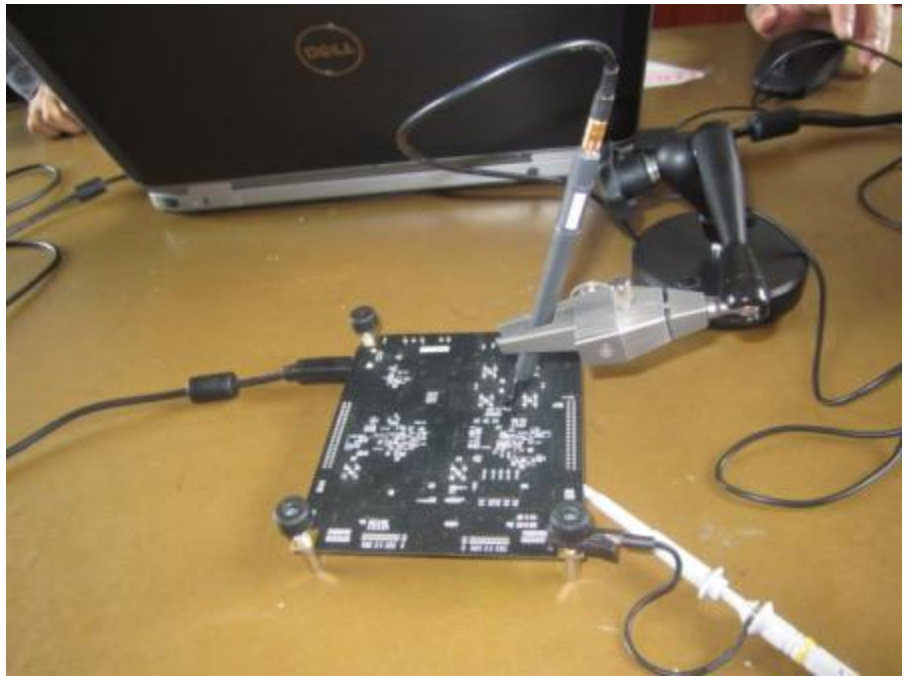
SSTIC 2011

Jour 1 matin

- Thoughts on Client Systems Security, J. Rutkowska
- BitLocker, A. Bordes
- Silverlight ou comment surfer à travers .NET, T. Caplin
- XSSF : démontrer le danger des XSS, L. Courgnaud

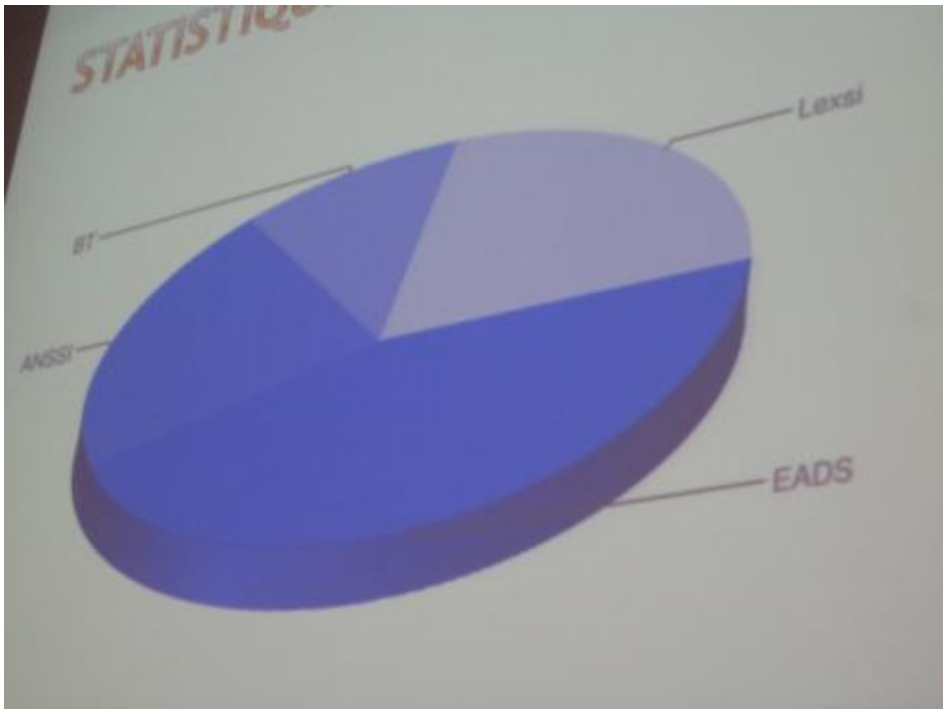
Jour 1 après-midi

- **Rainbow Tables probabilistes, Alain Schneider**
- Memory Eye Yoann Guillot
- Attaque d'implémentations cryptographiques par canaux cachés, Philippe Nguyen
- Sécurité du système Android, Nicolas RUFF



Jour 2 matin

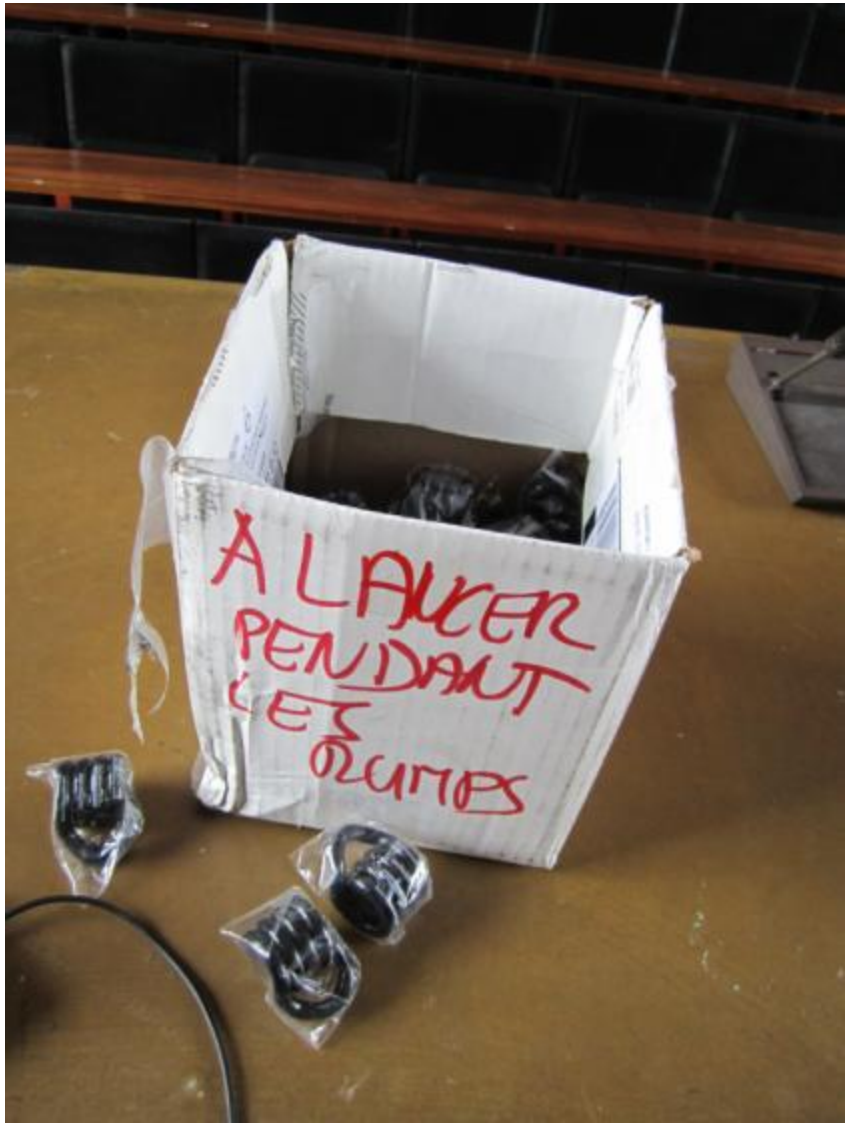
- Attaques DMA peer-to-peer et contremesures, F. Lone Sang, et. Al
- Sticky fingers & KBC Custom Shop, A. Gazet
- Virtualisation depuis le boot, S. Duverger
- Challenge
- **Attacking and Fixing PKCS#11 Tokens, G. Steel**



Jour 2 après-midi

- Peut-on éteindre l'Internet ?
  S. Bortzmeyer
- Architecture DNS sécurisée,
  G. Valadon et al.

- Rump session



Jour 3 matin

- RRABBIDS, R. Ludinard
- Usages offensifs de XSLT
  N. Gregoire
- Faille ou défaut de sécurité
  E. Barbry
- Typologie des attaques […]
  J. Zimmermann
- Stockage-en-ligne […]
  L. Montalvo et al.

Jour 3 après-midi

- Un framework de fuzzing pour cartes à puce EMV, J. Lancia
- Sécurité ? H. Schauer

Rainbow Tables probabilistes
Alain Schneider, LEXSI

- Objectif : accélérer le cassage de mots de passe
- Moyen : combiner deux méthodes connues
  - Les rainbow tables
  - Les automates de Markov
- Caractéristique : ça n'est pas simple !
- Résultat
  - Vitesse : gain de plusieurs ordres de grandeur
  - Couverture : 0Go/50%-12Go/64%-8,5 Go/83%

Photo LinkedIn

Cassage de mots de passe

- On dispose d'une base de hash
- On recherche les mots de passe initiaux

- Exemple
  - 2a33c23a7ab740c82c16ab26d8d59ad59376da9a
  - P@ssW0rd
- Hypothèse
  - Mots de passe non salés

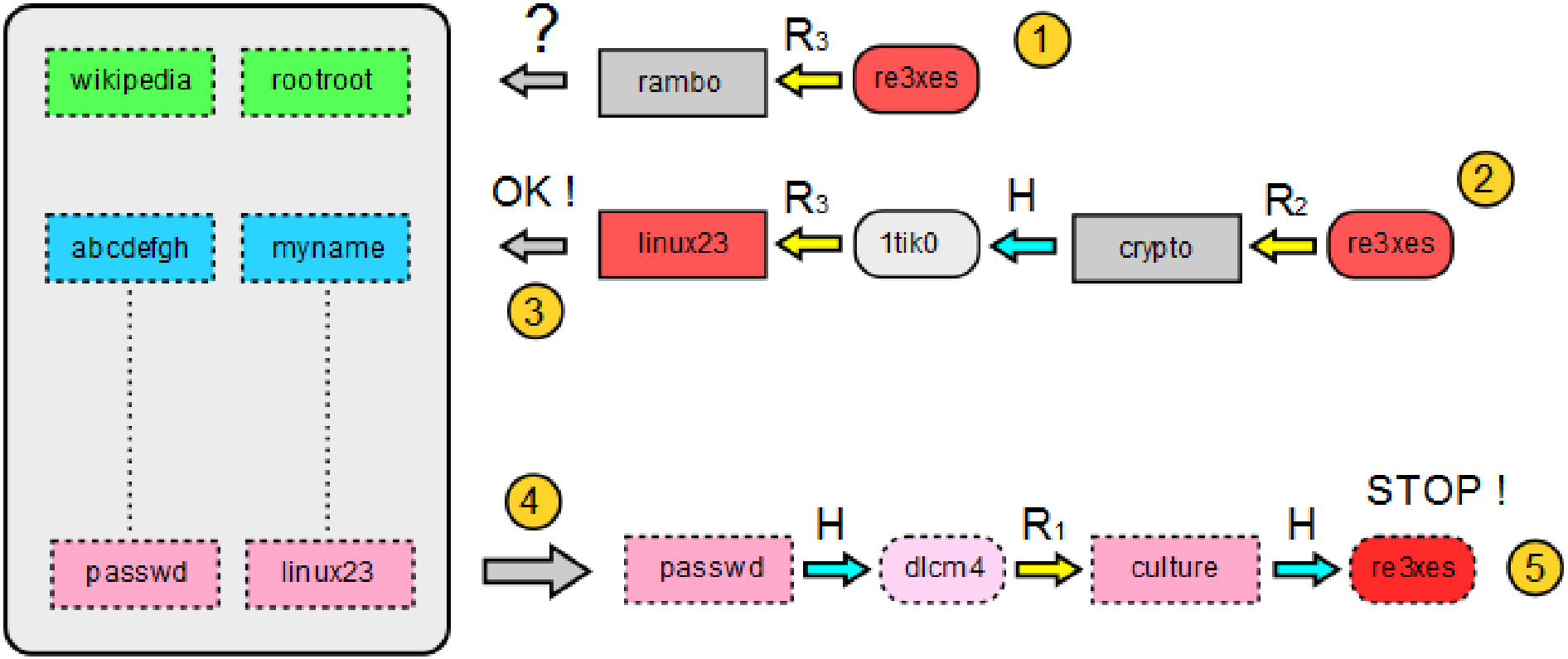
Force brute

- On génère des mots de passe, on calcule leurs hash, on compare avec les hash de la base
- En cas d'égalité on a trouvé un mot de passe
  - ou un mot ayant le même hash

- Tout essayer : trop long
  - a,b,…,aa,ab,…,aaa,aab,…,aba,abb,…,**zz…zzzz…zZZZ**
- Tout pré-calculer : trop gros

Compromis temps-mémoire

- On pré-calcule des chaines de hash et de mots de passe
- On ne stocke que les extrémités des chaines
- Il faut
  - La même fonction de hachage que celle des mots de passe
  - Des fonctions de réduction qui projette des hash vers les mots de passe
  - Du temps de calcul



Source Wikipedia

Probabilités

- Les mots de passe sont rarement aléatoires
- Dans une base de mots de passe les lettres ne se succèdent pas avec la même probabilité
- Exemple
  - P@ssW0rd, sstic2011, kissme!, n0s**s**ecure!, sossos!
  - ‘s’ est cinq fois suivi de ‘s’
  - ‘e’ est deux fois suivi de ‘!’
  - ‘e’ n’est jamais suivi de ‘a’
  - …

Automate de Markov

- On peut calculer ces probabilités pour une base donnée (comme RockYou)
- Et régénérer beaucoup de mots en suivant les mêmes règles de probabilité
- **Difficulté : numéroter !**
  - Pour pouvoir fabriquer les fonctions de réduction
  - Numérotation par parcours en largeur (et profondeur bornée)

Automate de Markov

- La numérotation prend du temps
  - Pas traitable sur CPU / Traitable sur GPU
  - Recherche dichotomique plutôt que itérative
- Nette accélération de la génération
  - 10^7 hash/sec. sur GTS 450, 4.10^7 sur GTX 580
  - Tables RockYou: 2^{39}/40Go/15h, 2^{47}/1,2To/230j
- Léger ralentissement de la recherche
  - Facteur 50

Résultats

(rainbow classique vs. probabiliste)

Espace parcouru	Taille de la rainbow	Rockyou	500k
LowerAlphaNum 1-7	0,3 Go	46%	9%
LowerAlphaNum 1-8	12 Go	64%	49%
MixedAlphaNum 1-7	15 Go	48%	13%
LowerAlpha 1-9	23 Go	35%	6%

Espace parcouru	Taille de la rainbow	Rockyou	500k
Markov 265 optim	1,04Go (4x267M)	71%	58%
Markov 265	1,11Go(4x283M)	75%	50%
Markov 285 optim	7,99Go (4x2,00G)	79%	69%
Markov 285	8,50Go (4x2,13G)	83%	60%
Markov 300 optim	36,76Go (4x9,19Go)	84%	76%
Markov 300	39,19Go (4x9,80Go)	87%	66%

Contre-mesures

- Saler !
  - H(rnd+pwd), H^{rnd}(pwd), …
  - Bonus : cache les utilisateurs ayant un même mot de passe
- Mots de passe longs
- Caractères spéciaux/régionaux
  - Attention ! Qwerty / Azerty
- Mots de passe aléatoires
  - au moins en partie

Attacking and Fixing PKCS#11 Security Tokens with Tookan

- Graham Steel
  - LSV, INRIA & CNRS & ENS-Cachan
  - Avec R. Focardi, M. Bortolozzo, M. Centenaro
- Travaux déjà présentés à CCS'10

PKCS#11

- Notations
  - h(n,k) : un pointeur sur une clé k
  - Wrap : la clé peut « wrapper » une clé
  - Decrypt : la clé peut déchiffrer une donnée
  - Sensitive : la clé ne peut pas sortir ; une fois posé, l'attribut sensitive ne peut pas être enlevé
  - Extractable : la clé peut sortir si elle est wrappée

Wrap and Decrypt (Clulow, 2003)

- Situation initiale
  - L'intrus connait h(n1,k1) et h(n2,k2)
  - k1 est sensitive et extract, k2 est wrap et decrypt
- Attaque
  - Wrap: h(n2,k2), h(n1,k1)→ $\{\mathbf{k1}\}_{k2}$
  - Decrypt: h(n2,k2), $\{\mathbf{k1}\}_{k2}$ → **k1**
- Résultat
  - K1 est sortie, malgré l'attribut sensitive

Automatisation

- Ces attaques sont difficiles à voir « à l'œil nu »
- Un système formel peut les rechercher systématiquement

Exemple

Source sstic slides

http://www.sstic.org/media/SSTIC2011/SSTIC-actes/Tookan/SSTIC2011-Slides-Tookan-steel.pdf

	Device		Supported Functionality						Attacks found					
	Company	Model	sym	asym	cobj	chan	w	ws	a1	a2	a3	a4	a5	mc
USB	Aladdin	eToken PRO	✓	✓	✓	✓	✓	✓	✓	✓				a1
	Athena	ASEKey	✓	✓	✓									
	Bull	Trustway RCI	✓	✓	✓	✓	✓	✓	✓	✓				a1
	Eutron	Crypto Id. ITSEC		✓	✓									
	Feitian	StorePass2000	✓	✓	✓	✓	✓	✓		✓	✓	✓		a3
	Feitian	ePass2000	✓	✓	✓	✓	✓	✓		✓	✓	✓		a3
	Feitian	ePass3003Auto	✓	✓	✓	✓	✓	✓		✓	✓	✓		a3
	Gemalto	Smart Enterprise Guardian		✓		✓								
	MXI Security	Stealth MXP Bio	✓	✓		✓								
	SafeNet	iKey 2032	✓	✓	✓		✓							
	Sata	DKey	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	a3
Card	ACS	ACOS5	✓	✓	✓	✓								
	Athena	ASE Smartcard	✓	✓	✓									
	Gemalto	Cyberflex V2	✓	✓	✓		✓	✓		✓				a2
	Gemalto	SafeSite Classic TPC IS V1		✓		✓								
	Gemalto	SafeSite Classic TPC IS V2	✓	✓	✓	✓	✓	✓	✓		✓	✓	✓	a3
	Siemens	CardOS V4.3 B	✓	✓	✓		✓					✓		a4
Soft	Eracom	HSM simulator	✓	✓		✓	✓	✓	✓	✓		✓		a1
	IBM	opencryptoki 2.3.1	✓	✓	✓	✓	✓	✓	✓	✓		✓		a1

	Acronym	Description
Supported functionality	sym	symmetric-key cryptography
	asym	asymmetric-key cryptography
	cobj	inserting new keys via `C_CreateObject`
	chan	changing key attributes
	w	wrapping keys
	ws	wrapping sensitive keys
Attacks	a1	wrap/decrypt attack based on symmetric keys
	a2	wrap/decrypt attack based on asymmetric keys
	a3	sensitive keys are directly readable
	a4	unextractable keys are directly readable (forbidden by the standard)
	a5	sensitive/unextractable keys can be changed into nonsensitive/extractable
	mc	first attack found by Tookan

Réactions

- RSA
  - Réponse, enregistrement sous CVE-2010-3321, avis de sécurité émis en octobre 2010
- Safenet (ex. Aladdin)
  - Réponse publique de 2 pages
- Les autres
  - Réponse minimale
  - Demandent **qui d'autre** est vulnérable

SSTIC 2011

STOP