ALESSANDRO DE VITO *(@_cube0x8)*

truelit

HOW I MET YOUR BROWSER: GOING INCOGNITO DOESN'T HIDE YOUR BROWSING FROM RAGAMUFFIN

#OSDFCon

OSDFCON 2017 – HERNDON (VIRGINIA)

AGENDA

- Introduction
  - What is Chrome Ragamuffin
  - What Chrome Ragamuffin is not
  - Why Chrome Ragamuffin should be useful
  - Goal
  - Tools of the trade
- Forensic Overview
  - Chrom(e|ium) overview
  - Objects we have focused on
  - Case Study
  - Chrome Ragamuffin architecture
  - State of art
  - Work in progress and future works

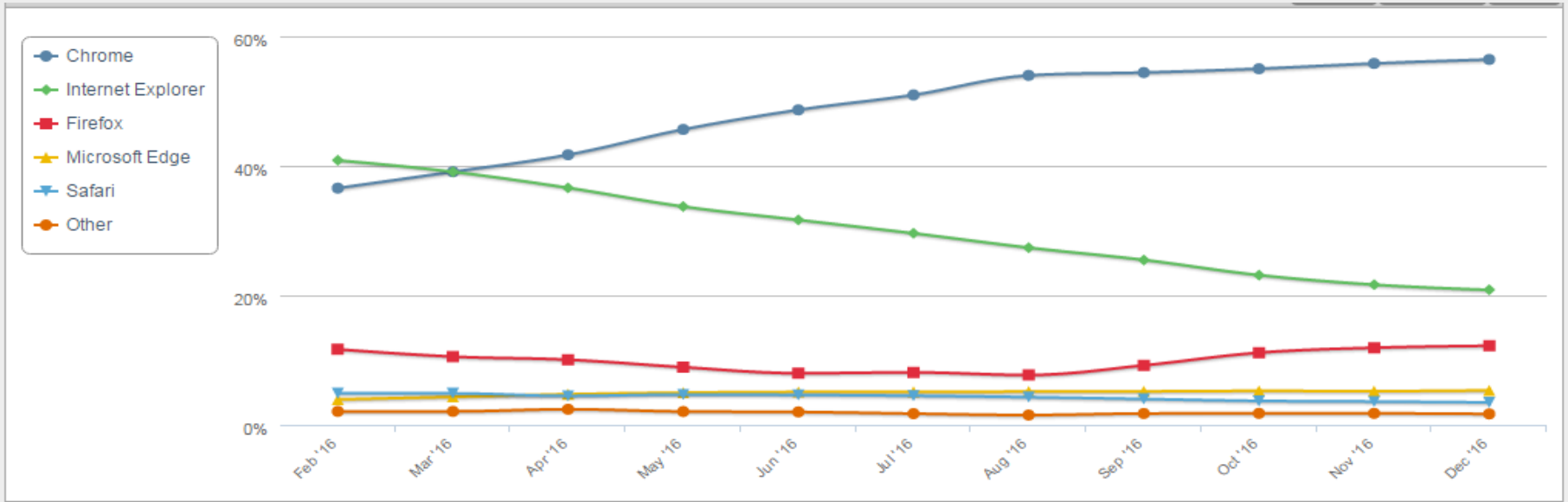
WHAT IS CHROME RAGAMUFFIN?

- Internal research project
- It is a memory forensics tool
- Three main functionalities:
  - Detailed overview about user navigation
  - Web page contents
  - Client-site attacks traces

WHAT CHROME RAGAMUFFIN IS NOT

- IDS/NIDS
- Browser extension
- Anti-virus
- SQLite databases analyzer

WHY CHROME RAGAMUFFIN SHOULD BE USEFUL



MONTH	CHROME	INTERNET EXPLORER	FIREFOX	MICROSOFT EDGE	SAFARI	OTHER
February, 2016	36.56%	40.85%	11.68%	3.94%	4.88%	2.08%
March, 2016	39.09%	39.10%	10.54%	4.32%	4.87%	2.09%
April, 2016	41.71%	36.61%	10.06%	4.73%	4.47%	2.42%
May, 2016	45.63%	33.71%	8.91%	4.99%	4.69%	2.07%
June, 2016	48.65%	31.65%	7.98%	5.09%	4.64%	1.99%
July, 2016	50.95%	29.60%	8.12%	5.09%	4.51%	1.73%
August, 2016	53.97%	27.38%	7.69%	5.16%	4.28%	1.51%
September, 2016	54.41%	25.48%	9.19%	5.16%	4.00%	1.75%
October, 2016	54.99%	23.13%	11.14%	5.26%	3.69%	1.79%
November, 2016	55.83%	21.66%	11.91%	5.21%	3.61%	1.78%
December, 2016	56.43%	20.84%	12.22%	5.33%	3.47%	1.70%

- Most used browser
- We hadn't tools to analyze web browser internals
- It can **find** a lot of **new artifacts**

truelit

TOOLS OF THE TRADE

- Source code analysis

- Debugging and data structures analysis

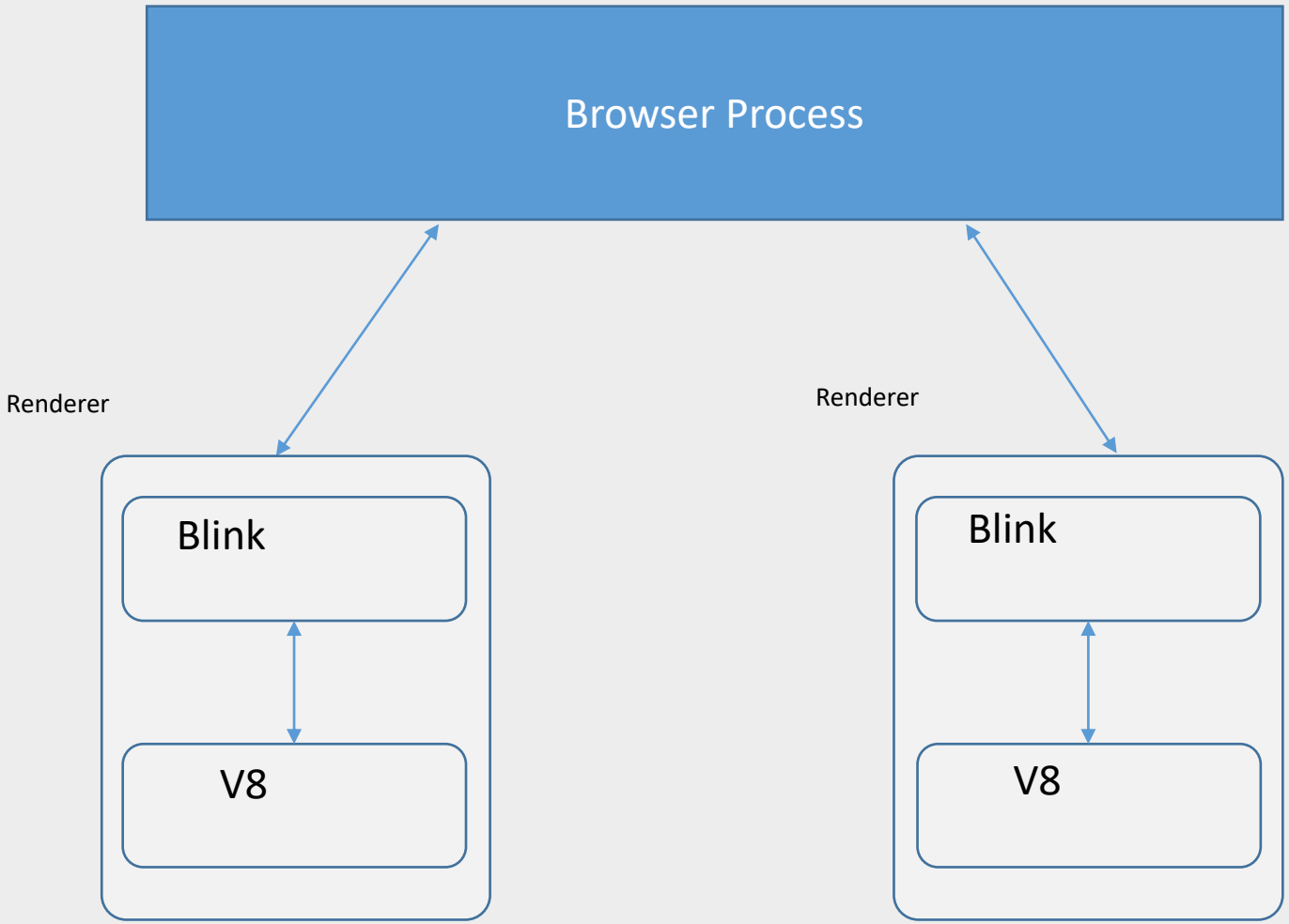
- Data structures extraction

- to implement the Proof of Concept

truelit

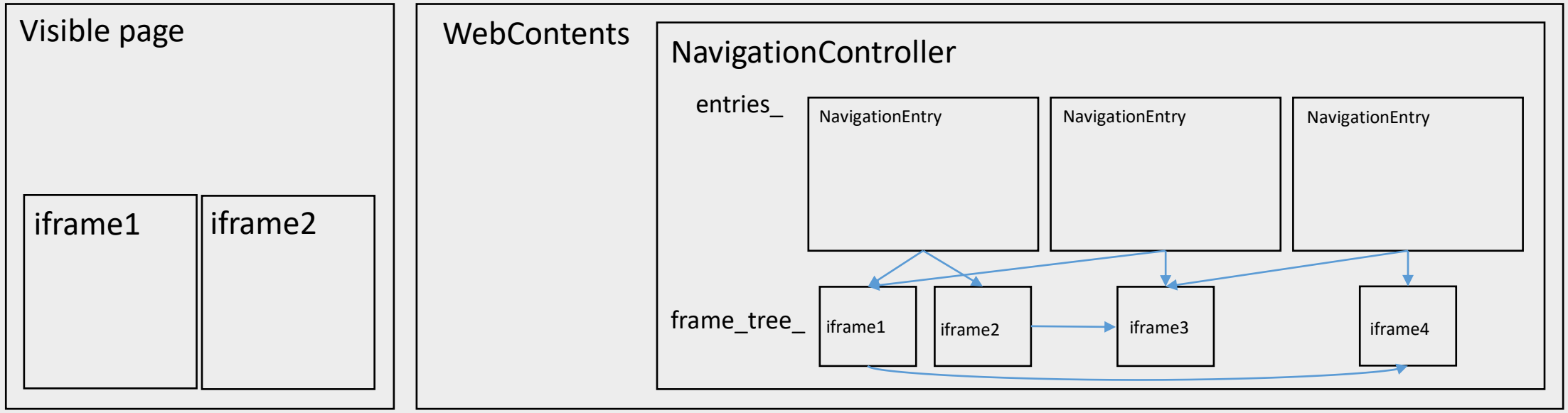
FORENSIC OVERVIEW

GOOGLE CHROME OVERVIEW



OBJECTS WE HAVE FOCUSED ON

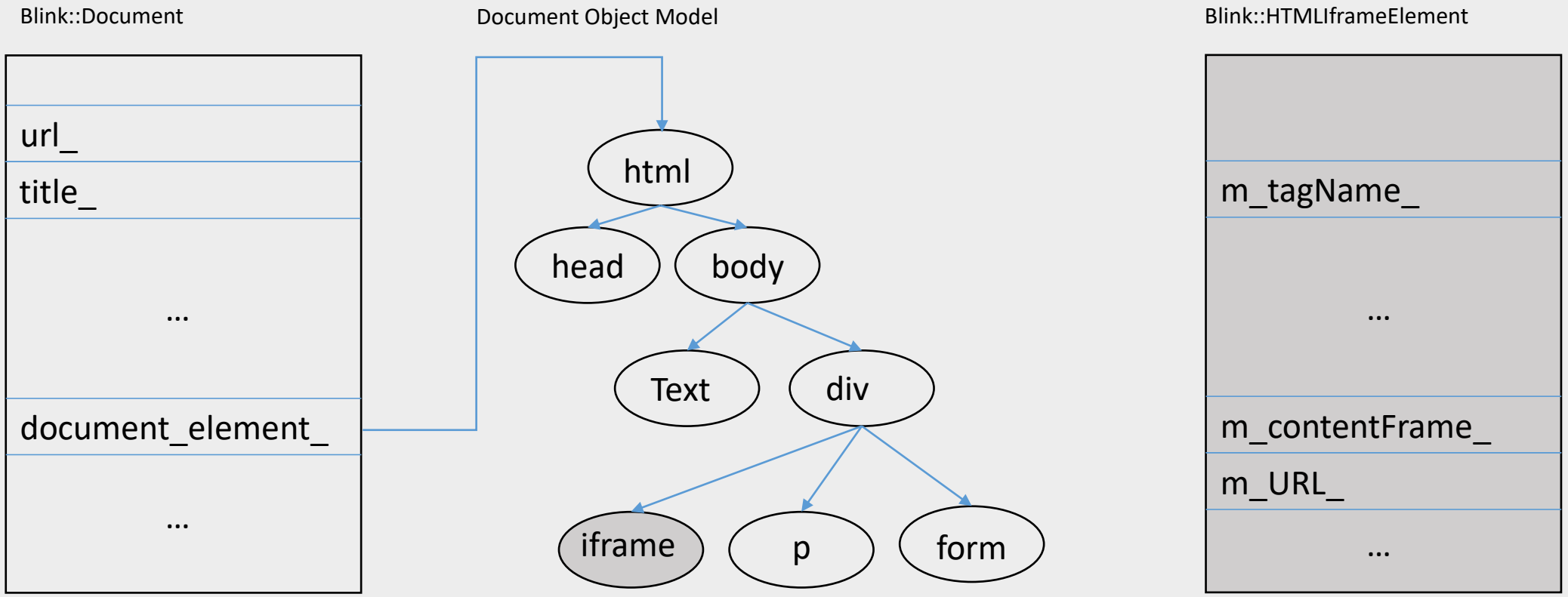
Browser process



truelit

OBJECTS WE HAVE FOCUSED ON

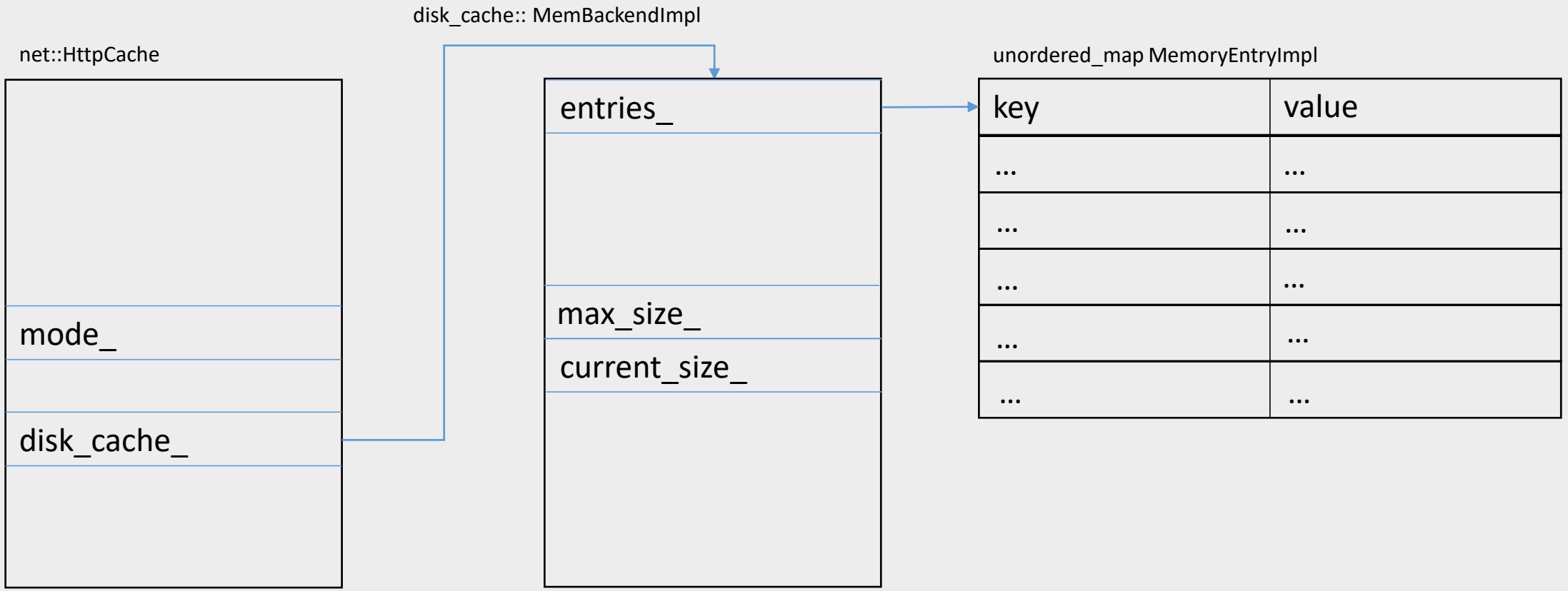
Renderer process



truelit

OBJECTS WE HAVE FOCUSED ON

Browser process



truelit

OBJECTS WE HAVE FOCUSED ON
Example 1

Browser objects:

- Evidence: offset object, url, status code, method, transition, timestamp, restore type, page type, **form params**

ID	Offset	Title	User typed url	Original request url	Status code	Method	PageState address	Transition	Referer	UTC Timestamp	Restore Type	Page type
5	2.5719 1E+12	None	**http://192.168.0.11 /example1.html**	http://192.168.0.11 /example1.html	200	GET	None	**User used the address bar to trigger this navigation**	None	10/1/2017 16:12	Entry was not restored	NORMAL
5	2.5719 1E+12	frame_entry_ object	http://192.168.0.11/ **iframe1**.html	http://192.168.0.11 /iframe1.html	None	GET	None	None	**http://192.168.0.11/example1.html**	10/1/2017 16:12	None	**<!--framePath //<!--frame0-->-->**
5	2.5719 1E+12	frame_entry_ object	http://192.168.0.11/ **iframe2**.html	http://192.168.0.11 /iframe2.html	None	GET	None	None	**http://192.168.0.11/example1.html**	10/1/2017 16:12	None	**<!--framePath //<!--frame1-->-->**
7	2.5718 8E+12	frame_entry_ object	http://192.168.0.11/ iframe3.html	http://192.168.0.11 /iframe2.html	None	GET	None	None	http://192.168.0.11/iframe1.html	10/1/2017 16:12	None	<!--framePath //<!--frame0-->-->
8	2.5719 E+12	frame_entry_ object	http://192.168.0.11/ iframe3.html	http://192.168.0.11 /iframe3.html	None	GET	None	None	http://192.168.0.11/iframe2.html	10/1/2017 16:12	None	<!--framePath //<!--frame1-->-->

truelit

OBJECTS WE HAVE FOCUSED ON
Example 1

Renderer objects:

- Evidence: PID of the **tab** which **contains** the **specific document**, document **offset**, **URL** of the document, **title**, **<html>** node address of the DOM tree

```
✘ cubo@cubo /run/media/cubo/7E8B39162F6BE7AA/dump vol.py --plugins $SECURITY/tools/volatility/plugins/chrome_ragamuffin -f example1.vmem --profile Win10
x64_14393 chrome_ragamuffin --analysis renderer -p 4088
Volatility Foundation Volatility Framework 2.6
Pid      Document offset      URL                                                Title                                              DOM start address
-------- -------------------- -------------------------------------------------- -------------------------------------------------- ------------------
    4088 0x3e7ca5a25c0        http://192.168.0.11/example1.html                  PROVA                                              0x3e7ca5a3210
    4088 0x3e7ca5a9220        http://192.168.0.11/iframe3.html                   404 Not Found                                      0x3e7ca5a9ed0
    4088 0x3e7ca5aa588        http://192.168.0.11/iframe3.html                   404 Not Found                                      0x3e7ca5ab238
cubo@cubo /run/media/cubo/7E8B39162F6BE7AA/dump
```

truelit

OBJECTS WE HAVE FOCUSED ON
Example 1

Renderer objects:

Evidence: By the "**DOM start address**" field we can **get** the entire **Document Object Model tree** in its **dot** (**high-level** structure of the page) and **text** notation (**detailed** contents)

Node tag: iframe
Node attributes: {'src': 'iframe1.html'}
Memory offset: 0x3e7ca5a3650
Contained document offset: 0x3e7ca5aa588

Node tag: iframe
Node attributes: {'src': 'iframe2.html'}
Memory offset: 0x3e7ca5a4570
Contained document offset: 0x3e7ca5a9220

OBJECTS WE HAVE FOCUSED ON
Cross-Site Request Forgery

Browser objects:

- Evidence: offset object, url, status code, method, transition, timestamp, restore type, page type, **form params**

Entry ID	Offset	Title	User typed url	Original request url	Status code	Method	PageState address	Transition	Referer	UTC Timestamp	Restore type	Page type
2	0x192d2a03c60	None	http://192.168.1.124/notexists.html	http://192.168.1.124/notexists.html	0	GET	None	Typed URL in the address bar	None	18/09/17 14.50	**Entry was not restored**	**ERROR**
3	0x192d2a02900	Test	http://192.168.1.124/index.html	http://192.168.1.124/index.html	200	GET	None	Typed URL in the address bar	None	18/09/17 14.50	Entry was not restored	**NORMAL**
3	0x192D2508540	None	http://192.168.1.124/changepassword.php	http://192.168.1.124/changepassword.php	None	**POST**	**0x192D64A2EE0**	None	http://192.168.1.124/hghrueguoeir.html	19/09/17 14.50	None	**<!--framePath //<!--frame0--->**

truelit

OBJECTS WE HAVE FOCUSED ON
Cross-Site Request Forgery

POST Params:

- With the memory address, we can dump the **PageState** object which contains **serialized information** about the **submitted form**

```
In [18]: i.dump_page_state()
Out[18]: '\xac\x01..\x19.......N...h.t.t.p.:./.1.9.2...1.6.8...1...1.2.4./.c.h.a.n.g.e.p.a.s.s.w.o.r.d...p.h.p...@.
..<.!.-.-.f.r.a.m.e.P.a.t.h. ./.<.!.-.-.f.r.a.m.e.0.-.-.>.-.-.>.....L...h.t.t.p.:./.1.9.2...1.6.8...1...1.2.4./.h
.g.h.r.u.e.g.u.o.e.i.r...h.t.m.l.....\xab\x02\x07\x85\xafY\x05.\xac\x02\x07\x85\xafY\x05.\x02..........\x01...\x01..
.....-...new_username=username&new_password=qwerty1345...\x0e\xb8\x07\x85\xafY\x05.....B...a.p.p.l.i.c.a.t.i.o.n./.x.
-.w.w.w.-.f.o.r.m.-.u.r.l.e.n.c.o.d.e.d.......'

In [19]:
```

truelit

OBJECTS WE HAVE FOCUSED ON
Cross-Site Request Forgery

Renderer objects:

- Evidence: PID of the **tab** which **contains** the **specific document**, document **offset**, **URL** of the document, **title**, **<html>** node address of the DOM tree

```
cube@trallallino ~/security/tools/ragamuffin/dump develop ● vol.py --plugins ~/security/tools/volatility/plugins/chrome_ragamuffin -f windows_csrf.
vmem --profile Win10x64_14393 chrome_ragamuffin --analysis renderer
Volatility Foundation Volatility Framework 2.6
Pid      Document offset      URL                                                Title                                              DOM start address
-------- -------------------- -------------------------------------------------- -------------------------------------------------- ------------------
    4916 0x11dad2f8380        https://www.google.it/_/.../newtab?espv=2&ie=UTF-8 None                                               0x11dad2f9030
    4916 0x11dad4025c0        chrome-search://most-vis...%20su%20questa%20pagina None                                               0x11dad403270
    2212 0x3416fde25c0        chrome://apps/                                     Nuova scheda                                       0x3416fde3270
    3268 0x31da1fe25c0        https://it-it.facebook.com/                        Facebook: accedi o iscriviti                       0x31da1fe3270
    3268 0x31da21101e8        https://staticxx.faceboo...ommon/referer_frame.php None                                               0x31da2110e98
    5416 0x1e1842825c0        chrome://settings/                                 Impostazioni                                       0x1e184283270
    5416 0x1e184327ab8        about:blank                                        None                                               0x1e1843286d8
    5416 0x1e184329680        about:blank                                        None                                               0x1e18432a2a0
    5416 0x1e18432b248        about:blank                                        None                                               0x1e18432be68
    3212 0x216b1d41838        None                                               None                                               0x216b1d42488
    3212 0x216b1d425c0        http://192.168.1.124/index.html                    PROVA                                              0x216b1d43210
```

truelit

OBJECTS WE HAVE FOCUSED ON
what do we get from those?

Renderer objects:

Evidence: By the "**DOM start address**" field we can **get** the entire **Document Object Model tree** in its **dot** (**high-level** structure of the page) and **text** notation (**detailed** contents)

Node tag: iframe
Node attributes: {'src': **'hghrueguoeir.html**', 'style': **'display: none**;'}
Memory offset: 0x216b1d43650
Contained document offset: 0x216b1d41838

- Node tag: html
Node attributes: {lang=it}
Memory offset: 0x2669a62b8b8
- Node **tag**: head
Node attributes: {}
Memory offset: 0x2669a62b920
- Node tag: body
Node **attributes**: {class=test, id=123}
Memory offset: 0x2669a62b988
- Node tag: h1
Node attributes: {id=title}
Memory offset: 0x2669a62b9f0
- Node tag: **Text**
Content: **You've successfully changed your password**

truelit

CHROME RAGAMUFFIN ARCHITECTURE

Implemented in two parts:

1. *libchrome_$release.py* library
   - **VTypes**
   - **WTF::StringImpl, Vector** objects and other **platform-specific data types**
2. *chrome_ragamuffin.py* plugin
   - This is the **main plugin**.
   - **signatures**, **validation,** render the **output**

STATE OF ART

Other tools

- WebCapsule/ChromePic
  - **Instrumentation** of the web browser source code
  - Records and Replay **key logger**

- Chrome History (@superponible)
  - **SQLite databases** in memory (visited pages, cookies, search terms ecc.)
  - SQLite databases are **saved on disk**

STATE OF ART

Chrome Ragamuffin

- Pro
  - Agnostic approach
  - Whole address space (a lot of new artifacts)
  - Overcoming incognito mode

STATE OF ART

Chrome Ragamuffin

- Limitations
  - Garbage Collector (Olipan, Scavenger ecc.) **collects** unused **objects**

WORK IN PROGRESS

- ⏳ HTTP cached Response Body (Work in progress)
- ⏳ V8 (for now, Isolate, Heap, Spaces, Page Memories) (Almost-Work-in-progress)
- ⏳ Automatic VTypes extraction
- ⏳ Linux/macOSx support

THANKS

- Join the project on github! Search for *cube0x8 (“cube” “zero” “x” “eight”) (https://github.com/cube0x8/chrome_ragamuffin)*
- Email: **alessandro.devito@truel.it**

truelit