

(WEB)?mail pourquoi mon navigateur RAM

CoRI & IN 2016

(WEB)?mail pourquoi mon navigateur RAM?

Introduction

Rappels

modes de collecte

Précisions techniques

résultats obtenus

Présentation des travaux effectués (exemple de GMAIL)

Démo

<INTRODUCTION>

▶ Nicolas SCHERRMANN

- ▶ Ex Ingénieur LORIA - Ingénieur R&D @ TRACIP
- ▶ nscherrmann@tracip.fr - @nschermi

▶ Pierre VEUTIN

- ▶ Ingénieur Responsable R&D @ TRACIP
- ▶ pveutin@tracip.fr - @Geeko_forensic

<SONDAGE>

- ▶ Qui fait de l'investigation numérique (HDD) ?
- ▶ Qui fait de l'investigation numérique (RAM) ?
- ▶ Qui fait le lien entre les 2 ?
- ▶ Qui a déjà rencontré des fichiers hiberfil.sys ?
- ▶ Quels outils utilisez-vous ?

EnCase / X-Ways / IEF / DFF ... xxd ?

<SUJET DU TALK>

- ▶ Montrer les différences significatives obtenues avec 2 méthodologies d'analyse de la RAM
- ▶ Récupérer les données provenant d'un webmail actuel (GMail) à l'aide d'outils simples et gratuits
- ▶ Objectifs du talk :
 - ▶ Introduire un traitement plus « adapté » des dumps de RAM
 - ▶ Présenter des types de données ayant de la valeur pour un enquêteur / analyste forensique.

<RAPPELS>

- ▶ Les différents modes / outils de collecte de la RAM
 - ▶ le plus « cool » : Cold Boot Attack
 - ▶ le plus « direct » : DMA (Inception)
 - ▶ le plus « natif » : driver noyau / kernel
 - ▶ outils forensiques traditionnels : IEF, X-Ways, FTK Imager, EnCase, ...
 - ▶ outils libres : (win|osx)?pmem, LiME, ...
 - ▶ le moins « volatile » : fichiers de crash, d'hibernation, de pagination, de swap



PINGOUIN QUI CHUTE

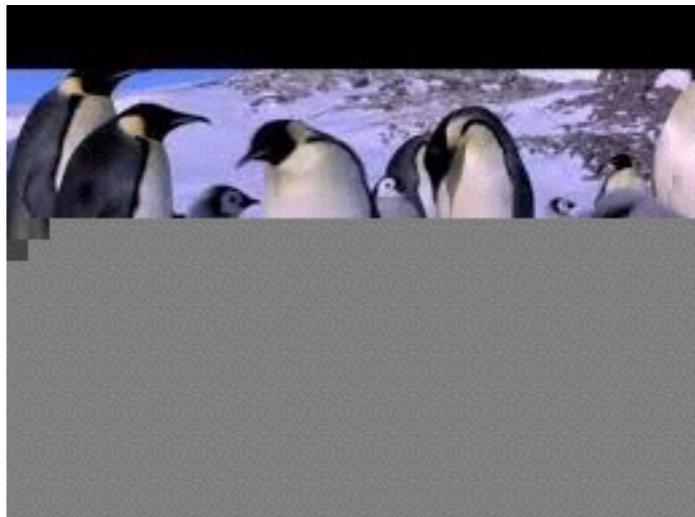
Taille ~ 100 Ko

<UTILISATION D'OUTILS TRADITIONNELS>

Etape 1 : Recherche Google Image : « chute de pingouin » - affichage d'une grande image

Etape 2 : Mise en veille de la machine ou dump -> hiberfil.sys ou dump.dd = 1,11Go

Etape 3 : Analyse du « dump » (Carving des images JPEG)



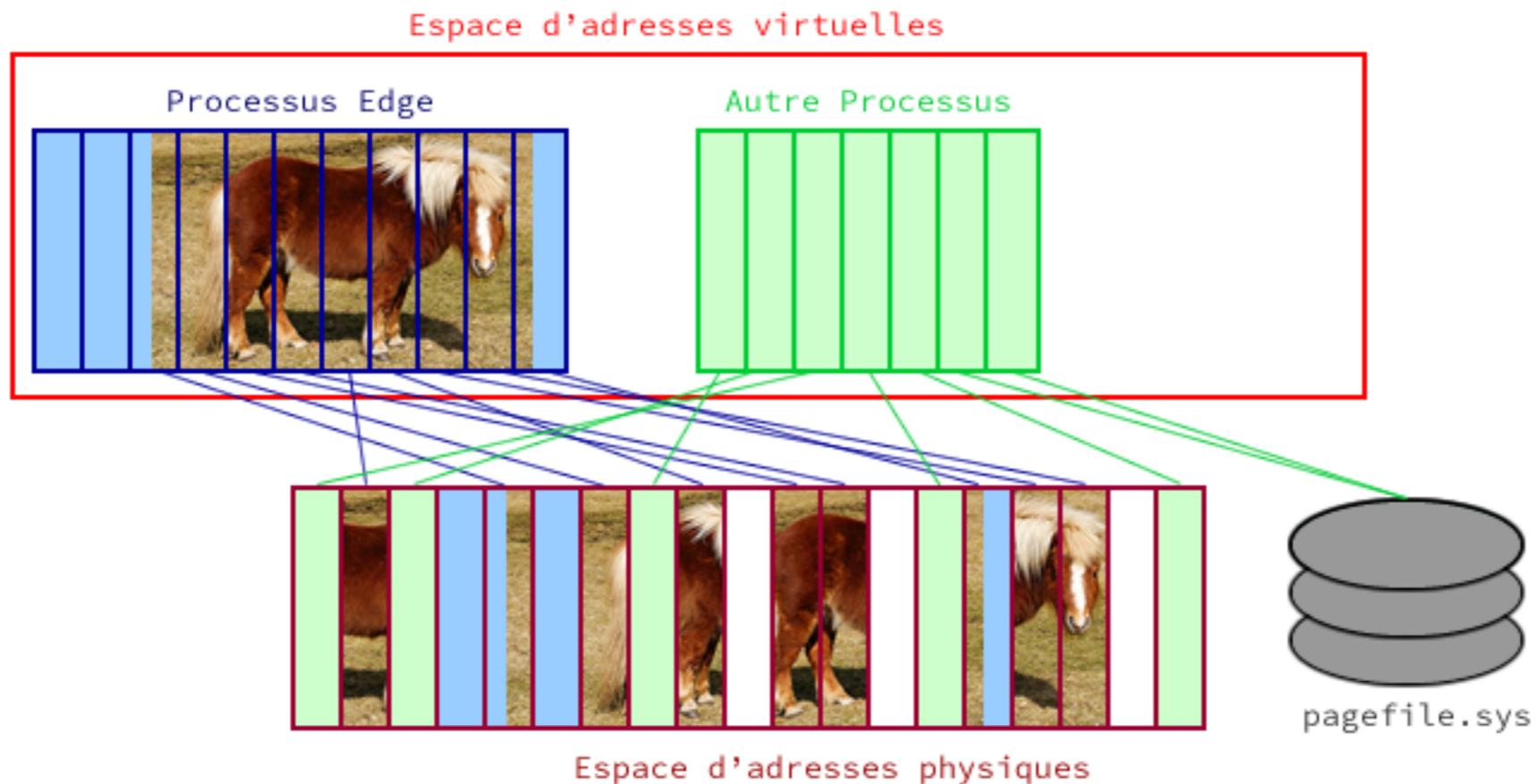
Outils	# fichiers	Pingouin
IEF	469	<i>non</i>
EnCase 7	478	<i>non</i>
X-Ways	510	<i>non</i>
FTK	519	<i>non</i>
APF	593	<i>non</i>
scalpel	520	<i>non</i>

**ON PEUT, SUR LES VÉRITÉS DE FAIT, SE PASSER DE
LA DÉMONSTRATION SI L'ON SAIT SE SERVIR DE
L'EXPÉRIENCE.**

ROGER BACON (miam ...)

<PRÉCISIONS TECHNIQUES>

Explication des résultats



- ▶ Gestion de pages de 4 Ko - ASLR
- ▶ Problème : Outils traditionnels = analyse séquentielle

<ADAPTATION DE LA MÉTHODE>

- ▶ 1 / Reconstruction des plages d'adresses (Utilisation d'outils dédiés)

```
dfir:~/ $ vol.py -f dump.dd --profile=Win10x86 pslist
```

```
dfir:~/ $ vol.py -f dump.dd --profile=Win10x86 memmap -p PID
```

Virtual	Physical	Size	DumpFileOffset
---------	----------	------	----------------

0x008c0000	0x1c3e8000	0x1000	0x0
------------	------------	--------	-----

0x008d0000	0x18be5000	0x1000	0x1000
------------	------------	--------	--------

0x008e0000	0x3e20b000	0x1000	0x2000
------------	------------	--------	--------

<—truncated—>

0xffd06000	0x00001000	0x1000	0x1fc35000
------------	------------	--------	------------

0xffd09000	0x00109000	0x1000	0x1fc36000
------------	------------	--------	------------

0xffdf0000	0x00398000	0x1000	0x1fc37000
------------	------------	--------	------------

<ADAPTATION DE LA MÉTHODE (SUITE)>

- ▶ 2 / Extraction de la plage mémoire dédiée au navigateur (Microsoft Edge)

```
dfir:~/ $ vol.py -f dump.dd --profile=Win10x86 memdump -D . (-p PID_Edge)
```

- ▶ 3/ Carving (exemple avec scalpel)

```
dfir:~/ $ scalpel edge.dmp
```

<EXPLICATIONS>

- ▶ Tous les outils retrouvent le fichier recherché
- ▶ Le nombre de fichiers récupérés est plus restreint* mais plus qualitatif



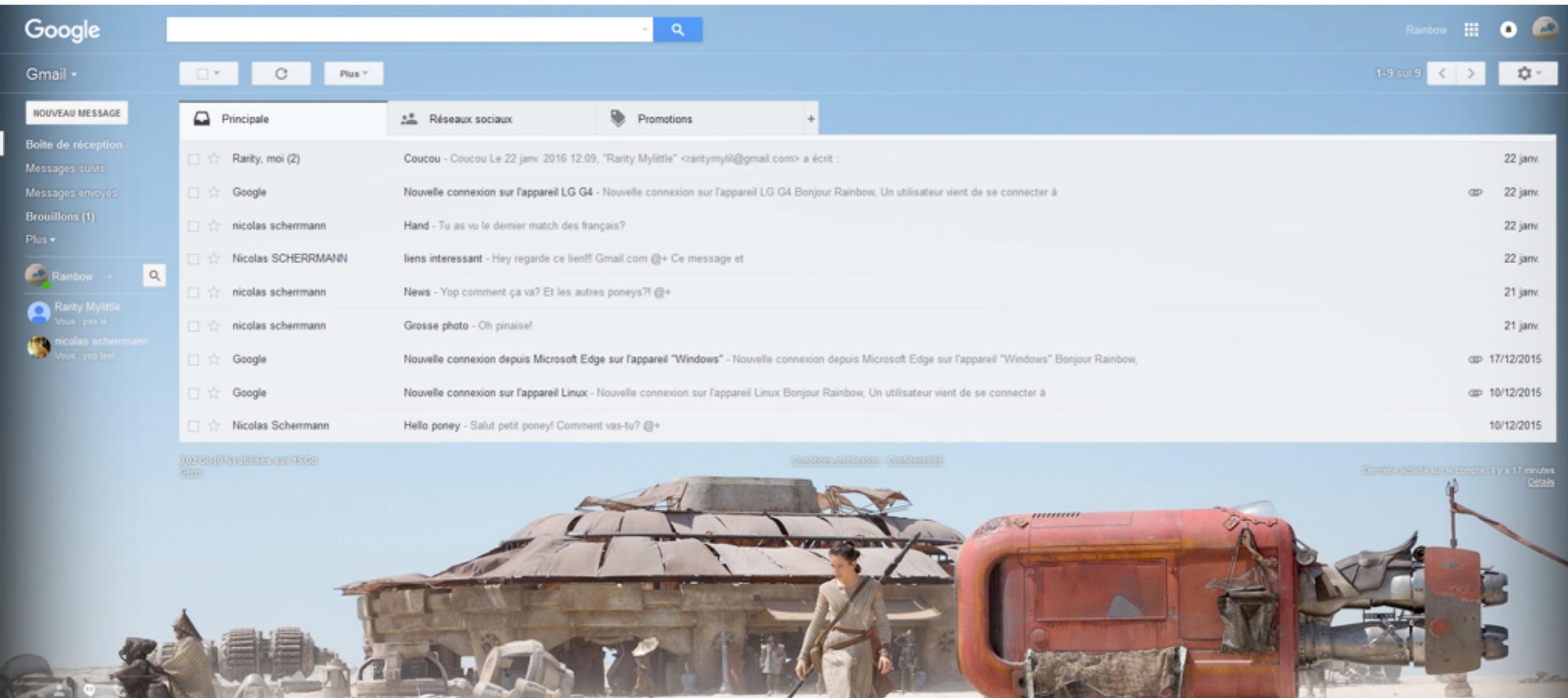
Outils	# fichiers	Pingouin
IEF	116	<i>oui</i>
EnCase 7	119	<i>oui</i>
X-Ways	120	<i>oui</i>
FTK	121	<i>oui</i>
APF †	118	<i>oui</i>
scalpel	119	<i>oui</i>



ANALYSE EN PROFONDEUR

(WEB)?mail pourquoi mon navigateur RAM?

<EXEMPLE AVEC LE WEBMAIL GMAIL>



... et quelques poneys

(WEB)?mail pourquoi mon navigateur RAM?

<DERRIÈRE LE NAVIGATEUR>

The screenshot displays the Network tab of Chrome DevTools for the URL `https://mail.google.com/mail/u/0/#inbox`. The interface includes tabs for Inspecteur, Console, Débugueur, Éditeur de st..., Performances, Réseau, Paramètres, Réponse, Délais, Sécurité, and Aperçu. The Réseau tab is active, showing a list of requests with columns for Méthode, Fichier, En-têtes, Cookies, Paramètres, Réponse, Délais, Sécurité, and Aperçu. The list includes various GET and POST requests for resources like `/mail/u/0/`, `bind?ctype=hangouts&prop=gmail&appver=c...`, `rs=AHGWq9AcCQuntVo8eH--hZ2PQ8a_p0-DBw`, `light_hilt_ldpi_v1.png`, `clear dot.gif?zx=2ggjnhs3igke`, `setgmail?zx=a8ob92ux5e58`, `/availability/?s=gmail&a=viewinbox&c=scs..`, `/mail/u/0/?ui=2&ik=4962bc11da&view=ad&ak.`, `/mail/u/0/?ui=2&ik=4962bc11da&rid=maili.b9.`, `v1_4593b7d7.png`, `photo.jpg`, `arrow_down_white.png`, `yUGOIENX1coZXEPRIFnEpeWLYbNwlZkdiC6.`, `clear dot.gif`, `rs=AHGWq9AcCQuntVo8eH--hZ2PQ8a_p0-DBw`, `clear dot.gif`, `arrow_down.png`, `tabicons_black.png`, `plus_black.png`, `star4.png`, `/mail/u/0/?ui=2&view=dim&iv=17s2paofw..`, and `contacts_white_Icon21.png`. The response for the last request is visible, showing HTML content with a highlighted email snippet: `33enicolas scherrmann\u003c/span\u003e", "\u0026raquo;\u0026nbsp;", "Grosse photo", "Oh pinaise!",0,"", "21 janv.", "21 janvier 2016 à 20:51",`. The bottom status bar indicates 136 requêtes, 2,33 Ko, and 129,78 s.

<JSON>

- ▶ L'analyse réseau de la page d'accueil fait apparaître des structures JSON de ce type :

```
[ "^a[l]", "^i", "^im", "^io_im", "^io_imc1", "^io_lr", "^o", "^smartlabel_personal" ],  
 [ ], "\u003cspan class\u003d\"yP\" email\u003d\"nico.scherrmann@gmail.com\" name  
\u003d\"nicolas scherrmann\" \u003enicolas scherrmann\u003c/span  
\u003e", "\u0026raquo;\u0026nbsp;", "Grosse photo", "Oh pinaise!", 0, "", "", "21  
janv.", "21 janvier 2016 à 20:51",
```

- ▶ Ce qui donne en décodant l'unicode :

```
[ "^a[l]", "^i", "^im", "^io_im", "^io_imc1", "^io_lr", "^o", "^smartlabel_personal" ],  
 [ ], "<span class=\"yP\" email=\"nico.scherrmann@gmail.com\" name=\"nicolas  
scherrmann\">nicolas scherrmann</span>", "&raquo;&nbsp;", "Grosse photo", "Oh  
pinaise!", 0, "", "", "21 janv.", "21 janvier 2016 à 20:51", 1453450240294000, [, , 0,  
[, , [, , "3", [1], , "nico.scherrmann@gmail.com", , , 0, 0]
```

<A LA RECHERCHE DES STRINGS>

```
$ strings dump.dd | grep -C2 "smartlabel" | more
--
["^all", "^cob-processed-
gmr", "^i", "^io_unim", "^os_social", "^smartlabel_social", "^sq_ig_i_social", "^u", "
^unsub"], "\u003cspan class\u003d\"zF\" email\u003d\"notification
+kjdkk3vdw5__@facebookmail.com\" name\u003d\"Facebook\" \u003eFacebook\u003c/
span\u003e", "\u003cb\u003e\u0026raquo;\u003c/b\u003e\u0026nbsp;", "\u003cb
\u003eWobniar, vous avez 2 nouvelles notifications et 1 message\u003c/b
\u003e", "Pas mal de choses se sont pass\u00e9es sur Facebook depuis votre derni\u00e8re
connexion. Voici quelques", 0, "", "", "\u003cb\u003e21/12/2015\u003c/b\u003e", "21
d\u00e9cembre 2015
--
```

<GARDER LE CONTACT>

- ▶ La recherche d'adresses connues permet de retrouver ces contacts et aussi leur structure :

```
[1,"nicolas scherrmann","nicolas","//lh5.googleusercontent.com/-s-w-tth9bMo/AAABAAABAAI/AAAABAAABoM/pq0J06u7l5o/photo.jpg",  
["nicoscherrmann@gmail.com","nico.scherrmann@gmail.com","nscherrmann@tracip.fr","n.i.c.o.s.c.h.e.r.r.m.a.n.n@gmail.com"],["06 12 34 56 78"],,,1,2,[],[],"nicoscherrmann@gmail.com"]
```

- ▶ Nous identifions ici :
 - ▶ un id
 - ▶ le nom complet
 - ▶ le nom
 - ▶ l'url de l'image de profil
 - ▶ une liste d'adresses mails
 - ▶ le numero de téléphone
 - ▶ l'adresse mail

<COMMENCER UNE CORRESPONDANCE>

- ▶ Pour les mails, nous retrouvons la structure vue précédemment :

```
[ "^all", "^i", "^jim", "^io_im", "^io_imc1", "^io_lr", "^o", "^smartlabel_personal" ], [ ], "\u003cspan class\u003d\n\"yP\" email\u003d\n\"nico.scherrmann@gmail.com\" name\u003d\n\"nicolas scherrmann\" \u003enicol\nas scherrmann\n\u003c/span\u003e\", \"\u0026\nraquo;\u0026nbsp\n;\", \"Grosse photo\", \"0h\npinaise!\", 0, \"\", \"\", \"21\njanv.\", \"21 janvier\n2016\"
```

- ▶ Une fois décodée :

```
[ "^all", "^i", "^jim", "^io_im", "^io_imc1", "^io_lr", "^o", "^smartlabel_personal" ], [ ], "<span class=\n\"yP\" email=\n\"nico.scherrmann@gmail.com\" name=\n\"nicolas scherrmann\">nicolas\nscherrmann</span>\", \"&\nraquo;\u0026nbsp\n;\", \"Grosse photo\", \"0h\npinaise!\", 0, \"\", \"\", \"21\njanv.\", \"21 janvier\n2016\"
```

- ▶ Nous identifions ici :
 - ▶ une structure contenant le tag du message (ici *personal*)
 - ▶ l'adresse mail et le nom de l'expéditeur
 - ▶ le sujet du mail
 - ▶ une partie du message
 - ▶ la date du message

<VALIDATION DES RESULTATS>

▶ navigateur :

```
[ "^a[l]", "^i", "^im", "^io_im", "^io_imc1", "^io_lr", "^o", "^smartlabel_personal" ],  
[], "<span class='yP' email='nico.scherrmann@gmail.com' name='nicolas  
scherrmann'>nicolas scherrmann</span>", "&raquo;&nbsp;", "Grosse photo", "Oh  
pinaise!", 0, "", "", "21 janv.", "21 janvier 2016 à 20:51 »", 1453450240294000, [],  
0, [], [], "3", [1], "nico.scherrmann@gmail.com",,,,0,0]
```

▶ en RAM :

```
[ "^a[l]", "^i", "^im", "^io_im", "^io_imc1", "^io_lr", "^o", "^smartlabel_personal" ],  
[], "<span class='yP' email='nico.scherrmann@gmail.com' name='nicolas  
scherrmann'>nicolas scherrmann</span>", "&raquo;&nbsp;", "Grosse photo", "Oh  
pinaise!", 0, "", "", "21 janv.", "21 janvier 2016
```

il ne manque pas grand chose ... dans la regex « à »

<FAIRE PASSER LE MESSAGE>

- ▶ Pour les hangouts, les structures sont les suivantes :

```
[[0, "Comment a va?"]]
```

- ▶ Il n'y a donc aucune méta donnée ... pour le moment ... :-)

(WEB)?mail pourquoi mon navigateur RAM?

DÉMONSTRATION

(WEB)?mail pourquoi mon navigateur RAM?

((COMMENTAIRE) | (SUGGESTION)) ?

rot13(rainbowdash)@gmail.com



MERCI.

N. Scherrmann

P. VEUTIN