

AIL Framework for Analysis of Information Leaks

Practical and Efficient Data-Mining of Suspicious Websites, Forums and Tor Hidden-Services



CIRCL

Computer Incident
Response Center
Luxembourg

Alexandre Dulaunoy

alexandre.dulaunoy@circl.lu

Aurelien Thirion

aurelien.thirion@circl.lu

Jean-Louis Huynen

jean-louis.huynen@circl.lu

info@circl.lu

March 8, 2022

Links

- AIL project <https://github.com/ail-project>
- AIL framework
<https://github.com/ail-project/ail-framework>
- Training materials
<https://github.com/ail-project/ail-training>
- Online chat <https://gitter.im/ail-project/community>

Legal and Ethics

Privacy, AIL and GDPR (PII)

- Many modules in AIL can process personal data and even special categories of data as defined in GDPR (Art. 9).
- The data controller is often the operator of the AIL framework (limited to the organisation) and has to define **legal grounds for processing personal data**.
- To help users of AIL framework, a document is available which describe points of AIL in regards to the regulation¹.

¹[https:](https://www.circl.lu/assets/files/information-leaks-analysis-and-gdpr.pdf)

[//www.circl.lu/assets/files/information-leaks-analysis-and-gdpr.pdf](https://www.circl.lu/assets/files/information-leaks-analysis-and-gdpr.pdf)

Potential legal grounds

- **Consent of the data subject** is in many cases not feasible in practice and often impossible or illogical to obtain (Art. 6(1)(a)).
- Legal obligation (Art. 6(1)(c)) - This legal ground applies mostly to CSIRTs, in accordance with the powers and responsibilities set out in CSIRTs mandate and with their constituency, as they may have the legal obligation to collect, analyse and share information leaks without having a prior consent of the data subject.
- Art. 6(1)(f) - Legitimate interest - Recital 49 explicitly refers to CSIRTs' right to process personal data provided that they have a legitimate interest but not colliding with fundamental rights and freedoms of data subject.

Ethics in Information Security and Cybersecurity

- The materials and tools presented can open a significant numbers of questions regarding ethics;
- Our researches and tools are there for education, supporting the public good and improve incident response;
- We ask all users and participants to **follow ethical principles and act professionally**².

²<https://www.acm.org/code-of-ethics>

<https://www.first.org/global/sigs/ethics/ethics-first>

Introduction

Concepts - Deep Web

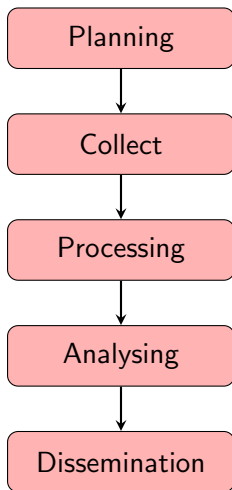
- **Deep Web** is the part of World Wide Web not indexed or directly accessible by standard web search-engines;
- This can be content hidden from **crawlers** by requiring a specific access and this can includes private social media, password-protected forums or content protected by different measures such as paywalls or specific security interface to access the information;
- A large portion of content accessible via Internet is part of the deep web³.

³also called invisible web, hidden web or non-indexed web

Concepts - darknet

- **Darknet** is an overlay network running on top of Internet requiring specific software to access the network and its services;
- Tor, I2P and Freenet are the most commonly used ones. Many are used for hidden services access and some for proxy access to the Internet;
- There are **legitimate use-cases** for such network but also many **illegal or criminal usage**.

Lifecycle of collection and analysis



Collecting, processing and analysing content - web pages

- Building a search engine on the web is a challenging task because:
 - it has to crawl webpages,
 - it has to make sense of **unstructured data**,
 - it has to **index** these data,
 - it has to provide a way to retrieve data and structure data (e.g. correlation).
- Doing so on Tor is even more challenging because:
 - services don't always want to be found,
 - parts of the dataset have to be discarded.
- in each case, it requires a lot of bandwidth, storage and computing power.

Collecting, processing and analysing content - structured data

- Some data are structured and are easy to process:
 - metadata!
 - API responses.
- Some even provide cryptographic evidences:
 - authentication mechanisms between peers,
 - OpenPGP can leak a lot of metadata
 - key ids,
 - subject of email in thunderbird,
 - Bitcoin's Blockchain is public,
 - pivoting on these data with external sources yields interesting results.

AIL design Objectives

Objectives of the session

- Show how to use and extend an open source tool to monitor web pages, pastes, forums and hidden services
- Explain challenges and the design of the AIL open source framework
- Review different **collection mechanisms** and **sources**
- Learn how to create new modules
- Learn how to use, install and start AIL
- **Supporting investigation using the AIL framework** and including it in cyber threat intelligence lifecycle

AIL Framework

From a requirement to a solution: AIL Framework

History:

- AIL initially started as an **internship project** (2014) to evaluate the feasibility to automate the analysis of (un)structured information to find leaks.
- In 2019, AIL framework is an **open source software** in Python. The software is actively used (and maintained) by CIRCL and many organisations.
- In 2020, AIL framework is now a complete project called **ail project**⁴.

⁴<https://github.com/ail-project/>

Capabilities Overview

Common usage

- **Check** if mail/password/other sensitive information (terms tracked) leaked
- **Detect** reconnaissance of your infrastructure
- **Search** for leaks inside an archive
- **Monitor** and crawl websites

Support CERT and Law Enforcement activities


- Proactive investigation: leaks detection
 - List of emails and passwords
 - Leaked database
 - AWS Keys
 - Credit-cards
 - PGP private keys
 - Certificate private keys
- Feed Passive DNS or any passive collection system
- CVE and PoC of vulnerabilities most used by attackers

Support CERT and Law Enforcement activities

- Website monitoring
 - monitor booters
 - Detect encoded exploits (WebShell, malware encoded in Base64, ...)
 - SQL injections
- Automatic and manual submission to threat sharing and incident response platforms
 - MISP
 - TheHive
- Term/Regex/YARA monitoring for local companies/government

Sources of leaks

Mistakes from users:



remove_password

[Pull requests](#) [Issues](#) [Marketplace](#) [Gist](#)

[Repositories](#) **135**

[Code](#) **1K**

[Commits](#) **322K**


[Issues](#)

[Wikis](#)


[Users](#)

322,302 commit results


Sort: **Best match** ▾




Make remove_password actually work
javitonino committed to freaktiful/cartodb on 1 Mar




def411c







remove password
wenlei committed to cjw1990/wap_demo 2 days ago




e9611e0






remove password
yejune committed to yejune/dockerfile-sshd 3 days ago





037b956



22 of 109

Removed Passwords





Sources of leaks: Paste monitoring

- Example: <https://gist.github.com/>
 - Easily storing and sharing text online
 - Used by programmers and legitimate users
 - Source code & information about configurations

Sources of leaks: Paste monitoring

- Example: <https://gist.github.com/>
 - Easily storing and sharing text online
 - Used by programmers and legitimate users
 - Source code & information about configurations
- Abused by attackers to store:
 - List of vulnerable/compromised sites
 - Software vulnerabilities (e.g. exploits)
 - Database dumps
 - User data
 - Credentials
 - Credit card details
 - More and more ...

Examples of pastes (items)

The image displays three overlapping screenshots of text pastes, likely from a forum or chat application. Each paste is shown in a separate window with a title bar and a list of line numbers on the left.

Top-left paste (4.41 KB): The title bar indicates it is a 'text' file of 4.41 KB. The visible content is a C program snippet:

```
1. - - - - - Tool by Y3t1y3t ( u
2.
3.
4.
5.
6. 3. int zapisz (FILE *plik_
7. 4.     int i, j;
8. 5.     if (obr->KOLOR==0) {
9. 6.
10. 7.         fprintf (plik_wy, "P2
11. 8.         fprintf (plik_wy, "%d
12. 9.         fprintf (plik_wy, "%d
13. 10.        for (i=0; i<obr->wymy
14. 11.            for (j=0; j<obr->wymx; j++
15. 12.                fprintf (plik_wy, "%d ",
16. 13.            }
```

Top-right paste (2.82 KB): The title bar indicates it is a 'text' file of 2.82 KB. The visible content is a forum post title:

```
1. KillerGram - Yuffie - Smoke The Big Dick [smkwhr] (Upload
2.
3.
```

Bottom paste (4.57 KB): The title bar indicates it is a 'text' file of 4.57 KB. The visible content is an XML snippet:

```
1. #include "wejwyj.h"
2.
3.
4.
5.
6. 3. int zapisz (FILE *plik_
7. 4.     int i, j;
8. 5.     if (obr->KOLOR==0) {
9. 6.
10. 7.         fprintf (plik_wy, "P2
11. 8.         fprintf (plik_wy, "%d
12. 9.         fprintf (plik_wy, "%d
13. 10.        for (i=0; i<obr->wymy
14. 11.            for (j=0; j<obr->wymx; j++
15. 12.                fprintf (plik_wy, "%d ",
16. 13.            }
```

Bottom-right paste (2.66 KB): The title bar indicates it is a 'text' file of 2.66 KB. The visible content is an XML snippet:

```
1. <item name="%the_component_to_be_disabled%" xsi:type="array">
2.     <item name="config" xsi:type="array">
3.         <item name="componentDisabled" xsi:type="boolean">true</item>
4.     </item>
5. </item>
6.
7. <?xml version="1.0"?>
8.
9. <page xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespace
10. /etc/page_configuration.xsd">
11.     <body>
12.         <referenceBlock name="checkout.root">
13.             <arguments>
14.                 <argument name="jsLayout" xsi:type="array">
```

Why so many leaks?

- Economical interests (e.g. Adversaries promoting services)
- Ransom model (e.g. To publicly pressure the victims)
- Political motives (e.g. Adversaries showing off)
- Collaboration (e.g. Criminals need to collaborate)
- Operational infrastructure (e.g. malware exfiltrating information on a pastie website)
- Mistakes and errors

Are leaks frequent?

Yes!

and we have to deal with this as a CSIRT.

- **Contacting companies or organisations** who did specific accidental leaks
- **Discussing with media** about specific case of leaks and how to make it more practical/factual for everyone
- Evaluating the economical market for cyber criminals (e.g. DDoS booters⁵ or reselling personal information - reality versus media coverage)
- Analysing collateral effects of malware, software vulnerabilities or exfiltration

→ And it's important to detect them automatically.

⁵<https://github.com/D4-project/>

Paste monitoring at CIRCL: Statistics

- Monitored paste sites: 27
 - *gist.github.com*
 - *ideone.com*
 - ...

	2016	2017	08.2018
Collected pastes	18,565,124	19,145,300	11,591,987
Incidents	244	266	208

Table: Pastes collected and incident⁶ raised by CIRCL

⁶<http://www.circl.lu/pub/tr-46>

Current capabilities

AIL Framework: Current capabilities

- Extending AIL to add a new **analysis module** can be done in 50 lines of Python
- The framework **supports multi-processors/cores by default**. Any analysis module can be started multiple times to support faster processing during peak times or bulk import
- **Multiple** concurrent **data input**
- Tor Crawler (handle cookies authentication)

AIL Framework: Current features

- Extracting **credit cards numbers, credentials, phone numbers, ...**
- Extracting and validating potential **hostnames**
- Keeps track of **duplicates**
- Submission to threat sharing and incident response platform (**MISP** and **TheHive**)
- **Full-text indexer** to index unstructured information
- **Tagging** for classification and searches
- Terms, sets, regex and YARA **tracking and occurrences**
- Archives, files and raw **submission** from the UI
- PGP, Cryptocurrency, Decoded (Base64, ...) and username Correlation
- And many more

Terms Tracker

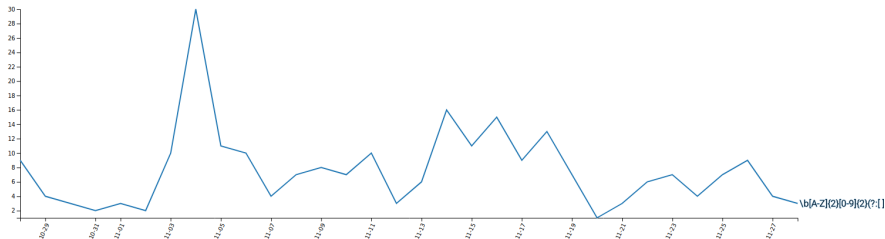
- Search and monitor specific keywords/patterns
 - Automatic Tagging
 - Email Notifications
- Track Term
 - ddos
- Track Set
 - booter,ddos,stresser;2
- Track Regex
 - circl\.lu
- YARA rules
 - <https://github.com/ail-project/ail-yara-rules>

Terms Tracker

82a87a6a-88f1-4ab1-ba53-1bf15211b4b8



Type	Tracker	Date added	Level	Created by	First seen	Last seen	Tags	Email
regex	\b[A-Z](2)[0-9](2)(?.[]?[0-9](4))(4)(?.[]?[0-9](3))(?.[]?[0-9](1,2))?.b	2019/09/12	1	admin@admin.test	2018/08/31	2019/11/28		






yyyy-mm-dd



yyyy-mm-dd

Search Tracked Items

YARA Tracker



Type	Tracker	Date added	Level	Created by	First seen	Last seen	Tags 	Email 	
yara	all-yara-rules/rules/code/vbscript.yar	2020/09/17	1	admin@admin.test	2020/09/17	2021/04/01			

rule test_vbscript

{
 meta:
 author = "kevthehermit"
 info = "Part of Pastehunter"
 reference = "https://github.com/kevthehermit/Pastehunter"

 strings:
 \$a = "function" nocase wide ascii fullword
 \$b = "createobject" nocase wide ascii fullword
 \$c = "wscript" nocase wide ascii fullword
 \$d = "as long" nocase wide ascii fullword
 \$e = "run" nocase wide ascii fullword
 \$f = "for each" nocase wide ascii fullword
 \$g = "end function" nocase wide ascii fullword
 \$h = "mtallocatevirtualmemory" nocase wide ascii fullword
 \$i = "mtwritevirtualmemory" nocase wide ascii fullword


 condition:
 5 of them
}


Edit Tracker  





Terms Tracker - Practical part

- **Create and test** your own tracker

 Tags (optional, space separated)

 E-Mails Notification (optional, space separated)

 Tracker Description (optional)

☒  Show tracker to all Users

– Select a tracker type –

 Add Tracker

Recon and intelligence gathering tools

- **Attacker also share informations**
- Recon tools detected: 94
 - sqlmap
 - dnscan
 - whois
 - msfconsole (metasploit)
 - dnmap
 - nmap
 - ...

Recon and intelligence gathering tools









```
#####
=====
Hostname      www.pabloquintanilla.cl      ISP      Wix.com Ltd.
Continent     North America              Flag
US
Country       United States              Country Code  US
Region Unknown              Local time   19 Nov 2019 07:59 CST
City Unknown              Postal Code  Unknown
IP Address    185.230.60.195              Latitude     37.751
                                   Longitude    -97.822
=====
#####
> www.pabloquintanilla.cl
Server:       38.132.106.139
Address:      38.132.106.139#53

Non-authoritative answer:
www.pabloquintanilla.cl canonical name = www192.wixdns.net.
www192.wixdns.net      canonical name = balancer.wixdns.net.
Name:   balancer.wixdns.net
Address: 185.230.60.211
>
#####
Domain name: pabloquintanilla.cl
Registrant name: SERGIO TORO
Registrant organisation:
Registrar name: NIC Chile
Registrar URL: https://www.nic.cl
```

Decoder

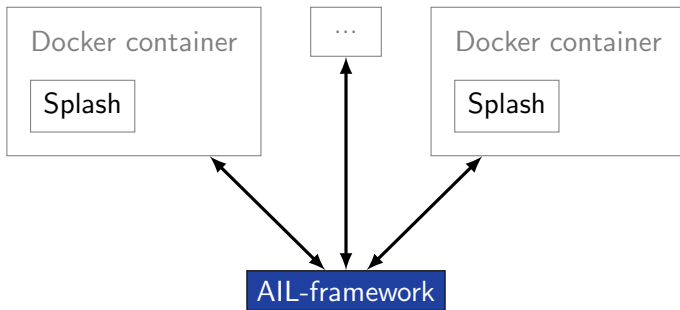
- Search for encoded strings
 - Base64
 - Hexadecimal
 - Binary
- Guess Mime-type
- Correlate paste with decoded items

Decoder:

estimated type	hash	first seen	last seen	nb item	size	Virus Total	Sparkline
application/x-dosexec	c11c2be8d9ba4e86c8effaa411aa6b867ba75abe	2019/11/28	2019/11/28	1	191	Send this file to VT 	
application/x-dosexec	a50cba731204ecce193b40178399a250b5ce6f67	2019/11/28	2019/11/28	1	32768	Send this file to VT 	
application/x-dosexec	cc5f2f0da71f443ec12ae1b3cb6ab8bad80f22c4	2019/11/28	2019/11/28	1	203	Send this file to VT 	
application/x-dosexec	eed67e8fa9cb9a43fea21ae653983a8e0a174f63	2019/11/26	2019/11/28	6	83	Send this file to VT 	

Crawler

- Crawlers are used to navigate on regular website as well as .onion addresses (via automatic extraction of urls or manual submission)
- Splash ("scriptable" browser) is rendering the pages (including javascript) and produce screenshots (HAR archive too)



Crawler

How a domain is crawled by default

1. Fetch the first url
2. Render javascript (webkit browser)
3. Extract all urls
4. Filter url: keep all url of this domain
5. crawl next url (max depth = 1)

Crawler: Cookiejar

Use your cookies to login and bypass captcha

Edit Cookiejar



Description	Date	UUID	User
3thxemke2x7hcibu.onion	2020/03/31	90674deb-38fb-4eba-a661-18899ccb3841	admin@admin.test

Edit Description

Add Cookies

```
{
  "domain": ".3thxemke2x7hcibu.onion",
  "name": "mybb[lastactive]",
  "path": "/forum/",
  "value": "1583829465"
}
```

```
{
  "domain": ".3thxemke2x7hcibu.onion",
  "name": "loginattempts",
  "path": "/forum/",
  "value": "1"
}
```

```
{
  "domain": ".3thxemke2x7hcibu.onion",
  "name": "sid",
  "path": "/forum/",
  "value": "847ab8cd97ff5bcc77eddb6a"
}
```

```
{
  "name": "remember_token",
  "value": "12158cddd151d74d341f23"
}
```

```
{
  "domain": ".3thxemke2x7hcibu.onion",
  "name": "mybb[announcements]",
  "path": "/forum/",
  "value": "0"
}
```

Crawler: Cookiejar

3thxemke2x7hcibu.onion :



First Seen Last Check Ports

2020/03/09 2020/03/30 [80]

infoleak:automatic-detection="onion"

infoleak:automatic-detection="base64"



manual

Show Domain Correlations 139

Add to MISP Export

Decoded 1

Screenshot 134

Crawled Items

Date: 2020/03/23 - 13:10:40 PORT: 80

Show 10 entries

Search:

Crawled Pastes



Shere Khan

Portal Search Member List Help

Welcome back, zuluport. You last visited: 03-20-2020, 01:35 PM Log Out

User: CP

View New Posts

View Today's Posts

Private Messages (Unread: 2, Total: 2)

You have 2 unread private messages. The most recent is from Jack3 (ID: KEY FOR PRIVATE SECTIONS)

Shere Khan - Official Forum

Private Messages

Home

User CP Home

Messages

Compose

Inbox

Send

Trash Can

Tracking

Build Folder

Your Profile

Get Profile

Change Password

Change Email

Change Avatar

Change Signature

Build Options

MicroNamex

Group Memberships

Buddy/Ignore List

Manage Attachments

Saved Drafts

Subscribed Threads

Forum Subscriptions

View Profile

Inbox | Compose Message | Manage Folders | Empty Folders | Download Messages 1% of PM space used.

Inbox Enter Keywords Search PMs (Advanced Search)

Message Title Sender Date/Time Sent (asc)

KEY FOR PRIVATE SECTIONS Jack3 3 hours ago

Verification Jack3 03-09-2020, 11:55 AM

Move To Inbox or Delete the selected messages

Jump to Folder: Inbox Go

Forum Team Contact Us Shere Khan - Hacking group Return to Top Lite (Archiva) Mode Mark all forums read RSS Syndication

Powered by MyBB, © 2002-2020 MyBB Group.

Current time: 03-23-2020, 01:33 PM

<http://3thxemke2x7hcibu.onion/forum/private.php>

Splash manager

- Splash manager⁷ is a standalone software⁸ to run the containers for crawling
- AIL crawlers are using Splash dockers to fetch and render domains
- Manage and control Splash Crawlers
 - Install and run Splash crawlers on another server,
 - Handle proxies (Web and Tor by default),
 - Launch/Kill Splash Dockers automatically,
 - Restart crawlers if these crash,

⁷<https://github.com/ail-project/ail-splash-manager>

⁸Crawlers can installed and executed outside AIL

Splash manager - UI in AIL

Splash Crawler Manager

Splash Manager URLhttps://test.cred.liu:7001

API Key*****

Edit

All Splash Crawlers:

Splash name	Proxy	Crawler type	Description
web_splash	web_proxy	web	web splash
default_splash_tor	default_tor	tor	default splash tor

All Proxies:

Proxy name	Host	Port	Type	Crawler Type	Description
web_proxy	172.17.0.1	2128	HTTP	web	web proxy
default_tor	172.17.0.1	9090	SOCKS5	tor	tor default proxy

Crawlers

THE CRAWLER TEST OUTPUT

It works!

Run Test

Number of Crawlers to Launch:


default_splash_tor	10
web_splash	1

Edit number of crawlers to launch

Relaunch Crawlers

44 of 109

Crawler: DDoS Booter



qy4n6ptiraa7mtfy73wcp6da2xrapmbanwfr5kei4zrq2va4uscvogid.onion :

First Seen	Last Check	Ports
2019/08/15	2019/10/06	[80]


infoleak:automatic-detection="bitcoin-address"

infoleak:automatic-detection="ethereum-address"



infoleak:automatic-detection="onion"




infoleak:automatic-detection="credit-card"


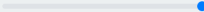

ddos



Last Origin: [crawled/2019/10/05/mqbyxj4ladgz5cd.onion0aa31681-fa45-4fc3-8151-7a7c5ac7e906](#)

 Show Domain Correlations 


 Cryptocurrencies  

[HOME](#) [ABOUT](#) [PROOF](#) [PRICE](#) [PAYMENT](#)

DDOSTECH

WICKR. DDOS. TECHNOLOGY








Reviews

April 25, 2019
I turned to this service on the recommendation of my friend, ordered an attack for a whole week, the work was done with high quality and responsibility.

September 21, 2018
I found this site through YAHOO, immediately contacted this service, and I had a free attack for almost ten minutes.

We accept:

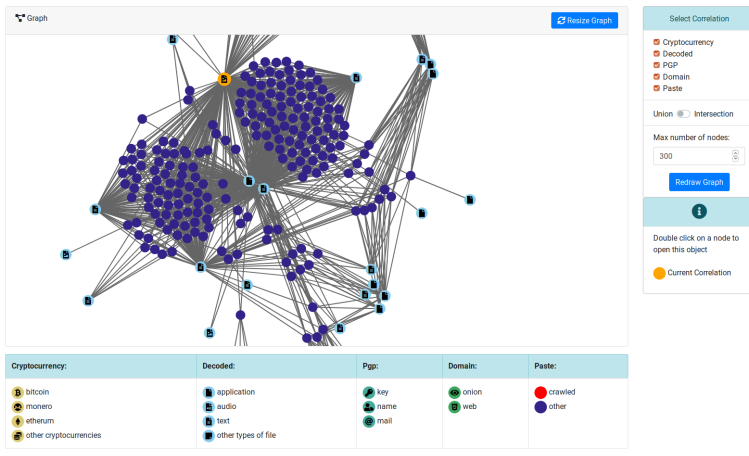
Accept payments cryptocurrency. Cryptocurrency transfers guarantee your our security transaction. We accept BTC, ETH, DASH, LTC, ETC, XMP ...



Wallets Addresses

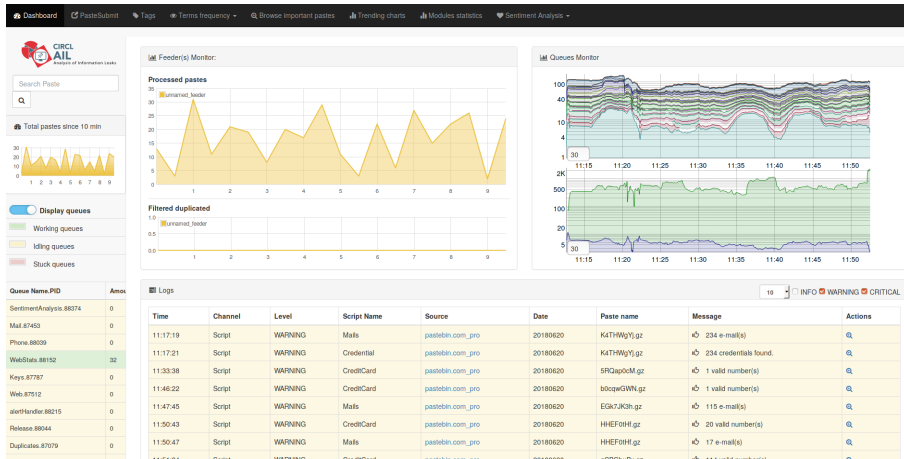
...

Correlations and relationship



Live demo!

Example: Dashboard



Example: Text search

Q 1 Results for "gandcrab"

Index: 2019-05-20 - 1365.328591 Mb

Show 10 entries **Search:**

#	Path	Date	Size (Kb)	Action
0	crawled/2019/05/17/vs5e7g245s3pxjoc.onion374a1a89-4b16-4c3f-a460-4be8898da140 crawled cve	2019/05/17	15.44	i Q

Showing 1 to 1 of 1 entries

Previous **1** Next

Totalling 1 results related to paste content









Example: Items Metadata (1)

infoleak:automatic-detection="phone-number"		infoleak:automatic-detection="mail"		infoleak:automatic-detection="base64"		+	
Date	Source	Encoding	Language	Size (Kb)	Mime	Number of lines	Max line length
04/05/2019	pastebin.com_pro	text/plain	None	6.12	text/plain	1650	100
Create  Event							

Duplicate list:

Show entries

Search:

Hash type	Paste info	Date	Path	Action
[tlsh]	Similarity: [19]%	2019-04-13	archive/pastebin.com_pro/2019/04/13/EbMVR87S.gz	
[tlsh]	Similarity: [10]%	2019-04-11	archive/pastebin.com_pro/2019/04/11/2X5HfVhX.gz	
[tlsh]	Similarity: [23]%	2019-04-25	archive/pastebin.com_pro/2019/04/25/TS2b6M4c.gz	
[tlsh]	Similarity: [14]%	2019-04-17	archive/pastebin.com_pro/2019/04/17/CuS93H7K.gz	
[tlsh]	Similarity: [23]%	2019-04-20	archive/pastebin.com_pro/2019/04/20/AQd0qGVQ.gz	
[tlsh]	Similarity: [20]%	2019-04-20	archive/pastebin.com_pro/2019/04/20/6DDc13b8.gz	
[tlsh]	Similarity: [21]%	2019-05-05	alerts/pastebin.com_pro/2019/05/05/X8nJLzda.gz	
[tlsh]	Similarity: [7]%	2019-04-13	archive/pastebin.com_pro/2019/04/13/Lyp4FVWW.gz	

Showing 1 to 8 of 8 entries





Previous **1** Next

Example: Items Metadata (2)

Hash files:

Show entries



Search:

estimated type	hash	saved_path	Virus Total
 application/octet-stream	3975f058bb0d445b60c10a11f1a5d88e19e4fa84 (1)	HASHS/application/octet-stream /39/3975f058bb0d445b60c10a11f1a5d88e19e4fa84	Send this file to VT 
 application/octet-stream	fed93c1753270fc849a4db37027b569cdd9a6108 (1)	HASHS/application/octet-stream /fe/fed93c1753270fc849a4db37027b569cdd9a6108	Send this file to VT 

Showing 1 to 2 of 2 entries

Previous **1** Next

Example: Items Metadata (3)


 Crawled Item 

Domain [2gtyctckj2y5e3ln.onion:80](#)


Father [crawled/2019/05/20/2gtyctckj2y5e3ln.onion954e1b05-aca-4586-a4bc-804bf27b54f7](#)

Url [http://2gtyctckj2y5e3ln.onion/index/forgot/password?tc=1](#)

Full resolution

 **Empire Market**

LOGIN REGISTER FORUMS VERIFY MIRROR

 **MNEMONIC VERIFICATION - PASSWORD/PIN RESET**

Please type your username and security mnemonic below that was provided to you at the time of registration.

Example: Browsing content

Content:

```
http://members2.mofosnetwork.com/access/login/  
somoextremos:buddy1990  
brazzers_glenn:cocklick  
brazzers61:braves01
```

```
http://members.naughtyamerica.com/index.php?m=login  
gernblanston:3unc2352  
Janhuss141200:310575  
igetalliwant:1377zeph  
pwilks89:mon22key  
Bman1551:hockey
```

```
MoFos IKnowThatGir1 PublicPickUps  
http://members2.mofos.com  
Chrismagg40884:loganm40  
brando1:zzbrando1  
aacoen:1q2w3e4r  
1rstunk1e23:my8self
```

```
BraZZers  
http://ma.brazzers.com  
gcjensen:gcj21pva  
skycsc17:rbcndnd
```

```
#####
```

```
>| Get Daily Update Fresh Porn Password Here |<
```

```
=> http://www.erq.io/4mF1
```

Example: Browsing content

Content:

```
Over 50000+ custom hacked xxx passwords by us! Thousands of free xxx passwords to the hottest paysites!

#####
>| Get Fresh New Premium XXX Site Password Here |<

=>  http://www.erq.io/4mF1

#####

http://ddfnetwork.com/home.html
eu172936:hCSBgKh
UecwB6zs:159X0$!r#6K78FuU

http://pornxn.stiffia.com/user/login
feldwWek8939:R0bluJ8XtB
dabudka:17891789
brajits:brajits1

http://members.pornstarplatinum.com/sblogin/login.php/
gigiriveracom:xxxjay
jayx123:xxxjay69

http://members.vividceleb.com/
Rufio99:fairhaven
ScHiFRvi:102091
Chaos84:HOLE5244
Riptor795:blade7
Domi80:harkonnen
GaggedUK:a1k0chan

http://www.ariellaferreira.com/
```

Example: Search by tags

Search Tags by date range :

2019-05-19

2019-05-21

infoleak:automatic-detection="cve" x infoleak:automatic-detection="bitcoin-address" x

Search Tags

Show

10

Search:

entries

Date	Path	# of lines	Action
2019/05/19	archive/pastebin.com_pro/2019/05/19/ej67tQ4b.gz cve bitcoin-address	71	
2019/05/21	archive/pastebin.com_pro/2019/05/21/vM2SwyTe.gz cve bitcoin-address	69	
2019/05/21	archive/pastebin.com_pro/2019/05/21/rsnHnp5L.gz cve bitcoin-address	71	

Showing 1 to 3 of 3 entries

Previous 1 Next

MISP

MISP Taxonomies

- **Tagging** is a simple way to attach a classification to an event or an attribute.
- **Classification must be globally used to be efficient.**
- Provide a set of already defined classifications modeling estimative language
- Taxonomies are implemented in a simple JSON format ⁹.
- Can be easily cherry-picked or extended

⁹<https://github.com/MISP/misp-taxonomies>

Taxonomies useful in AIL

- **infoleak**: Information classified as being potential leak.
- **estimative-language**: Describe quality and credibility of underlying sources, data, and methodologies.
- **admiralty-scale**: Rank the reliability of a source and the credibility of an information
- **fpf**¹⁰: Evaluate the degree of identifiability of personal data and the types of pseudonymous data, de-identified data and anonymous data.

¹⁰Future of Privacy Forum

Taxonomies useful in AIL

- **tor**: Describe Tor network infrastructure.
- **dark-web**: Criminal motivation on the dark web.
- **copine-scale**¹¹: Categorise the severity of images of child sex abuse.

¹¹Combating Paedophile Information Networks in Europe

threat sharing and incident response platforms



Goal: submission to threat sharing and incident response platforms.

threat sharing and incident response platforms




1. Use infoleak taxonomy¹²
2. Add your own tags
3. Export AIL objects to MISP core format
4. Download it or Create a MISP Event¹³

¹²<https://www.misp-project.org/taxonomies.html>

¹³<https://www.misp-standard.org/rfc/misp-standard-core.txt>

MISP Export

1Gt545E48EPsyTC8voKQDCFpTkwiuXduw :

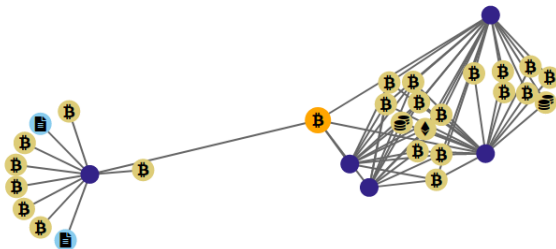
Object type	type	First seen	Last seen	Nb seen
cryptocurrency	 bitcoin	2020/01/17	2020/02/20	5

Expand Bitcoin address

Graph

Resize Graph

Add to  Export



MISP Export

nttfj36sp47cw2yecop572zjvjeazgazeunllouudplzqt2m
5h465yd.onion :



First Seen	Last Check	Ports
------------	------------	-------

2020/02/19	2020/02/19	['80']
------------	------------	--------

infoleak:automatic-detection="onion"



Last Origin: [crawled/2020/02/19/dark.failc126d32a-3ed1-468f-ba24-f2e5956f4035](#)

Show Domain Correlations **4**

Add to Export

Hide

Empire Market

[LOGIN](#)

[REGISTER](#)

[FORUMS](#)

[VE](#)

Login

LOGIN TO EMPIRE MARI

Welcome to Empire Market! Please log
Registrations are free and open to every


Username

Password







What's th

Login

MISP Export

 MISP Exporter

Select a list of objects to export

Object Type	Object ID	Lvl	
Object type... ▾		0	 
Object type... ▾	1Gt545E48EPsyTC8voKQDCfpTkwiuXduw	✓ 1	 
Domain ▾	nttfj36sp47cw2yecop572zjvjeazgazieunllouudplzqt2m5h465yd.onion	✓ 0	 

JSON Export ☒ Export to MISP Instance

Distribution:

Threat Level:

Analysis:

Event Info:


Publish Event ☐

Export Objects

Automatic submission on tags

MISP Auto Event Creation

Enabled



✕ Disable Event Creation

The hive auto export

Disabled



☑ Enable Alert Creation

Metadata : 6 / 25

Show 5 entries Search:

Whitelist	Tag
<input checked="" type="checkbox"/>	infoleak:automatic-detection="api-key"
<input checked="" type="checkbox"/>	infoleak:automatic-detection="aws-key"
<input checked="" type="checkbox"/>	infoleak:automatic-detection="base64"
<input type="checkbox"/>	infoleak:automatic-detection="bitcoin-address"
<input type="checkbox"/>	infoleak:automatic-detection="bitcoin-private-key"

Showing 1 to 5 of 25 entries

Previous 1 2 3 4 5

Next

Metadata : 23 / 25

Show 5 entries Search:

Whitelist	Tag
<input checked="" type="checkbox"/>	infoleak:automatic-detection="api-key"
<input checked="" type="checkbox"/>	infoleak:automatic-detection="aws-key"
<input checked="" type="checkbox"/>	infoleak:automatic-detection="base64"
<input checked="" type="checkbox"/>	infoleak:automatic-detection="bitcoin-address"
<input checked="" type="checkbox"/>	infoleak:automatic-detection="bitcoin-private-key"

Showing 1 to 5 of 25 entries

Previous 1 2 3 4 5

Next

API

AIL exposes a ReST API which can be used to interact with the back-end¹⁴.

```
1 curl https://127.0.0.1:7000/api/v1/get/item/default
2     --header "Authorization:
3     iHc1_ChZxj1aXmiFiF1mkxxQkzawwriEaZpPqyTQj "
4     -H "Content-Type: application/json"
5     --data @input.json -X POST
```

- AIL API is currently covering 60% of the functionality of back-end.

¹⁴[https:](https://github.com/ail-project/ail-framework/blob/master/doc/README.md)

[//github.com/ail-project/ail-framework/blob/master/doc/README.md](https://github.com/ail-project/ail-framework/blob/master/doc/README.md)

Setting up the framework

Setting up AIL-Framework from source

Setting up AIL-Framework from source

```
1 git clone  
   https://github.com/ail-project/ail-framework.git  
2 cd AIL-framework  
3 ./installing_deps.sh
```

Feeding the framework

Feeding AIL

There are different way to feed AIL with data:

1. Setup *pystemon* and use the custom feeder
 - *pystemon* will collect items for you
2. Use the new JSON Feeder (twitter)
3. Feed your own data using the API or the `import_dir.py` script
4. Feed your own file/text using the UI (Submit section)

Via the UI (1)

Files submission

Submit a file

Browse...

No file selected.

Archive Password

Optional

Tags :

Select Tags

Taxonomie Selection ▼

Select Tags

Galaxy Selection ▼

Submit this paste

Via the UI (2)


Submitting Pastes ...

100 %

Files Submitted 1/1

Submitted pastes

/home/all/git/AIL.framework/PASTES/submitted/2018/06/29/02071570-b464-4bbb-be59-37c58c9b8925.gz

Submitted Pastes 

Success ✓

Feeding AIL with your own data - API

api/v1/import/item

```
1 {  
2   "type": "text",  
3   "tags": [  
4     "infoleak:analyst-detection=\"private-key\""  
5   ],  
6   "text": "text to import"  
7 }
```

External feeders for AIL framework

- **12+ feeders are available** for all AIL users to feed from external sources
- External feeders can run anywhere and are completely separated from AIL framework
- The feeder can use their **own internal logic** and even JSON metadata
- Feeder are then pushing the generated JSON to AIL API

Feeding AIL with Twitter posts and associated urls

- AIL - feeder from Twitter¹⁵
- The AIL-feeder-twitter **search specific keywords on Twitter** using Twint (without API), crawls the urls and pushes the results in AIL
- The JSON format format can be extended via meta fields

¹⁵<https://github.com/ail-project/ail-feeder-twitter>

Certificate transparency feeder for AIL

- ail-feeder-cti¹⁶ is a generic software to extract information from a certstream server (certificate transparency)
- All metadata extracted will be processed by AIL
- Onion addresses crawled automatically by AIL if seen in a certificate

¹⁶<https://github.com/ail-project/ail-feeder-ct>

GitHub archive and GitHub repository

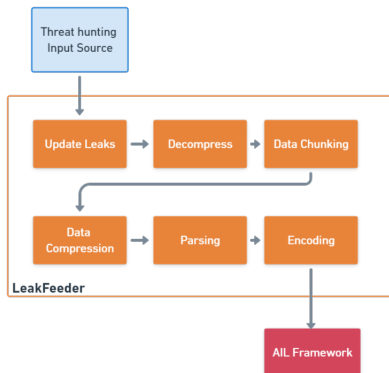
- ail-feeder-gharchive¹⁷ is a generic software to extract informations from **GHArchive**, collect and feed AIL via AIL ReST API
- ail-feeder-github-repo¹⁸ is collecting from a GitHub repository and push everything to AIL
- For monitoring a set of **suspicious git repositories** or finding leaks on existing or managed git repositories, it's a simple way to feed AIL with such source.

¹⁷<https://github.com/ail-project/ail-feeder-gharchive>

¹⁸<https://github.com/ail-project/ail-feeder-github-repo>

AIL LeakFeeder

- ail-feeder-leak¹⁹ automates the process to feed leaked large files automatically to AIL



¹⁹<https://github.com/ail-project/ail-feeder-leak>

AIL feeder ActivityPub

- ail-feeder-activity-pub²⁰ is feeder for the ActivityPub standard used in distributed social networks
- Accounts are required on the ActivityPub instance to get the stream

²⁰<https://github.com/ail-project/ail-feeder-activity-pub>

AIL feeder telegram

- ail-feeder-telegram²¹ is a **Telegram feeder**
- An API ID/hash for Telegram is required and linked to your Telegram phone number

²¹<https://github.com/ail-project/ail-feeder-telegram>

More feeders

- ail-feeder-discord²² is a generic **Discord** feeder for AIL
- ail-feeder-atom-rss²³ is an **Atom and RSS reader** and feeder for AIL
- ail-feeder-jsonlogs²⁴ is a **JSON aggregator** to submit generic JSON input into AIL

²²<https://github.com/ail-project/ail-feeder-discord>

²³<https://github.com/ail-project/ail-feeder-atom-rss>

²⁴<https://github.com/ail-project/ail-feeder-jsonlogs>

Feeding AIL with custom JSON

- AIL - feeder from Twitter²⁵
- The AIL-feeder-twitter search in Twitter using Twint (without API), crawls the urls and pushes the results in AIL
- The JSON format format can be extended via meta fields

²⁵<https://github.com/ail-project/ail-feeder-twitter>

Feeding AIL with custom JSON



conti leaks
@ContiLeaks

...

conti jabber leaks anonfiles.com/VeP6K6K5xc/1_t...

9:22 PM · 27 févr. 2022 · Twitter Web App

123 Retweets **23** Tweets cités **297** J'aime

```
{
  "ts": "2020-09-08T00:28:49.471678",
  "from": "ceram@q3mcco35auwcstmt.onion",
  "to": "stern@q3mcco35auwcstmt.onion",
  "body": "Проинструктируйте меня. Что делать?"
}
```

Feeding AIL with Conti leaks

- Conti jabber leaks are a good candidate for AIL analysis:
 - PGP keys
 - Bitcoin addresses, maybe others,
 - onion hidden services
- first we translated the files on english using deepl.com
- Support added in AIL to interpret jabber ids as usernames,
- then we created a feeder to import json data in AIL.

Feeding AIL with Conti leaks

```
from pyail import PyAIL
#... imports
#... setup code
for content in sys.stdin:
    elm = json.loads(content)
    tmp = elm['body']
    tmpmt = {}
    tmpmt['jabber:to'] = elm['to']
    tmpmt['jabber:from'] = elm['from']
    tmpmt['jabber:ts'] = elm['ts']
    tmpmt['jabber:id'] = "{}".format(uuid.uuid4())
    pyail.feed_json_item(tmp, tmpmt, ailfeedertype,
        source_uuid)
```

feeder.py

```
$ cat ~/conti/* | jq . -c | python ./feeder.py
```

Feeding AIL with Conti leaks

- use grep to limit the noise on an instance by only sending interesting bits:
 - PGP keys

```
$ cat ~/conti/* | jq . -c | grep PGP | python ./feeder.py
```

- onion hidden services | grep http:// |
- telegram addresses | grep tg:// |
- bitcoins addresses | egrep
--regexp="[13][a-km-zA-HJ-NP-Z1-9]25,34" |

Feeding ALL with your own data - import_dir.py (1)

/!\ requirements:

- Each file to be fed must be of a reasonable size:
 - ~ 3 Mb / file is already large
 - This is because some modules are doing regex matching
 - If you want to feed a large file, better split it in multiple ones

Feeding ALL with your own data - import_dir.py (2)

1. Check your local configuration `configs/core.cfg`
 - In the file `configs/core.cfg`,
 - Add `127.0.0.1:5556` in `ZMQ_Global`
 - (should already be set by default)
2. Launch `import_dir.py` with the directory you want to import
 - `import_dir.py -d dir_path`

Starting the framework

Running your own instance from source

Accessing the environment and starting AIL

```
1  
2 # Launch the system and the web interface  
3 cd bin/  
4 ./LAUNCH -l
```

Updating AIL

Launch the updater:

```
1 cd bin/
2 # git pull and launch all updates:
3 ./LAUNCH -u
4
5
6 # PS:
7 # The Updater is launched by default each time
8 # you start the framework with
9 # ./LAUNCH -l
```

AIL ecosystem - Challenges and design

ALL ecosystem: Technologies used

Programming language: Full python3

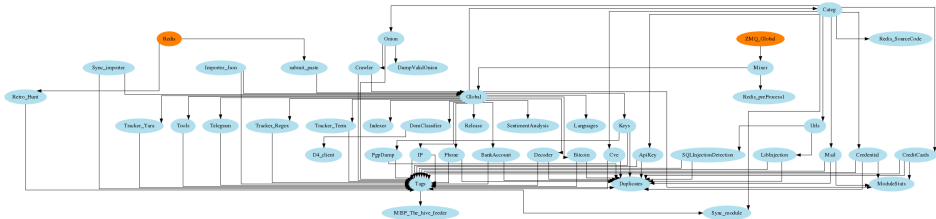
Databases: Redis and ARDB²⁶

Server: Flask

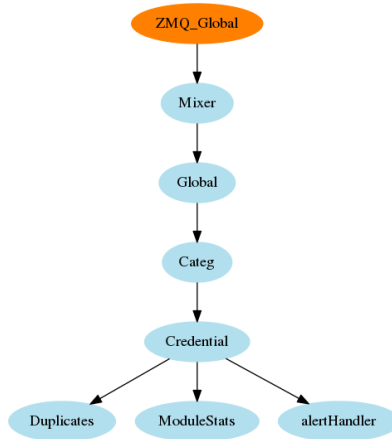
Data message passing: ZMQ, Redis list and Redis
Publisher/Subscriber

²⁶We are migrating to kvrocks

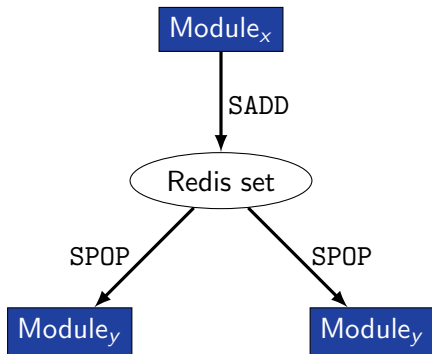
AIL global architecture: Data streaming between module



AIL global architecture: Data streaming between module (Credential example)



Message consuming

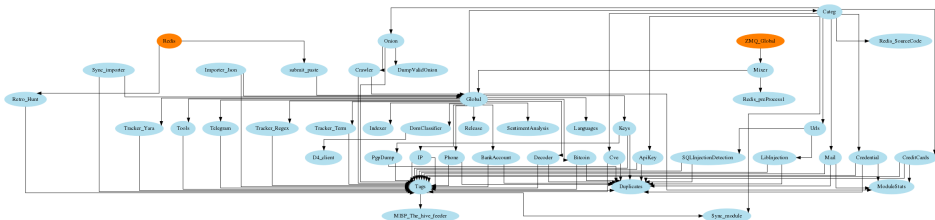


- No message lost nor double processing
- Multiprocessing!

Creating new features

Developing new features: Plug-in a module in the system

Choose where to put your module in the data flow:



Then, modify `bin/package/modules.cfg` accordingly

Writing your own modules - /bin/template.py

```
1 from modules.abstract_module import AbstractModule
2
3 class NewModule(AbstractModule):
4
5     def __init__(self):
6         super(NewModule, self).__init__()
7
8     # Do something with the message from the queue
9     def compute(self, message, r_result=False):
10         # Process Message
11
12 # LAUNCH MODULE
13 if __name__ == '__main__':
14     module = NewModule()
15     module.run()
16
17
```

Contribution rules

How to contribute



Glimpse of contributed features

- Docker
- Ansible
- Email alerting
- SQL injection detection
- Phone number detection

How to contribute

- Feel free to fork the code, play with it, make some patches or add additional analysis modules.

How to contribute

- Feel free to fork the code, play with it, make some patches or add additional analysis modules.
- Feel free to make a pull request for your contribution

How to contribute

- Feel free to fork the code, play with it, make some patches or add additional analysis modules.
- Feel free to make a pull request for your contribution
- That's it!

< (^ . ^)

Final words

- Building AIL helped us to find additional leaks which cannot be found using manual analysis and **improve the time to detect duplicate/recycled leaks**.

→ Therefore quicker response time to assist and/or inform proactively affected constituents.

Implementation Steps in AIL project

- **Gradual changes** in AIL to add required functionalities to support the objectives.
- **Time-memory trade-off** can be challenging to ensure a functional framework.
- Evaluation and integration of new modules in AIL based on time-memory comparisons.
- Semantic aspects are challenging due to the diverse data sources, unstructured data and languages seen.

Ongoing developments

- New JSON feeders
- Data retention and lifetime management of objects
- MISP modules expansion
- Extension of the tracker with typo-squatting library
- Auto classification of content by set of terms (semantic analysis)
- Improved export stream to third parties software
- Improved indexing relying on Solr, Lucene or other components

Annexes

Managing AIL: Old fashion way

Access the script screen

```
1 screen -r Script
```

Table: GNU screen shortcuts

Shortcut	Action
C-a d	detach screen
C-a c	Create new window
C-a n	next window screen
C-a p	previous window screen