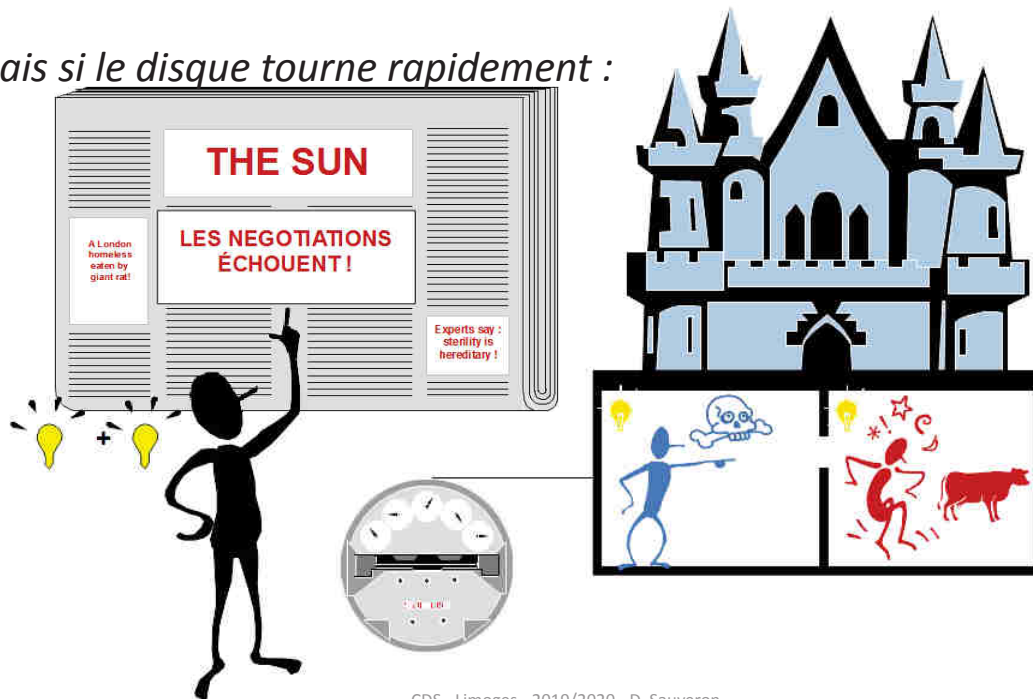
- *Mais si le disque tourne rapidement :*



164

Attaque sur la consommation

- *Mais si le disque tourne rapidement :*

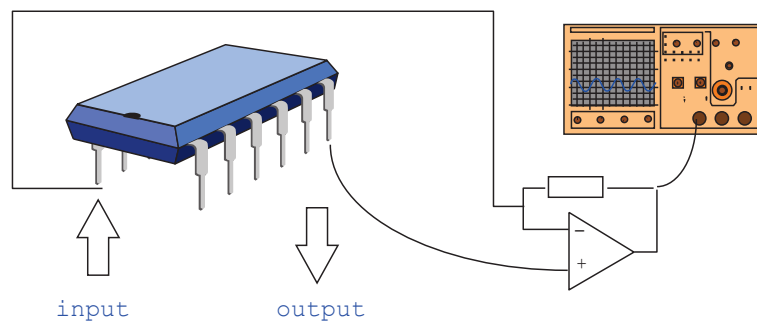


CDS - Limoges - 2019/2020 - D. Sauveron

165

Conclusion

- The card's current consumption **may reveal secret information.**

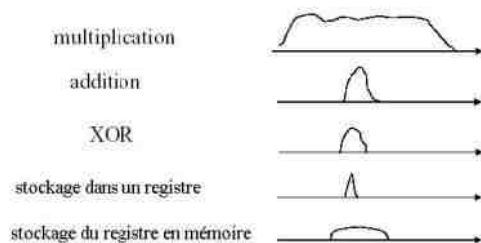


CDS - Limoges - 2019/2020 - D. Sauveron

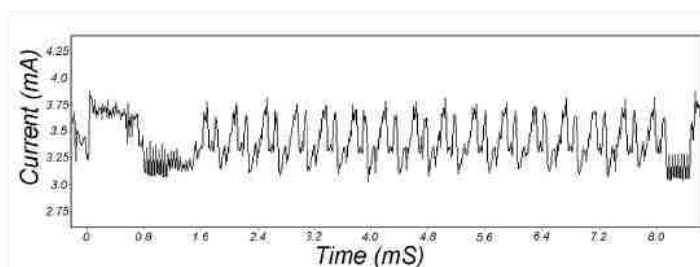
166

La SPA (Simple Power Analysis)

- Principe : Des instructions différentes ont une trace différente.



Consommation en courant d'un DES. On peut voir la permutation initiale, suivie des 16 tours.



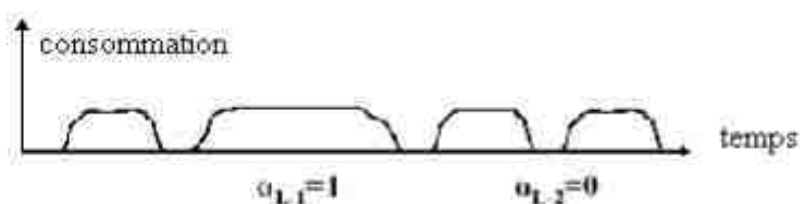
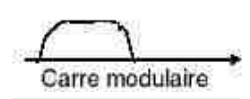
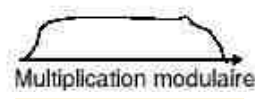
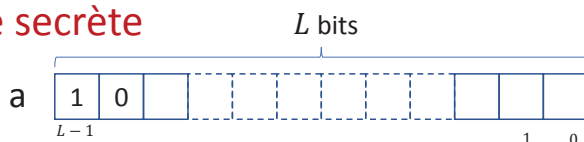
CDS - Limoges - 2019/2020 - D. Sauveron

167

SPA sur la signature RSA (théorie)

- Signature RSA : $y^a \mod n$
- y est le message à signer,
- n est public,
- a , l'exposant peut être considéré comme la clé secrète

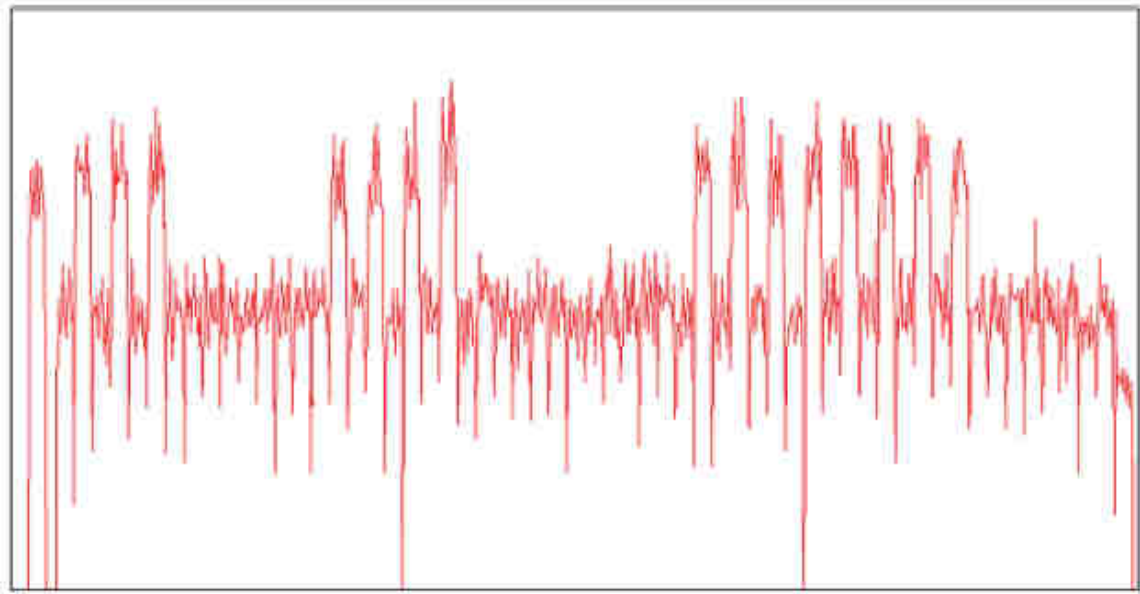
```
s = 1 ;
for ( i = L - 1 ; i >= 0 ; i-- )
{
    s = s*s mod n ;
    if ( a [ i ] == 1 )
        s = s*y mod n ;
}
```



CDS - Limoges - 2019/2020 - D. Sauveron

168

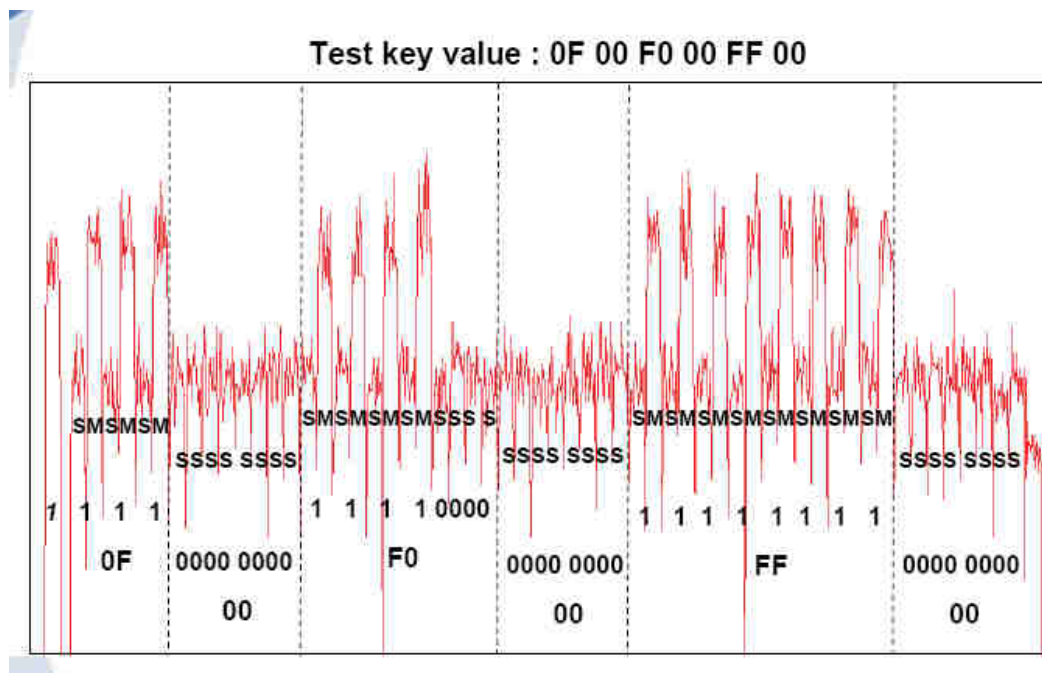
Un exemple réel



CDS - Limoges - 2019/2020 - D. Sauveron

169

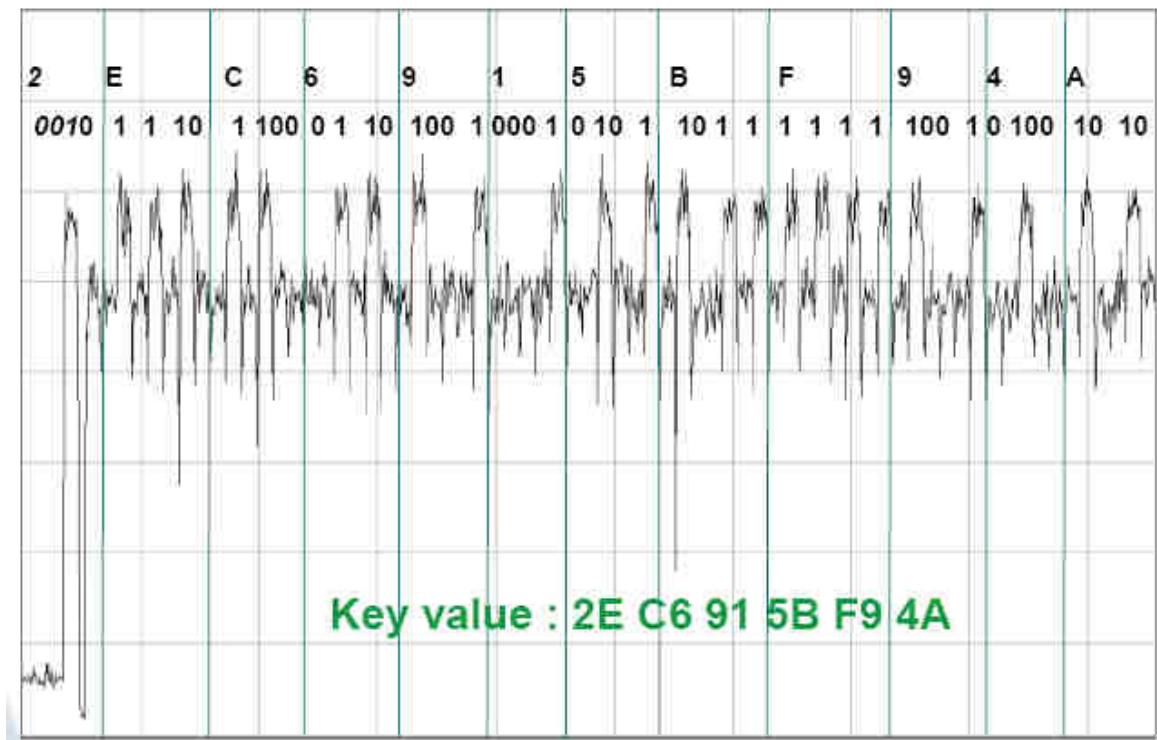
Un exemple réel



CDS - Limoges - 2019/2020 - D. Sauveron

170

Un exemple réel

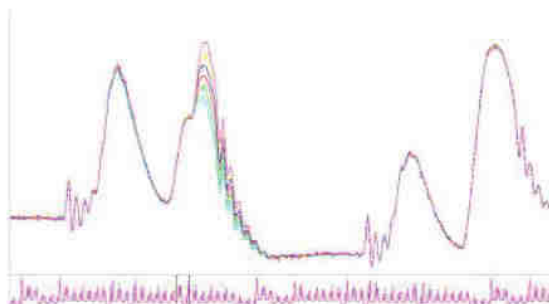


171

La DPA (Differential Power Analysis)

- Principe : La consommation dépend des opérations effectuées (les instructions) mais aussi des opérandes.
- Elle utilise des fonctions statistiques adaptées à l'algorithme visé qui font ressortir des corrélations entre un bit intermédiaire a (ne dépendant que d'un fragment Kr de r bits de la clé et du message d'entrée M) et la consommation de courant.

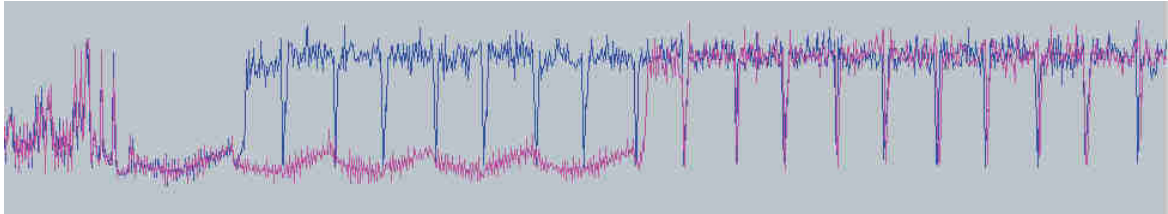
- Based on SPA
Adding the power of statistics to separate signal from noise



172

Concrete attack on concrete keys

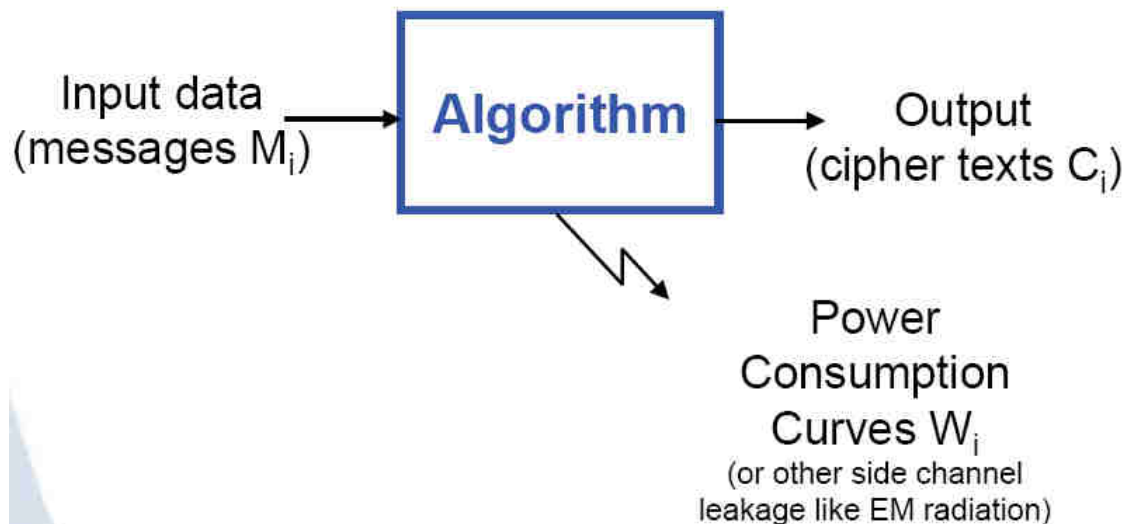
- Comparing:
 - decryption with key = 0000 1111 ...
 - decryption with key = 1111 1111 ...



token designed and manufactured in 1998...

DPA Hypothesis

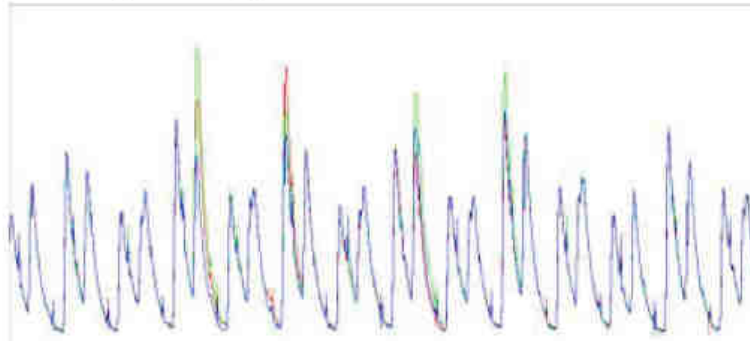
Play the algorithm N times
($100 < N < 100000$)



Acquisition procedure

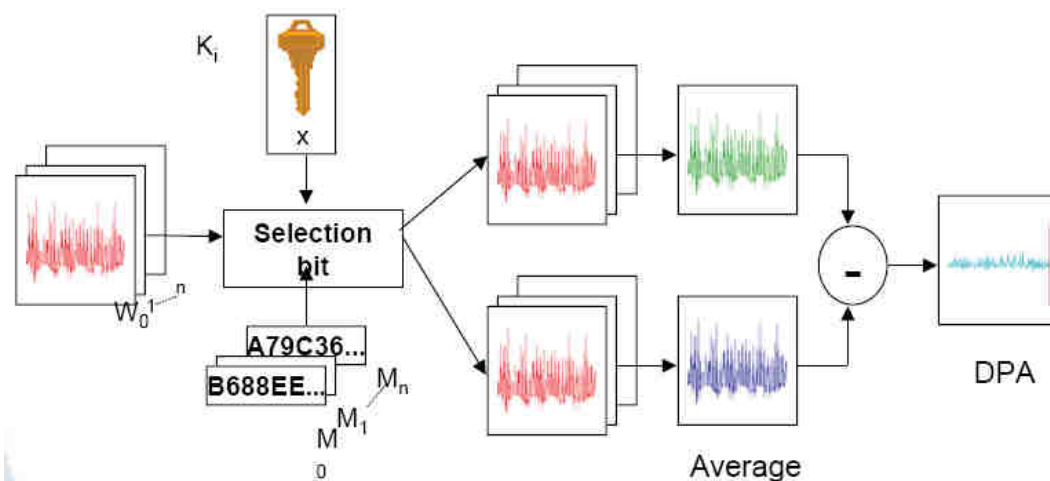
- After data collection, what is available ?
 - N plain and/or cipher random texts

00	B688EE57BB63E03E
01	185D04D77509F36F
02	C031A0392DC881E6 ...
 - N corresponding power consumption waveforms

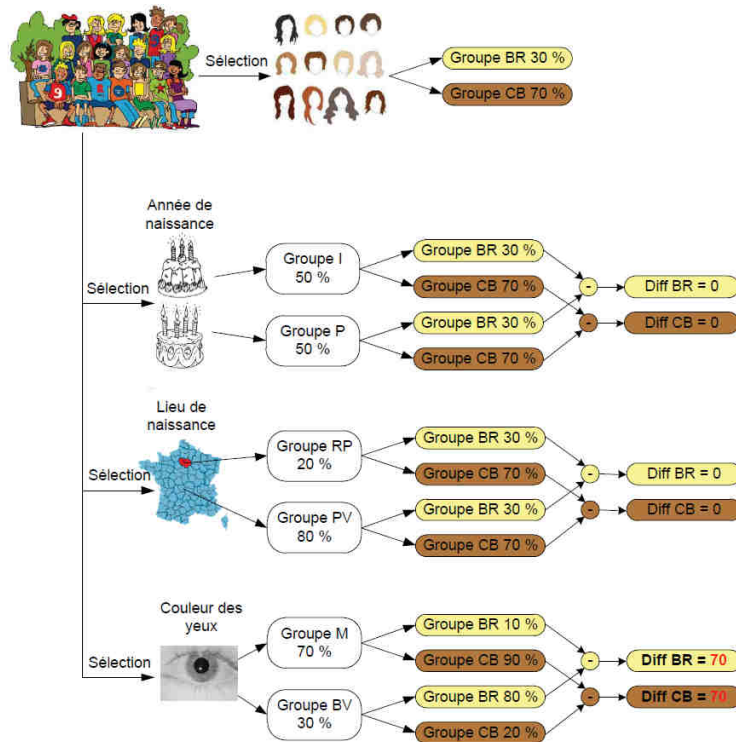


175

Hypothèse et test



Explication didactique



Notion de
corrélation

177

La HODPA (High Order DPA)

- La HODPA ou DPA d'ordre n est aussi basée sur une étude statistique de la consommation de courant de la carte.
- Différence : elle utilise des corrélations entre la consommation de courant et n variables intermédiaires ne dépendant que d'un fragment de la clé et du message d'entrée.
- Elle est beaucoup plus difficile à réaliser, mais beaucoup plus puissante.

Exercice

- Une pièce sans fenêtre dispose d'une lumière (une ampoule).
- Trois interrupteurs () sont placés à l'extérieur de cette pièce.
- Sachant que les trois interrupteurs sont en position OFF et **qu'on a le droit d'entrer** dans la pièce **qu'une seule fois**, comment savoir quel est l'interrupteur qui allume cette lumière ?
- *Attention : On ne peut pas laisser la porte ouverte, on ne peut pas se faire aider de quelqu'un et la porte ne contient pas de vitre et ne comporte pas de trou de serrure. Il n'y a pas de jour autour de la porte. Les câbles partant des interrupteurs sont blindés et ne sont pas accessibles physiquement. En gros la pièce ne laisse fuir aucune information d'aucun nature que ce soit vers l'extérieur.*

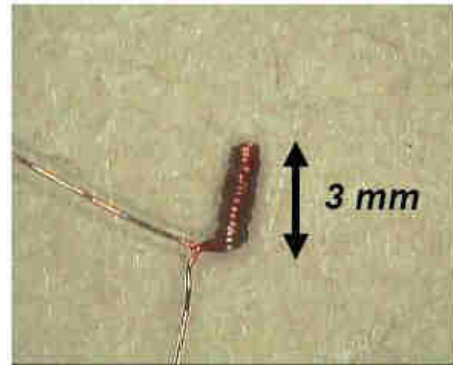
COUNTER-COUNTER MEASURE

- Against adding noise to the power consumption signal.
- Capture electromagnetic radiation at various chip locations!
- Equipment:



Les émissions EM

- **Principe** : Les courants qui circulent dans la puce induisent des champs électromagnétiques qui sont susceptibles de donner le même type d'information que le courant.
- **Différence** : L'information est plus locale. On peut déplacer la micro-sonde électromagnétique au dessus de la zone qui nous donnera le plus d'informations (exemple : co-processeur cryptographique).
- **Avantage** : Insensible aux contre-mesures physiques tels que l'ajout de bruit en sortie ou le lissage de la consommation globale de courant.
- **Inconvénient** : La reproductibilité des mesures est difficile.



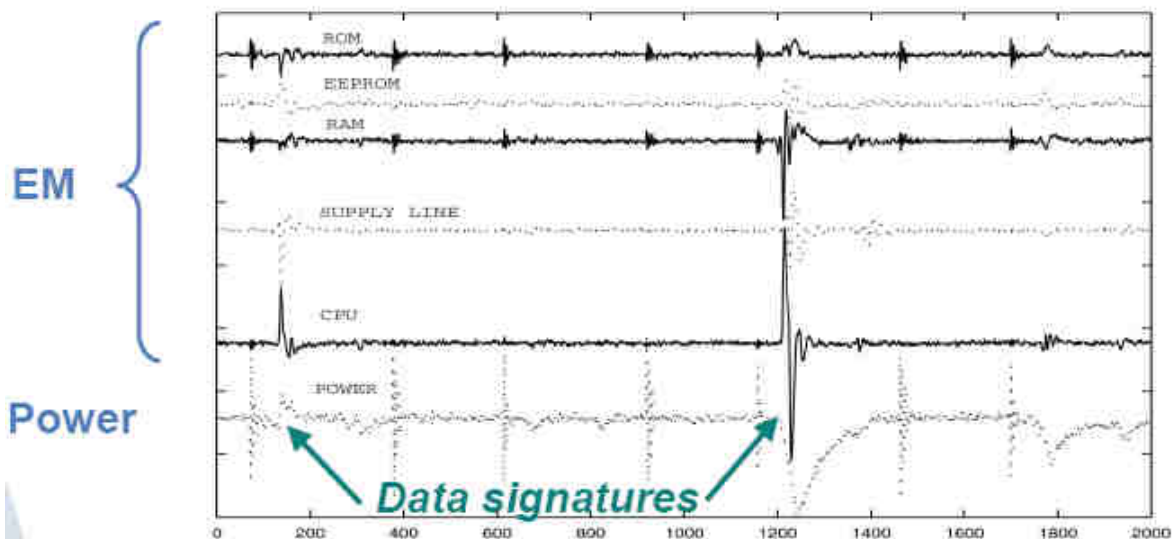
CDS - Limoges - 2019/2020 - D. Sauveron

181

Différence EM/Power

- **EM signals versus XY probe position**

Differential traces between (00h $\oplus$ 00h) and (FFh $\oplus$ 00h) picked up at different locations

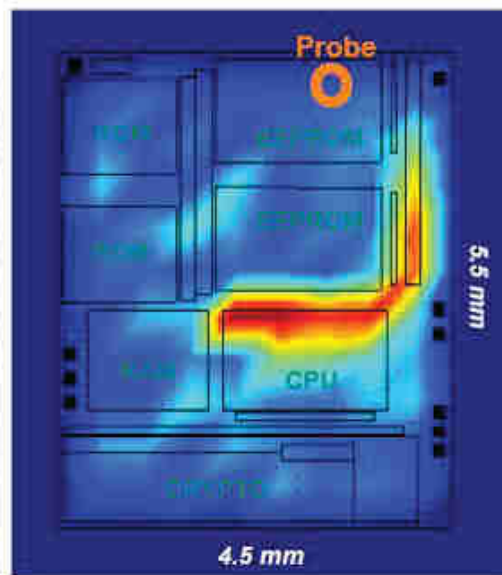
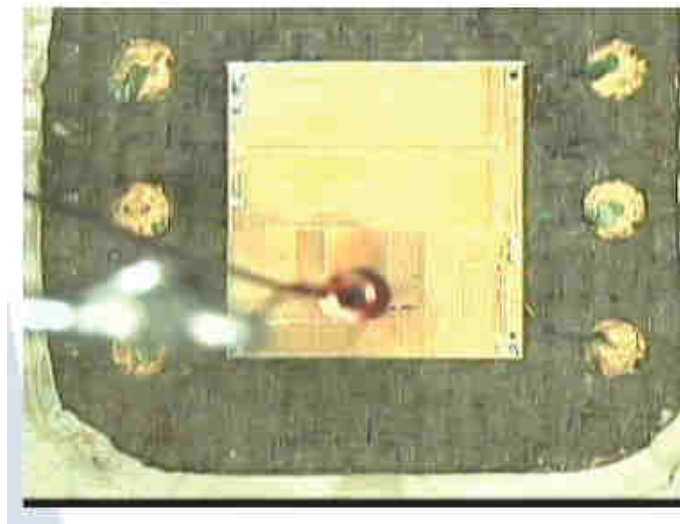


CDS - Limoges - 2019/2020 - D. Sauveron

182

Le dispositif de cartographie EM

- Horizontal cartography (XY plane)
 - to pinpoint instruction related areas
 - better if automated

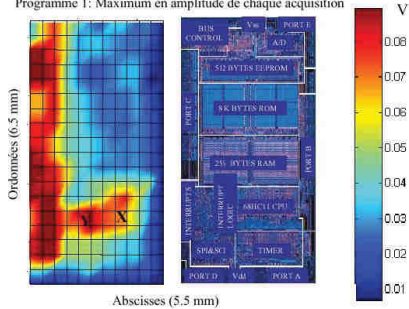


CDS - Limoges - 2019/2020 - D. Sauveron

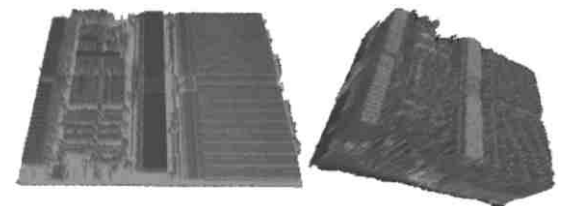
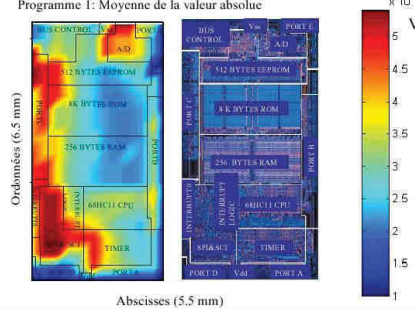
183

Cartographie pour 2 algo / Vue 3D

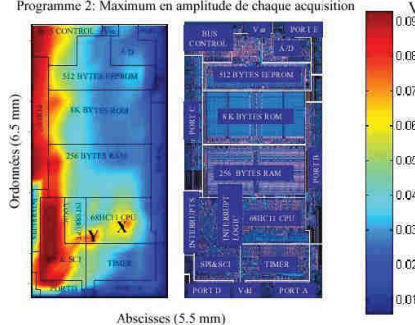
Programme 1: Maximum en amplitude de chaque acquisition



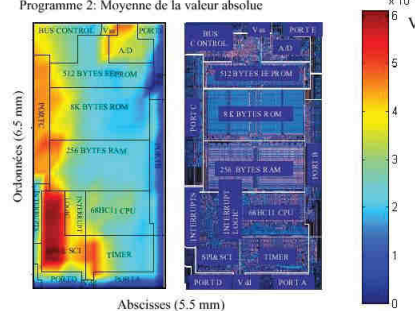
Programme 1: Moyenne de la valeur absolue



Programme 2: Maximum en amplitude de chaque acquisition



Programme 2: Moyenne de la valeur absolue



CDS - Limoges - 2019/2020 - D. Sauveron

184

Attaques EM

- la SEMA (Simple EM Analysis)
- la DEMA (Differential EM Analysis)
 - => nécessite moins d'acquisitions

La rétro-conception logicielle

- méthode du « dictionnaire » :

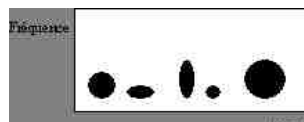
- Une instruction :



- Une autre instruction :



- La séquence des deux :



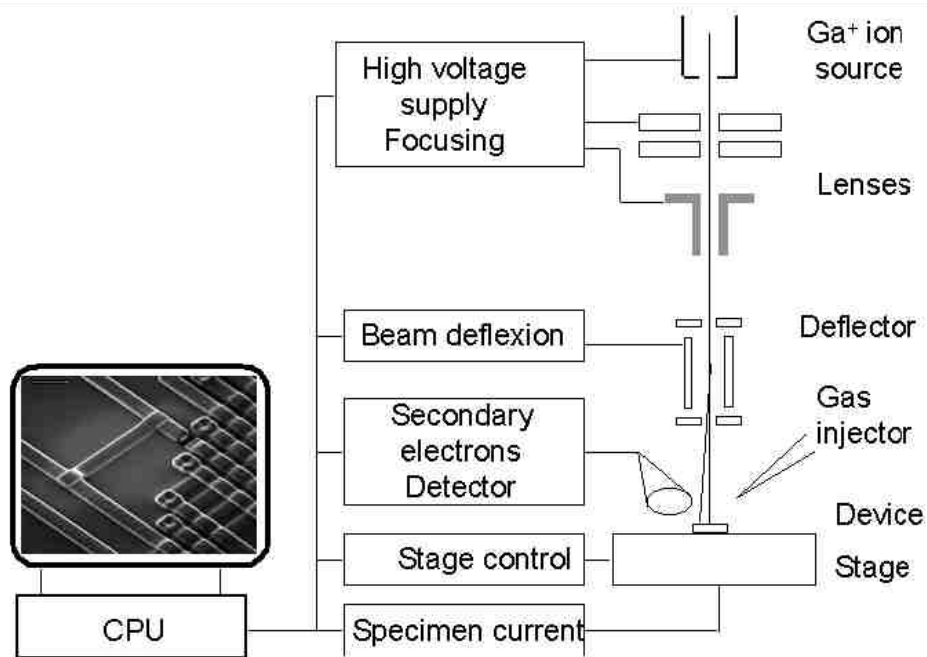
- Utilisation possible de la consommation en courant ou des émissions EM.
- Statut : recherches en cours mais de beaux résultats à WISTP 2007,

Quelques attaques invasives

Des attaques invasives

- micro-probing
- modification de circuit (FIB)
- rétro-conception du circuit

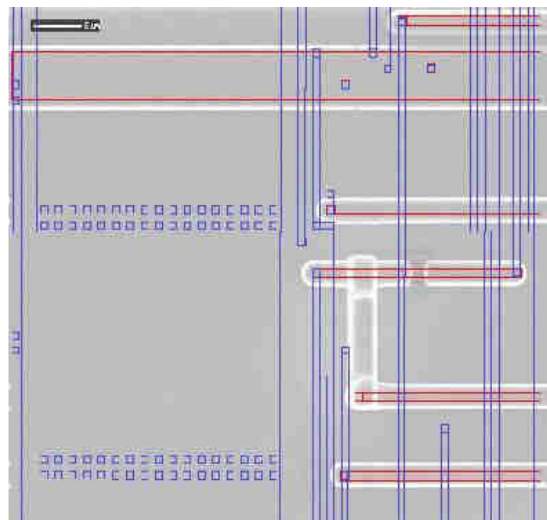
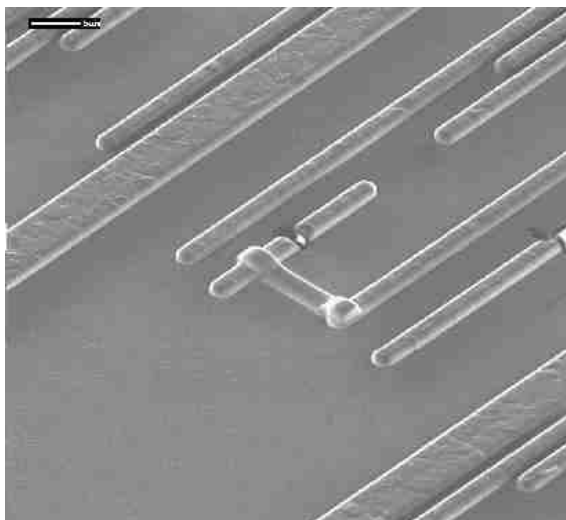
Le FIB (Focused Ion Beam)



CDS - Limoges - 2019/2020 - D. Sauveron

189

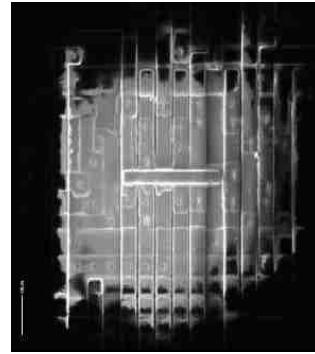
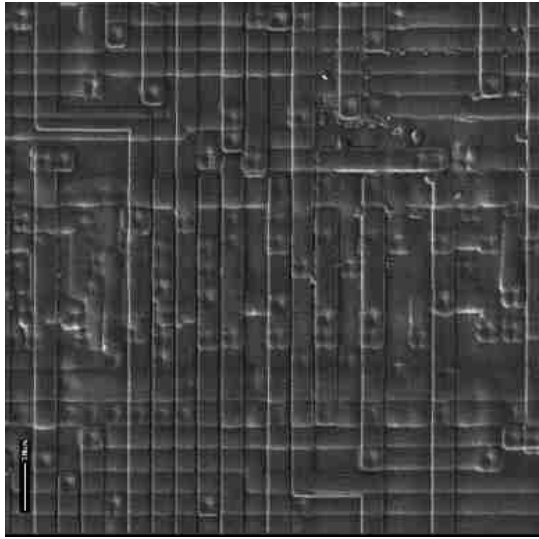
Modifications de circuits (1/2)



CDS - Limoges - 2019/2020 - D. Sauveron

190

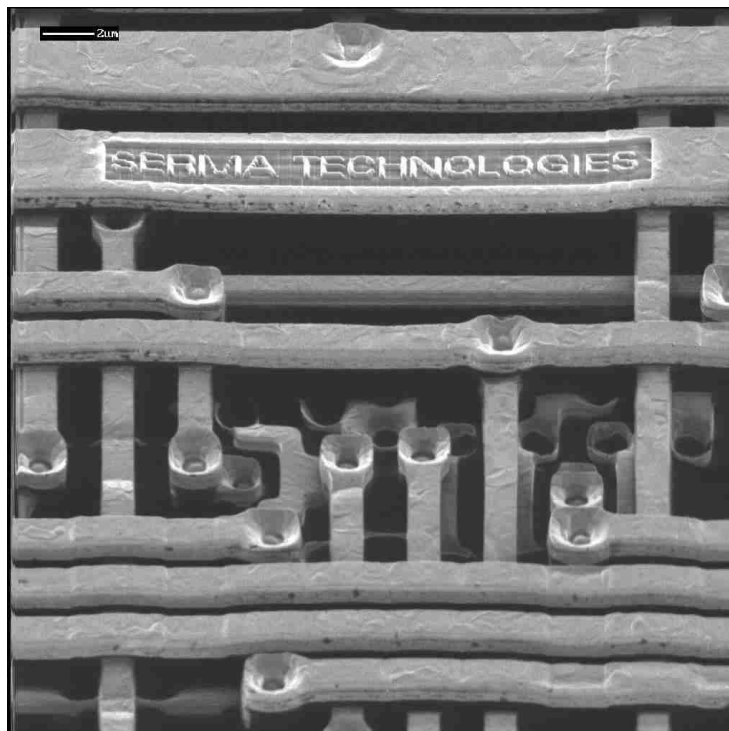
Chip Re-Wiring / Addition of a Track



CDS - Limoges - 2019/2020 - D. Sauveron

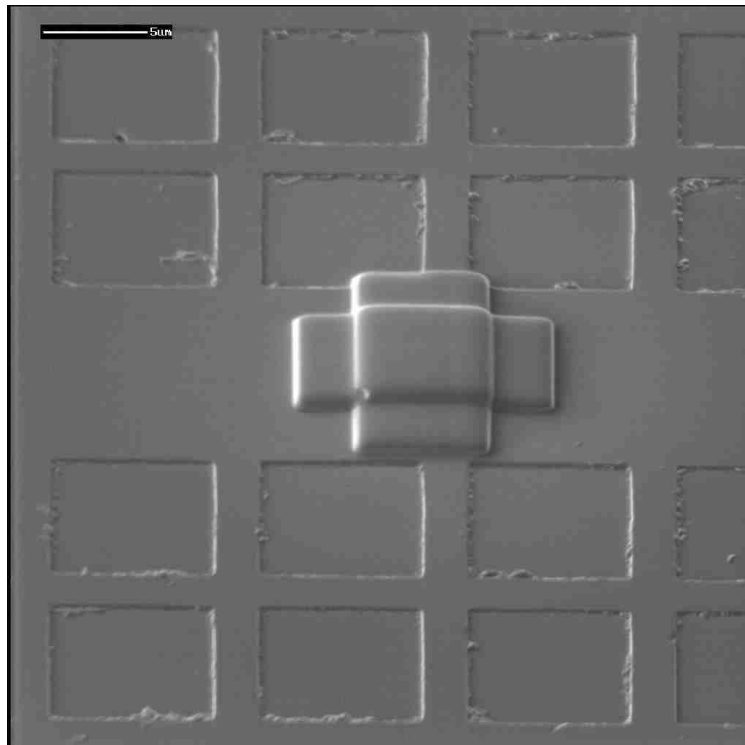
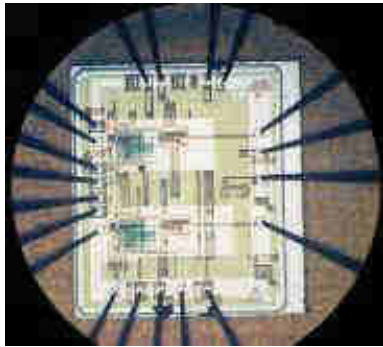
191

Modifications de circuits (2/2)



192

Micro-probing



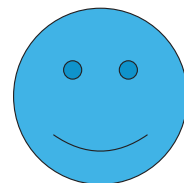
CDS - Limoges - 2019/2020 - D. Sauveron

193

Coût

Très cher !

- If you have more money or if you are a student.



CDS - Limoges - 2019/2020 - D. Sauveron

194

Cotation d'une attaque

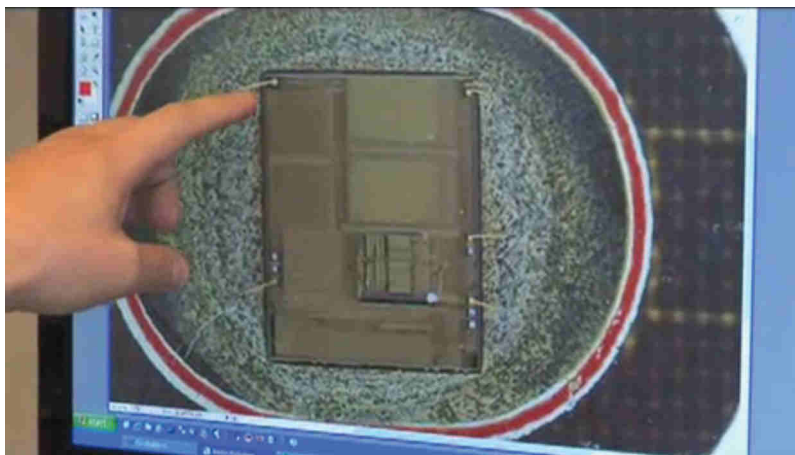
- Niveau d'expertise requise
 - Expert, ..., homme de la rue
- Connaissance du produit
 - Information publique, diffusion restreinte, sensible, critique
- Accès au produit
 - Nombre d'échantillon nécessaire : <10 , <100, <1000, non praticable
- Temps d'identification
 - <1 heure, 1 jour, 1 semaine, 1 mois, ...
- Temps d'exploitation
 - <1 heure, 1 jour, 1 semaine, 1 mois, ...
- Équipement
 - Aucun, standard, spécialisé, ...
- ...

=> La sécurité et le business s'opposent. Il faut donc trouver un juste compromis !

CDS - Limoges - 2019/2020 - D. Sauveron

195

Le piratage facile en vidéo



How to Reverse-Engineer a Satellite TV Smart Card.mp4

CDS - Limoges - 2019/2020 - D. Sauveron

196

A finaliser !!!!!!!!!!!!!!!!!!!!!!!

La norme FIPS 140

- FIPS 140-2, June 2001, Security requirements for Cryptographic Modules :

- <http://www.csrc.nist.gov/publications/fips/>

- Recommandations pour :

- Physique,
 - OS,
 - Key Management
 - EM
 - Self Tests
 - Design Assurance

Sécurité physique dans FIPS

	General Requirements for all Embodiments	Single-Chip Cryptographic Modules	Multiple-Chip Embedded Cryptographic Modules	Multiple-Chip Standalone Cryptographic Modules
Security Level 1	Production-grade components (with standard passivation).	No additional requirements.	If applicable, production-grade enclosure or removable cover.	Production-grade enclosure.
Security Level 2	Evidence of tampering (e.g., cover, enclosure, or seal).	Opaque tamper-evident coating on chip or enclosure.	Opaque tamper-evident encapsulating material or enclosure with tamper-evident seals or pick-resistant locks for doors or removable covers.	Opaque enclosure with tamper-evident seals or pick-resistant locks for doors or removable covers.
Security Level 3	Automatic zeroization when accessing the maintenance access interface. Tamper response and zeroization circuitry. Protected vents.	Hard opaque tamper-evident coating on chip or strong removal-resistant and penetration resistant enclosure.	Hard opaque potting material encapsulation of multiple chip circuitry embodiment or applicable Multiple-Chip Standalone Security Level 3 requirements.	Hard opaque potting material encapsulation of multiple chip circuitry embodiment or strong enclosure with removal/penetration attempts causing serious damage.
Security Level 4	EFP or EFT for temperature and voltage.	Hard opaque removal-resistant coating on chip.	Tamper detection envelope with tamper response and zeroization circuitry.	Tamper detection/response envelope with tamper response and zeroization circuitry.

Table 2: Summary of physical security requirements

199

A finir (un jour)

- UML
- Dev (cycle de vie)
- Méthodes formelles
 - Conception formelle de certains partie de l'OS (voire de l'OS complet)
 - Preuve formelle sur des aspects sécuritaire
 - Génération de jeux de tests associés aux spécifications
- Test
- Qualité du code
- Etc.

Questions ?

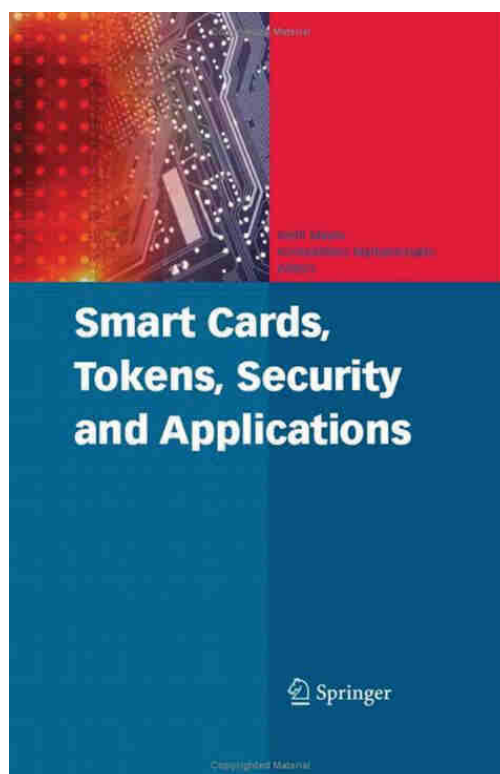


Bibliographie

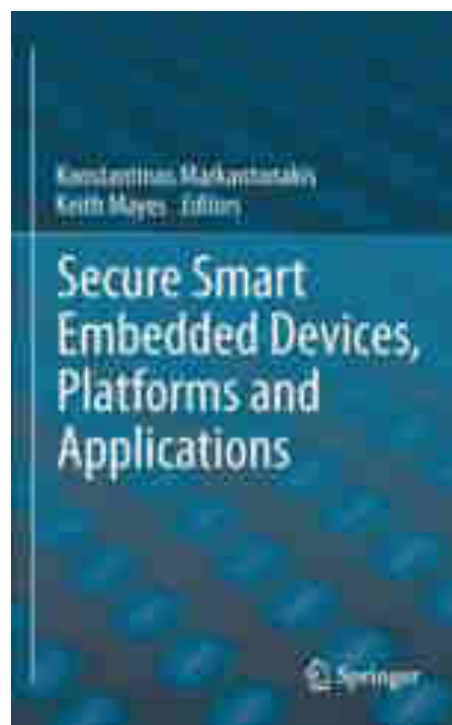


203

Smart Cards, Tokens, Security and Applications
 Mayes, Keith; Markantonakis, Konstantinos (Eds.)
 2008, XXXVIII, 392 p. 128 illus., Hardcover
 ISBN: 978-0-387-72197-2



Secure Smart Embedded Devices, Platforms and Applications
 Konstantinos Markantonakis, Keith Mayes
 2014, XLI, 568 p. 135 illus., Hardcover
 ISBN: 978-1-4614-7914-7



Les cartes à puce
 BOUZEFRA Samia, PARADINAS Pierre
 Ouvrage 326 p. 15.6x23.4 cm Relié

