AIL Project

Open source framework to efficiently collect, crawl, dig, and analyze unstructured data

CIRCL
Computer Incident
Response Center
Luxembourg

Aurelien Thirion
aurelien.thirion@circl.lu

info@circl.lu

October 4, 2023

AIL Project

Open source framework to efficiently collect, crawl, dig, and analyze unstructured data

CIRCL
Computer Incident
Response Center
Luxembourg

~~Aurelien Thirion~~
~~aurelien.thirion@circl.lu~~

info@circl.lu

October 4, 2023

Links

- AIL project `https://github.com/ail-project` (**all components including feeders and crawler infrastructure**)
- AIL framework `https://github.com/ail-project/ail-framework` (**analysis framework**)
- Training materials and slide deck `https://github.com/ail-project/ail-training`
- Online chat `https://gitter.im/ail-project/community`

Legal and Ethics

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 professionaly**[1].

[1] `https://www.acm.org/code-of-ethics` `https://www.first.org/global/sigs/ethics/ethics-first`

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 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,
  - OpenGPG 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

Session Objectives

- Demonstrate the practical usage and extensibility of an open source tool for monitoring web pages, pastes, forums, and hidden services
- Discuss the challenges involved and delve into the design principles of the AIL open source framework
- Explore various **collection mechanisms and sources utilized** by the AIL framework
- Gain knowledge on creating new modules within the AIL framework
- Acquire (quickly) proficiency in using, installing, and initializing AIL
- Understand the significance of integrating the AIL framework into the cyber threat intelligence life cycle, with notable tools such as MISP

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.
- 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 became a complete project called **ail project**[2].
- In 2023, AIL framework version 5.0 released with a new datastorage back-end.
- In 2023, AIL framework version 5.5 released with a new IM crawl functionality.

[2] `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 large leak archive
- **Monitor** and crawl websites

Supporting CERT and Law Enforcement Activities

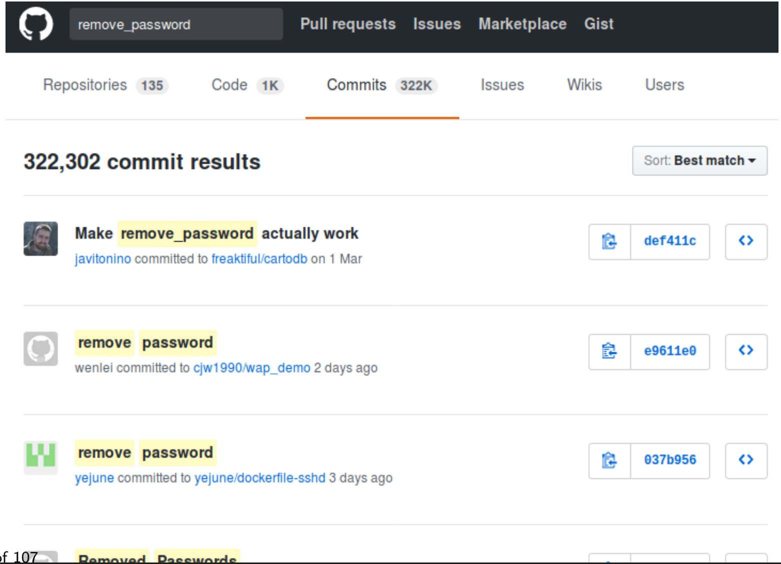
- Proactive Investigation: Detection of Leaks
  - Compilation of leaked emails and passwords
  - Analysis of leaked databases
  - Identification of exposed SaaS keys (AWS, Google,...)
  - Detection of compromised credit card information
  - Identification and analysis of compromised PGP private keys and certificate keys
- Contributing to Passive DNS and Metadata Collection Systems
- Sharing CVEs and Proof-of-Concepts (PoCs) for commonly exploited vulnerabilities
- Deanonymization of Hidden Services

Support CERT and Law Enforcement activities

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

Sources of leaks

Catching mistakes from users



322,302 commit results

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

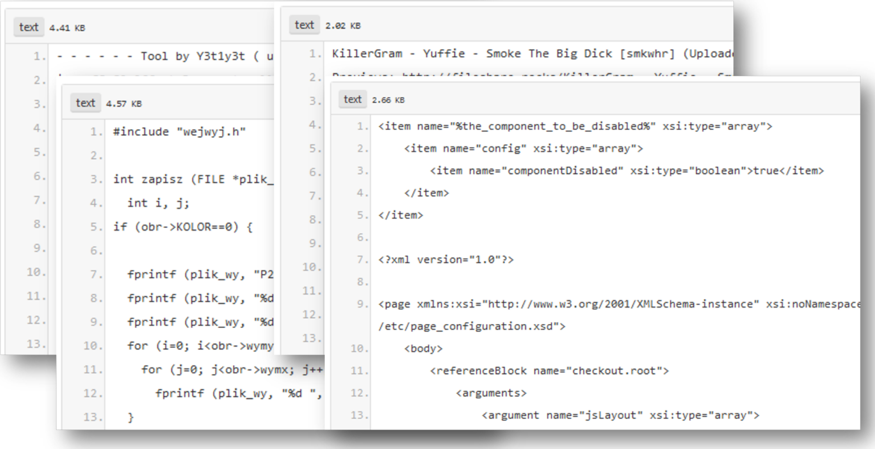
remove password
yejune committed to yejune/dockerfile-sshd 3 days ago
037b956

Removed Passwords

Example - 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)



text 4.41 KB

```
- - - - - - Tool by Y3t1y3t ( u
```

text 4.57 KB

```
#include "wejwyj.h"

int zapisz (FILE *plik_
  int i, j;
if (obr->KOLOR==0) {

  fprintf (plik_wy, "P2
  fprintf (plik_wy, "%d
  fprintf (plik_wy, "%d
  for (i=0; i<obr->wymy
    for (j=0; j<obr->wymx; j++
      fprintf (plik_wy, "%d ",
  }
```

text 2.02 KB

```
KillerGram - Yuffie - Smoke The Big Dick [smkwhr] (Upload
```

text 2.66 KB

```
<item name="%the_component_to_be_disabled%" xsi:type="array">
    <item name="config" xsi:type="array">
        <item name="componentDisabled" xsi:type="boolean">true</item>
    </item>
</item>

<?xml version="1.0"?>

<page xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespace
/etc/page_configuration.xsd">
    <body>
        <referenceBlock name="checkout.root">
            <arguments>
                <argument name="jsLayout" xsi:type="array">
```

Purposes of Leaks

- **Economic Interests**: Adversaries may promote services for their own financial gain.
- **Ransom Model**: Leaks can be used to publicly pressure victims into meeting certain demands.
- **Political Motives**: Adversaries may leak information to showcase their power or influence.
- **Collaboration**: Criminals may need to collaborate and share leaked information for their operations.
- **Operational Infrastructure**: Examples include malware that exfiltrates information to pastie websites.
- **Mistakes and Errors**: Leaks can also occur due to unintentional mistakes or errors.

Objectives for SOC/CSIRT Teams

- **Contacting Companies or Organizations**: Reach out to companies or organizations responsible for specific accidental leaks to address the issue
- **Engaging with Media**: Collaborate with the media to discuss specific leak cases and find practical ways to increase factual information available to the public
- Evaluate the Cybercriminal Economy: Analyze the cybercriminal market, including activities such as DDoS booters[3] and the reselling of personal information, in order to understand the disparity between reality and media coverage
- Analyze the Collateral Effects: Investigate the broader impact of malware, software vulnerabilities, or data exfiltration incidents

[3] `https://github.com/D4-project/`

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**
- Automatic Tor Crawler and website crawling (handle cookies authentication) via Lacus[4]

[4] `https://github.com/ail-project/lacus`

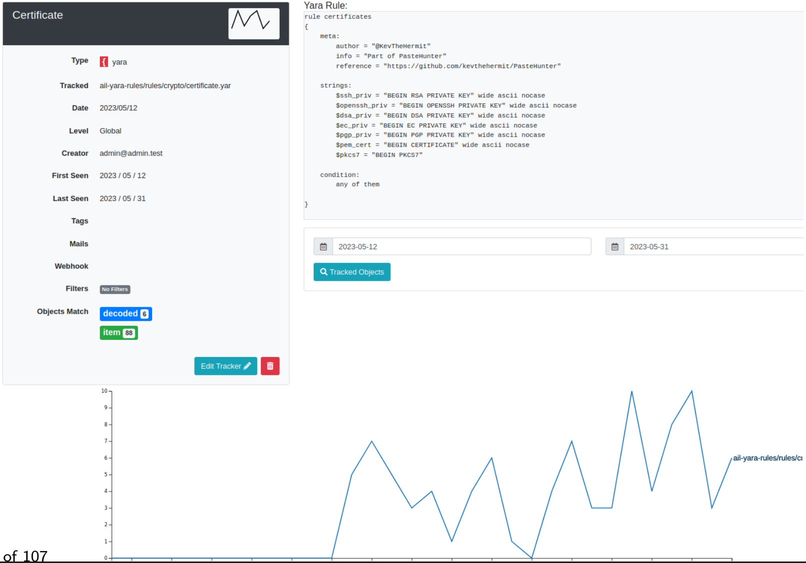
AIL Framework - features

- 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, Cryptocurrencies, Decoded (Base64, ...) and username Correlation
- And many more

Trackers - Retro Hunt

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

YARA Tracker



Certificate

Type	yara
Tracked	ail-yara-rules/rules/crypto/certificate.yar
Date	2023/05/12
Level	Global
Creator	admin@admin.test
First Seen	2023 / 05 / 12
Last Seen	2023 / 05 / 31
Tags	
Mails	
Webhook	
Filters	No Filters
Objects Match	decoded 6 item 88

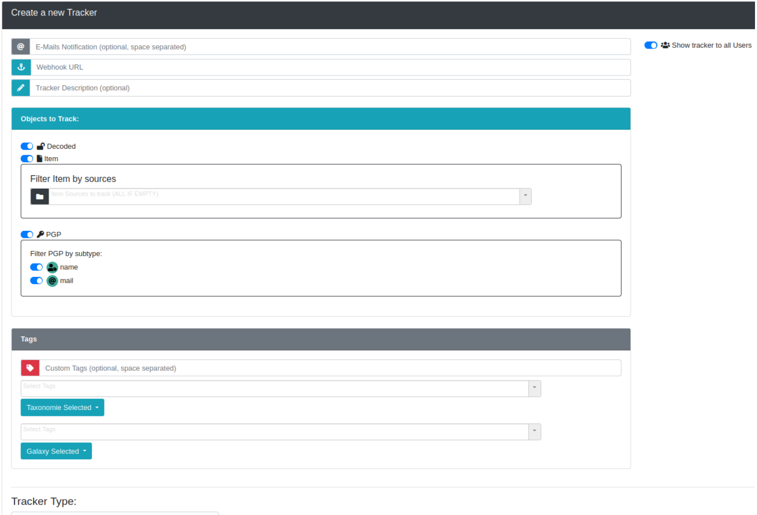
Yara Rule:

```
rule certificates
{
    meta:
        author = "@KevTheHermit"
        info = "Part of PasteHunter"
        reference = "https://github.com/kevthehermit/PasteHunter"

    strings:
        $ssh_priv = "BEGIN RSA PRIVATE KEY" wide ascii nocase
        $openssh_priv = "BEGIN OPENSSH PRIVATE KEY" wide ascii nocase
        $dsa_priv = "BEGIN DSA PRIVATE KEY" wide ascii nocase
        $ec_priv = "BEGIN EC PRIVATE KEY" wide ascii nocase
        $pgp_priv = "BEGIN PGP PRIVATE KEY" wide ascii nocase
        $pem_cert = "BEGIN CERTIFICATE" wide ascii nocase
        $pkcs7 = "BEGIN PKCS7"

    condition:
        any of them

}
```

2023-05-12 2023-05-31

Trackers - Practical part

- **Create and test** your own tracker



Create a new Tracker

E-Mails Notification (optional, space separated)

Webhook URL

Tracker Description (optional)

Show tracker to all Users

Objects to Track:

Decoded

Item

Filter Item by sources

Item Sources to track (ALL IF EMPTY)

PGP

Filter PGP by subtype:

name

mail

Tags

Custom Tags (optional, space separated)

Select Tags

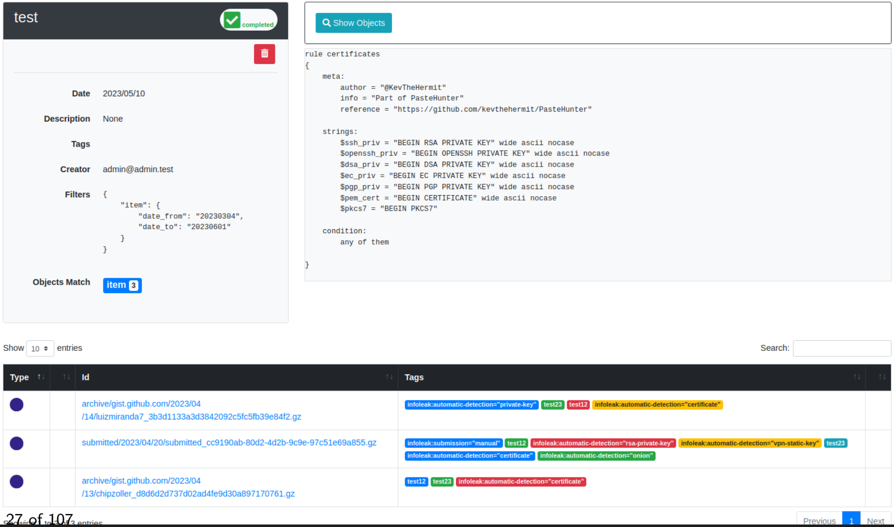
Taxonomie Selected

Select Tags

Galaxy Selected

Tracker Type:

Retro Hunt



test

completed

Date 2023/05/10

Description None

Tags

Creator admin@admin.test

Filters

```
{
    "item": {
        "date_from": "20230304",
        "date_to": "20230601"
    }
}
```

Objects Match item 3

Show Objects

```
rule certificates
{
    meta:
        author = "@KevTheHermit"
        info = "Part of PasteHunter"
        reference = "https://github.com/kevthehermit/PasteHunter"

    strings:
        $ssh_priv = "BEGIN RSA PRIVATE KEY" wide ascii nocase
        $openssh_priv = "BEGIN OPENSSH PRIVATE KEY" wide ascii nocase
        $dsa_priv = "BEGIN DSA PRIVATE KEY" wide ascii nocase
        $ec_priv = "BEGIN EC PRIVATE KEY" wide ascii nocase
        $pgp_priv = "BEGIN PGP PRIVATE KEY" wide ascii nocase
        $pem_cert = "BEGIN CERTIFICATE" wide ascii nocase
        $pkcs7 = "BEGIN PKCS7"

    condition:
        any of them

}
```

Show 10 entries

Search:

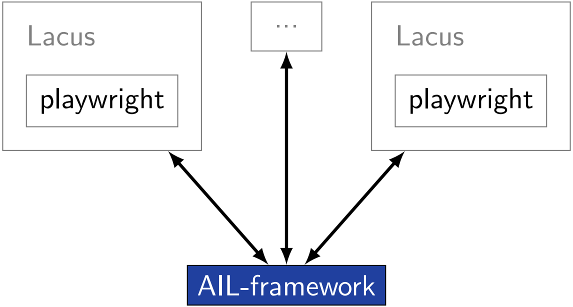
Type		Id	Tags	
●		archive/gist.github.com/2023/04 /14/luizmiranda7_3b3d1133a3d3842092c5fc5fb39e84f2.gz	infoleak:automatic-detection="private-key" test23 test12 infoleak:automatic-detection="certificate"	
●		submitted/2023/04/20/submitted_cc9190ab-80d2-4d2b-9c9e-97c51e69a855.gz	infoleak:submission="manual" test12 infoleak:automatic-detection="rsa-private-key" infoleak:automatic-detection="vpn-static-key" test23 infoleak:automatic-detection="certificate" infoleak:automatic-detection="onion"	
●		archive/gist.github.com/2023/04 /13/chipzoller_d8d6d2d737d02ad4fe9d30a897170761.gz	test12 test23 infoleak:automatic-detection="certificate"	

Showing 1 to 3 of 3 entries

Previous 1 Next

Crawler

- Crawlers are used to navigate on regular website as well as .onion addresses (via automatic extraction of urls or manual submission)
- Lacus[5] ("scriptable" browser) is rending the pages (including javascript) and produce screenshots (HAR archive too)



[5] https://github.com/ail-project/lacus

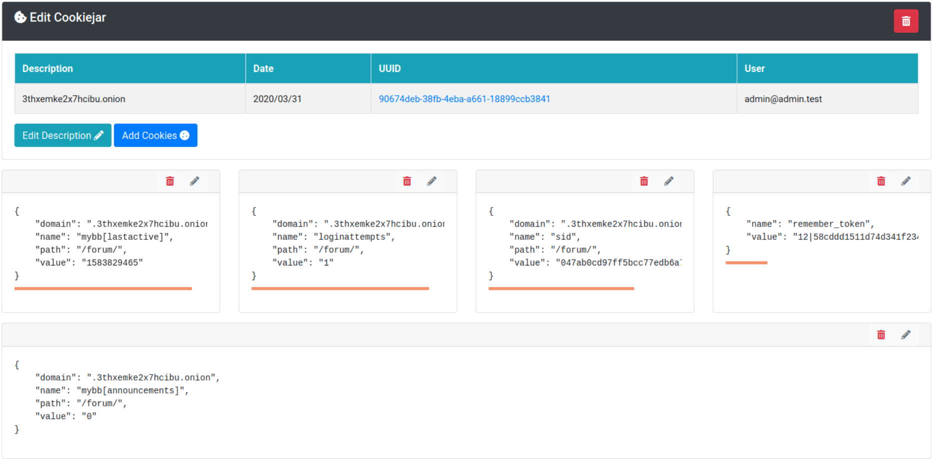
Auto Crawler

How a domain is crawled by default

1. Fetch the first url
2. Render the **web page including javascript** (done by playwright via Lacus)
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

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

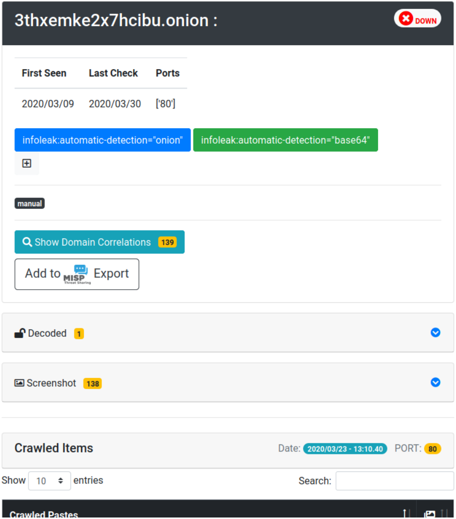
```
{
    "domain": ".3thxemke2x7hcibu.onio
    "name": "loginattempts",
    "path": "/forum/",
    "value": "1"
}
```

```
{
    "domain": ".3thxemke2x7hcibu.onio
    "name": "sid",
    "path": "/forum/",
    "value": "047ab0cd97ff5bcc77edb6a
}
```

```
{
    "name": "remember_token",
    "value": "12|58cddd1511d74d341f23
}
```

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

Crawler: Cookiejar



3thxemke2x7hcibu.onion :

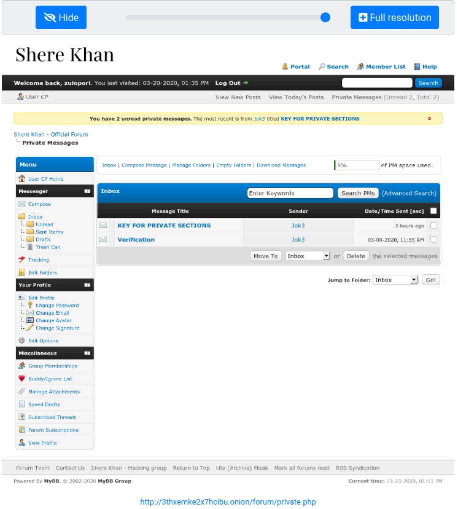
First Seen	Last Check	Ports
2020/03/09	2020/03/30	['80']

infoleak:automatic-detection="onion" infoleak:automatic-detection="base64"

manual

Crawled Items Date: 2020/03/23 - 13:10.40 PORT: 80

Show 10 entries Search:



Shere Khan

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

Private Messages (Unread 2, Total 2)

You have 2 unread private messages. The most recent is from Jok3 titled KEY FOR PRIVATE SECTIONS

Shere Khan - Official Forum
Private Messages

1% of PM space used.

Message Title	Sender	Date/Time Sent [asc]
KEY FOR PRIVATE SECTIONS	Jok3	3 hours ago
Verification	Jok3	03-09-2020, 11:55 AM

Powered By MyBB, © 2002-2020 MyBB Group. Current time: 03-23-2020, 01:11 PM

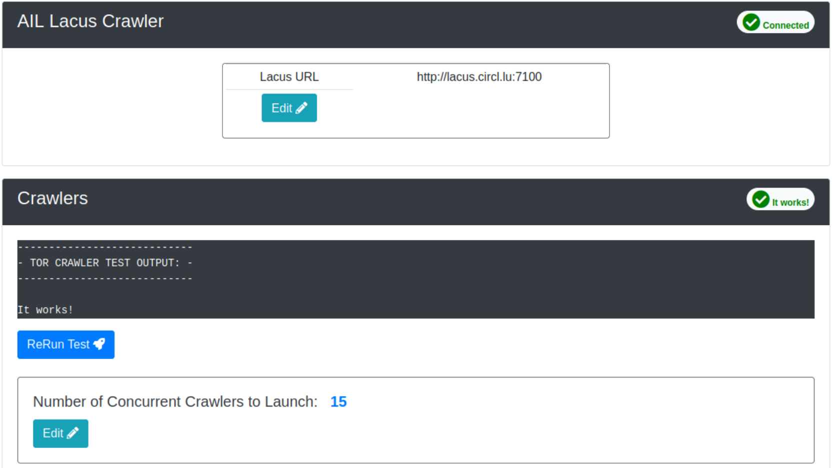
http://3thxemke2x7hcibu.onion/forum/private.php

Lacus: Web Capturing System

- Lacus[6] is a web capturing system built on playwright.
- AIL utilizes Lacus for fetching and rendering domains.
  - Lacus can be installed and used independently from AIL.
  - Capture what you need by enqueuing requests.
  - Initiate the capture process.
  - Retrieve the capture results.

[6] `https://github.com/ail-project/lacus`

Crawler Settings - Lacus



AIL Lacus Crawler

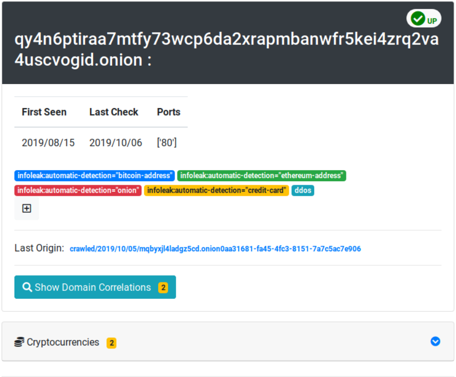
Lacus URL	http://lacus.circl.lu:7100

Crawlers

```
---------------------------
- TOR CRAWLER TEST OUTPUT: -
---------------------------

It works!
```

Number of Concurrent Crawlers to Launch: **15**

Crawler: DDoS Booter



UP

qy4n6ptiraa7mtfy73wcp6da2xrapmbanwfr5kei4zrq2va4uscvogid.onion :

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

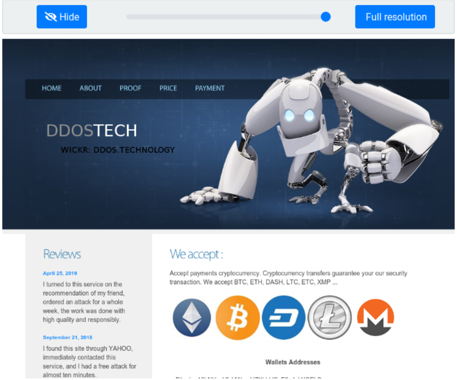
infoleak:automatic-detection="bitcoin-address" infoleak:automatic-detection="ethereum-address" infoleak:automatic-detection="onion" infoleak:automatic-detection="credit-card" ddos

⊞

Last Origin: crawled/2019/10/05/mqbyxjl4ladgz5cd.onion0aa31681-fa45-4fc3-8151-7a7c5ac7e906

Show Domain Correlations 2

Cryptocurrencies 2



Hide

Full resolution

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 responsibly.

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 :

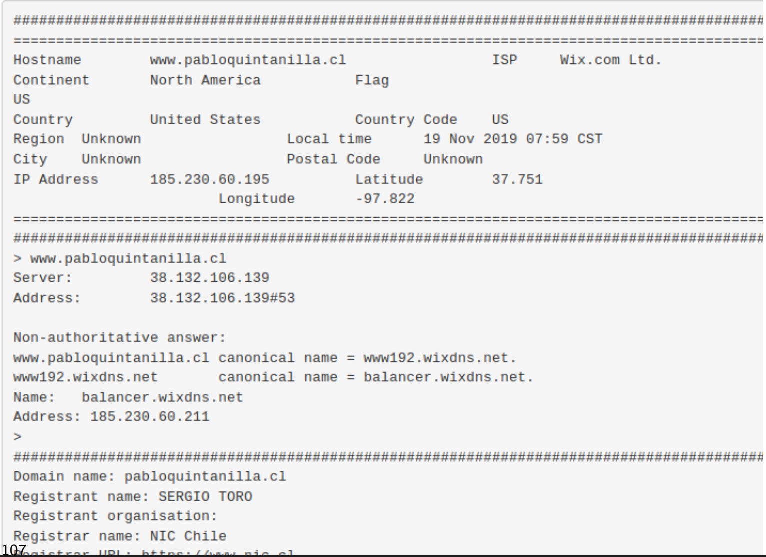
Accept payments cryptocurrency. Cryptocurrency transfers guarantee your our security transaction. We accept BTC, ETH, DASH, LTC, ETC, XMP ...

Wallets Addresses

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

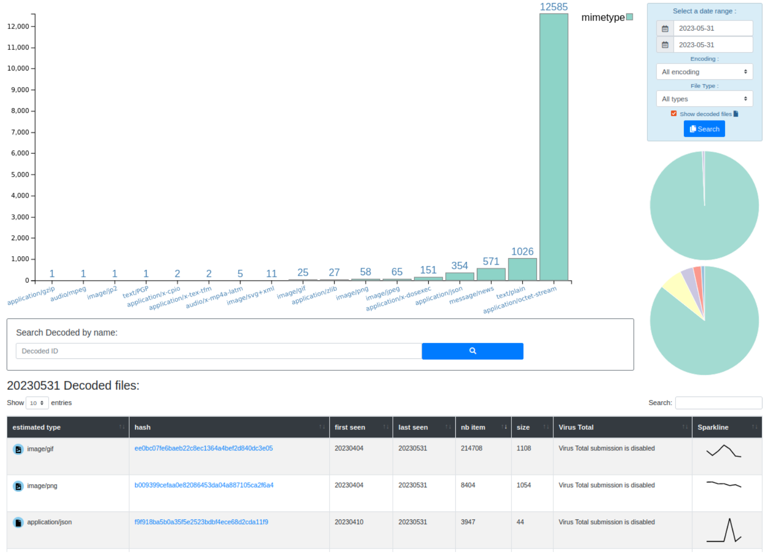


```
########################################################################################
========================================================================================
Hostname        www.pabloquintanilla.cl                 ISP     Wix.com Ltd.
Continent       North America           Flag
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
- Items/Domains Correlation

Decoder:



Select a date range :

2023-05-31

2023-05-31

Encoding :

All encoding

File Type :

All types

☑ Show decoded files

Search Decoded by name:

Decoded ID

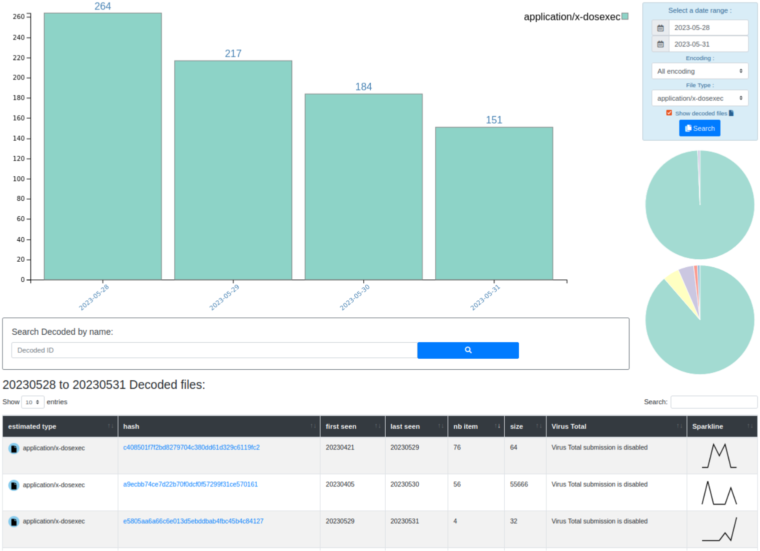
20230531 Decoded files:

Show 10 entries

Search:

estimated type	hash	first seen	last seen	nb item	size	Virus Total	Sparkline
image/gif	ee0bc07fe6baeb22c8ec1364a4bef2d840dc3e05	20230404	20230531	214708	1108	Virus Total submission is disabled	
image/png	b009399cefaa0e82086453da04a887105ca2f6a4	20230404	20230531	8404	1054	Virus Total submission is disabled	
application/json	f9f918ba5b0a35f5e2523bdbf4ece68d2cda11f9	20230410	20230531	3947	44	Virus Total submission is disabled	

Decoder:



Select a date range :

2023-05-28

2023-05-31

Encoding :

All encoding

File Type :

application/x-dosexec

☑ Show decoded files

Search Decoded by name:

Decoded ID

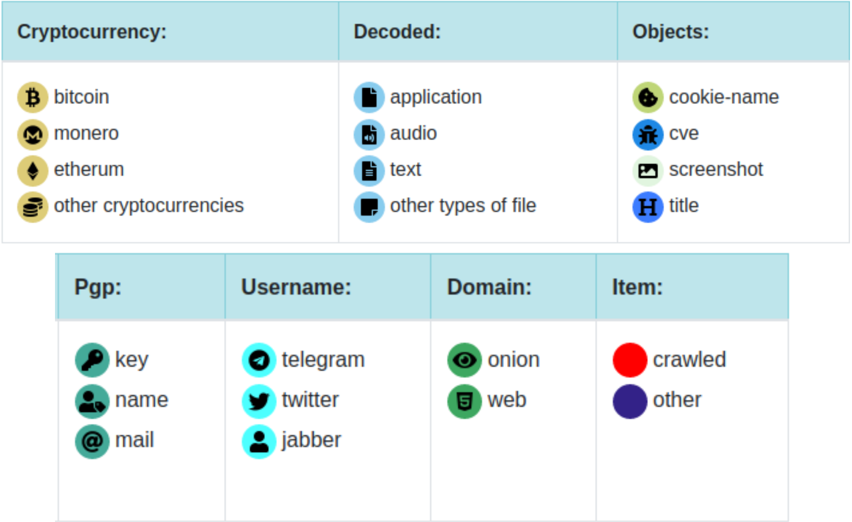
20230528 to 20230531 Decoded files:

Show 10 entries

Search:

estimated type	hash	first seen	last seen	nb item	size	Virus Total	Sparkline
application/x-dosexec	c408501f7f2bd8279704c380dd61d329c6119fc2	20230421	20230529	76	64	Virus Total submission is disabled	
application/x-dosexec	a9ecbb74ce7d22b70f0dcf0f57299f31ce570161	20230405	20230530	56	55666	Virus Total submission is disabled	
application/x-dosexec	e5805aa6a66c6e013d5ebddbab4fbc45b4c84127	20230529	20230531	4	32	Virus Total submission is disabled	

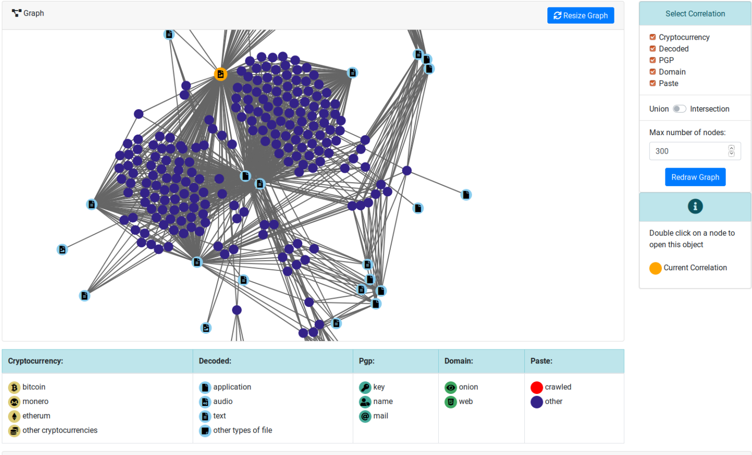
AIL Objects



Cryptocurrency:	Decoded:	Objects:
bitcoin monero etherum other cryptocurrencies	application audio text other types of file	cookie-name cve screenshot title

Pgp:	Username:	Domain:	Item:
key name mail	telegram twitter jabber	onion web	crawled other

Correlations and relationship



- ☑ Cryptocurrency
- ☑ Decoded
- ☑ PGP
- ☑ Domain
- ☑ Paste

Union ○ Intersection

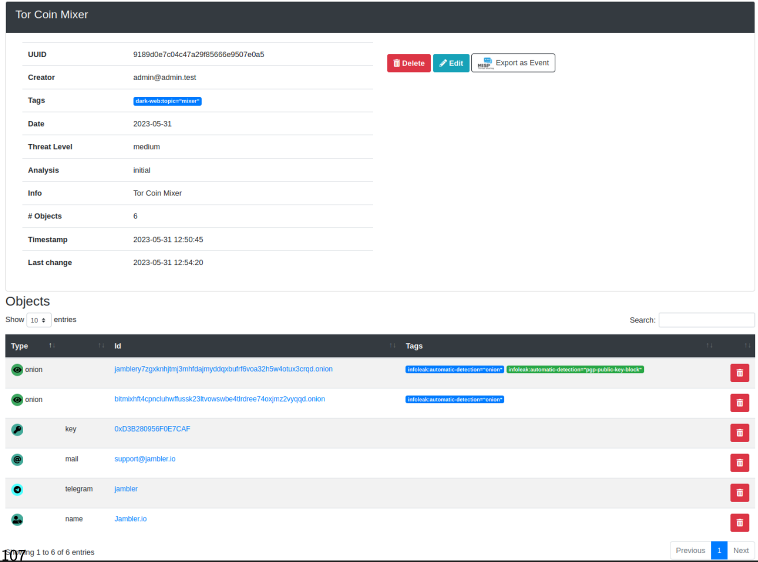
Max number of nodes: 300

Double click on a node to open this object

● Current Correlation

Cryptocurrency:	Decoded:	Pgp:	Domain:	Paste:
bitcoin	application	key	onion	crawled
monero	audio	name	web	other
etherum	text	mail		
other cryptocurrencies	other types of file			

Investigations



Tor Coin Mixer

UUID	9189d0e7c04c47a29f85666e9507e0a5
Creator	admin@admin.test
Tags	dark-web:topic="mixer"
Date	2023-05-31
Threat Level	medium
Analysis	initial
Info	Tor Coin Mixer
# Objects	6
Timestamp	2023-05-31 12:50:45
Last change	2023-05-31 12:54:20

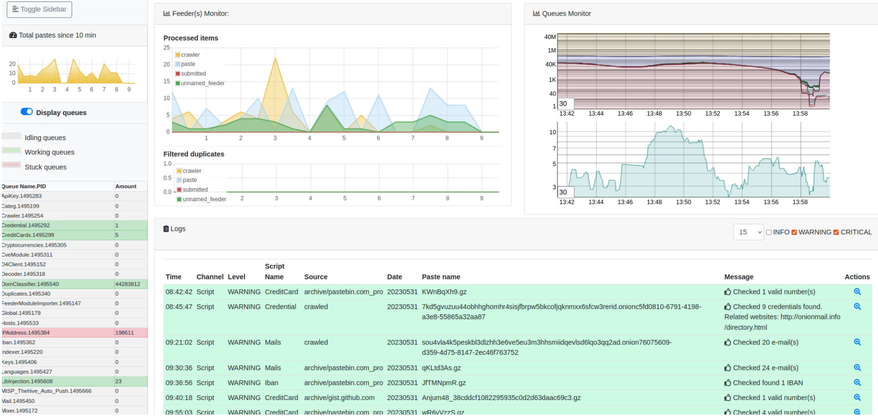
Objects

Type		Id	Tags	
onion		jamblery7zgxknhjtmj3mhfdajmyddqxbufrf6voa32h5w4otux3crqd.onion	infoleak:automatic-detection="onion" infoleak:automatic-detection="pgp-public-key-block"	
onion		bitmixhft4cpncluhwffussk23ltvowswbe4tlrdree74oxjmz2vyqqd.onion	infoleak:automatic-detection="onion"	
	key	0xD3B280956F0E7CAF		
	mail	support@jambler.io		
	telegram	jambler		
	name	Jambler.io		

Showing 1 to 6 of 6 entries

~~Live demo!~~

Example: Dashboard



Idling queues

Working queues

Stuck queues

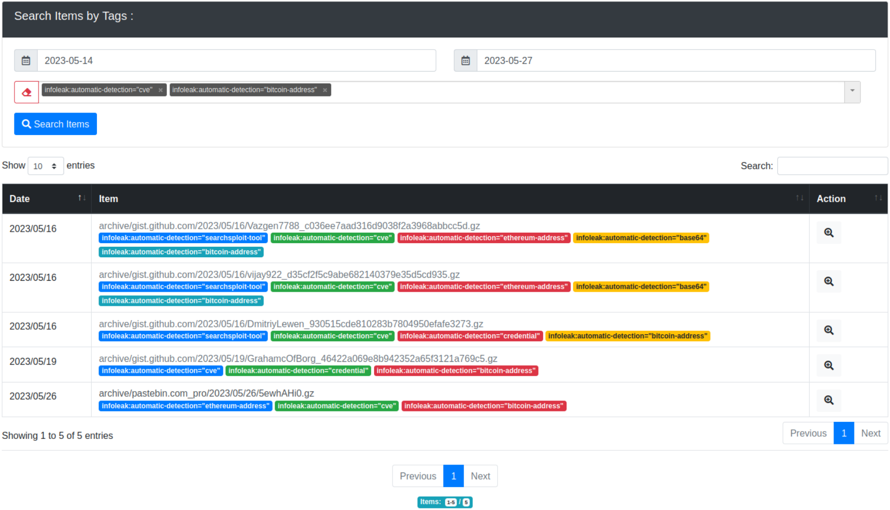
Queue Name.PID	Amount
ApiKey.1495283	0
Categ.1495199	0
Crawler.1495254	0
Credential.1495292	1
CreditCards.1495299	5
Cryptocurrencies.1495305	0
CveModule.1495311	0
D4Client.1495152	0
Decoder.1495318	0
DomClassifier.1495540	44283812
Duplicates.1495340	0
FeederModuleImporter.1495147	0
Global.1495179	0
Hosts.1495533	0
IPAddress.1495384	198611
Iban.1495362	0
Indexer.1495220	0
Keys.1495406	0
Languages.1495427	0
LibInjection.1495608	23
MISP_Thehive_Auto_Push.1495666	0
Mail.1495450	0
Mixer.1495172	0

Logs

15 □ INFO ☑ WARNING ☑ CRITICAL

Time	Channel	Level	Script Name	Source	Date	Paste name	Message	Actions
08:42:42	Script	WARNING	CreditCard	archive/pastebin.com_pro	20230531	KWnBqXh9.gz	Checked 1 valid number(s)	
08:45:47	Script	WARNING	Credential	crawled	20230531	7kd5gvuzuu44obhhghomhr4sisjfbrpw5bkcofjqknmxx6sfcw3rerid.onionc5fd0810-6791-4198-a3e8-55865a32aa87	Checked 9 credentials found. Related websites: http://onionmail.info /directory.html	
09:21:02	Script	WARNING	Mails	crawled	20230531	sou4vla4k5peskbl3dlzhh3e6ve5eu3m3hhsmiidqevlsd6lqo3qq2ad.onion76075609-d359-4d75-8147-2ec46f763752	Checked 20 e-mail(s)	
09:30:36	Script	WARNING	Mails	archive/pastebin.com_pro	20230531	qKLtd3As.gz	Checked 24 e-mail(s)	
09:36:56	Script	WARNING	Iban	archive/pastebin.com_pro	20230531	JfTMNpmR.gz	Checked found 1 IBAN	
09:40:18	Script	WARNING	CreditCard	archive/gist.github.com	20230531	Anjum48_38cddcf1082295935c0d2d63daac69c3.gz	Checked 1 valid number(s)	
09:55:03	Script	WARNING	CreditCard	archive/pastebin.com_pro	20230531	wR6vVzzS.gz	Checked 4 valid number(s)	

Example: Search by tags



Search Items by Tags :

2023-05-14

2023-05-27

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

Show 10 entries

Search:

Date	Item	Action
2023/05/16	archive/gist.github.com/2023/05/16/Vazgen7788_c036ee7aad316d9038f2a3968abbcc5d.gz infoleak:automatic-detection="searchsploit-tool" infoleak:automatic-detection="cve" infoleak:automatic-detection="ethereum-address" infoleak:automatic-detection="base64" infoleak:automatic-detection="bitcoin-address"	
2023/05/16	archive/gist.github.com/2023/05/16/vijay922_d35cf2f5c9abe682140379e35d5cd935.gz infoleak:automatic-detection="searchsploit-tool" infoleak:automatic-detection="cve" infoleak:automatic-detection="ethereum-address" infoleak:automatic-detection="base64" infoleak:automatic-detection="bitcoin-address"	
2023/05/16	archive/gist.github.com/2023/05/16/DmitriyLewen_930515cde810283b7804950efafe3273.gz infoleak:automatic-detection="searchsploit-tool" infoleak:automatic-detection="cve" infoleak:automatic-detection="credential" infoleak:automatic-detection="bitcoin-address"	
2023/05/19	archive/gist.github.com/2023/05/19/GrahamcOfBorg_46422a069e8b942352a65f3121a769c5.gz infoleak:automatic-detection="cve" infoleak:automatic-detection="credential" infoleak:automatic-detection="bitcoin-address"	
2023/05/26	archive/pastebin.com_pro/2023/05/26/5ewhAHi0.gz infoleak:automatic-detection="ethereum-address" infoleak:automatic-detection="cve" infoleak:automatic-detection="bitcoin-address"	

Showing 1 to 5 of 5 entries

MISP

MISP Taxonomies

- **Tagging** is a simple way to attach a classification to an event or 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 [7].
- Can be easily cherry-picked or extended

[7] `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**[8]: Evaluate the degree of identifiability of personal data and the types of pseudonymous data, de-identified data and anonymous data.

[8] Future of Privacy Forum

Taxonomies useful in AIL

- **tor**: Describe Tor network infrastructure.
- **dark-web**: Criminal motivation on the dark web.
- **copine-scale**[9]: Categorise the severity of images of child sex abuse.

[9] Combating Paedophile Information Networks in Europe

threat sharing and incident response platforms

ail project ⟶ MISP Threat Sharing

Goal: submission to threat sharing and incident response platforms.

threat sharing and incident response platforms

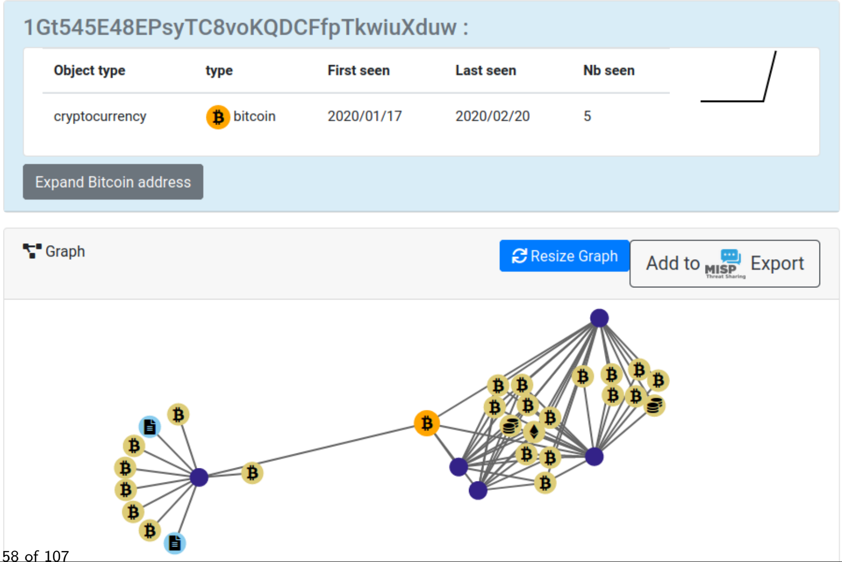
ail project ⟶ MISP Threat Sharing

1. Use infoleak taxonomy[10]
2. Add your own tags
3. Export AIL objects to MISP core format
4. Download it or Create a MISP Event[11]

[10] https://www.misp-project.org/taxonomies.html

[11] https://www.misp-standard.org/rfc/misp-standard-core.txt

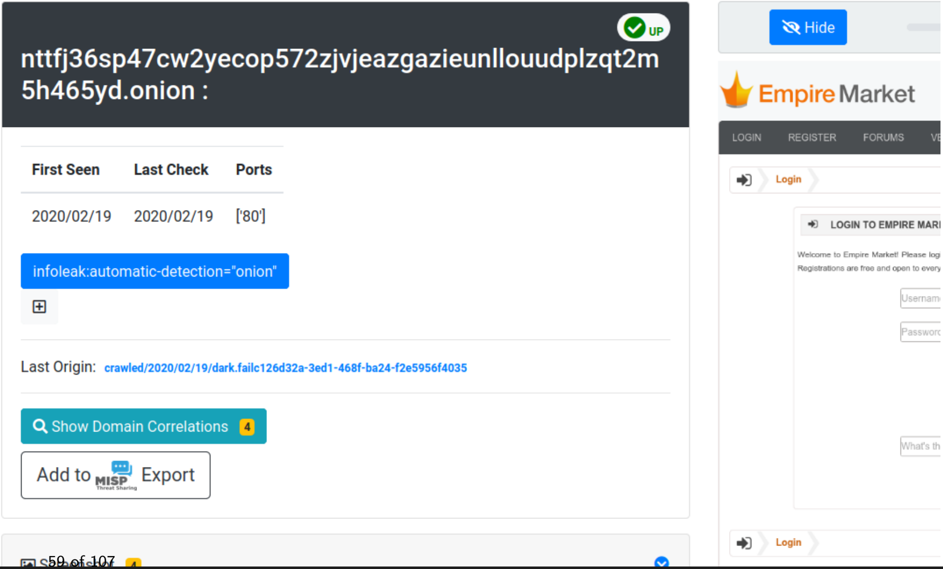
MISP Export



1Gt545E48EPsyTC8voKQDCFfpTkwiuXduw :

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

MISP Export



UP

nttfj36sp47cw2yecop572zjvjeazgazieunllouudplzqt2m5h465yd.onion :

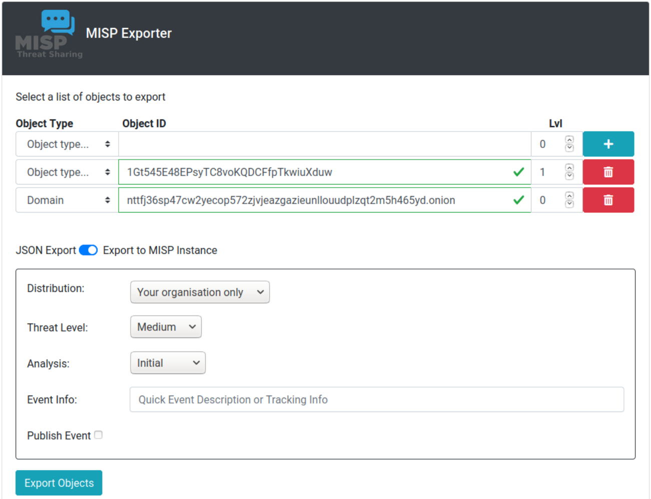
First Seen	Last Check	Ports
2020/02/19	2020/02/19	['80']

infoleak:automatic-detection="onion"

Last Origin: crawled/2020/02/19/dark.failc126d32a-3ed1-468f-ba24-f2e5956f4035

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

MISP Export



MISP Exporter

Select a list of objects to export

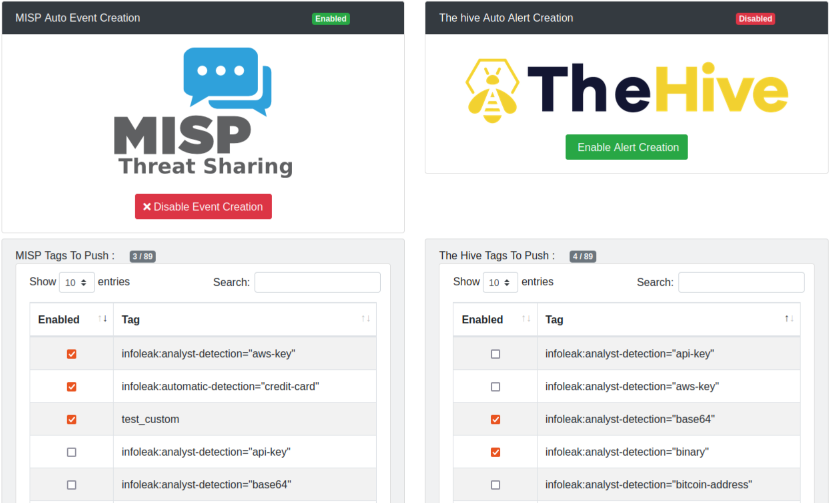
Object Type	Object ID	Lvl	
Object type...		0	+
Object type...	1Gt545E48EPsyTC8voKQDCFfpTkwiuXduw ✓	1	🗑
Domain	nttfj36sp47cw2yecop572zjvjeazgazieunllouudplzqt2m5h465yd.onion ✓	0	🗑

JSON Export ● Export to MISP Instance

Distribution:	Your organisation only
Threat Level:	Medium
Analysis:	Initial
Event Info:	Quick Event Description or Tracking Info

Publish Event □

Automatic MISP Export on tags



MISP Auto Event Creation — Enabled

The hive Auto Alert Creation — Disabled

MISP Tags To Push : 3 / 89

Show 10 entries Search:

Enabled	Tag
☑	infoleak:analyst-detection="aws-key"
☑	infoleak:automatic-detection="credit-card"
☑	test_custom
□	infoleak:analyst-detection="api-key"
□	infoleak:analyst-detection="base64"

The Hive Tags To Push : 4 / 89

Show 10 entries Search:

Enabled	Tag
□	infoleak:analyst-detection="api-key"
□	infoleak:analyst-detection="aws-key"
☑	infoleak:analyst-detection="base64"
☑	infoleak:analyst-detection="binary"
□	infoleak:analyst-detection="bitcoin-address"

Setting up the framework

AIL ecosystem: Technologies used

Programming language: Full python3

Databases: Redis and Kvrocks

Server: Flask

Data message passing: Redis Set

Setting up AIL-Framework from source

Setting up AIL-Framework from source

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

Starting the framework

Running your own instance from source

Accessing the environment and starting AIL

```

# Launch the system and the web interface
cd bin/
./LAUNCH -l
```

Feeding the framework

Feeding Data to AIL

There are different ways to feed data into AIL:

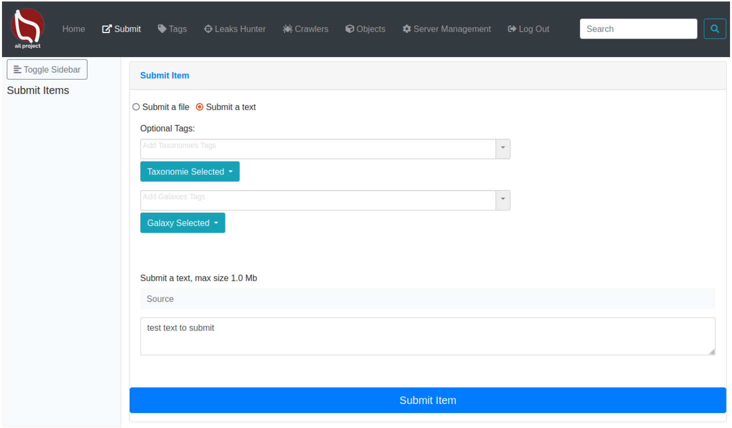
1. AIL Importers:
   - Dir / Files
   - ZMQ
   - *pystemon*
2. AIL Feeders (discord, telegram, ActivityPub, ...)
3. Feed your own data using the API
4. Feed your own file/text using the UI (`Submit section`)

Feeding Data to AIL - Technical Considerations

- It is important to consider the size of each file being fed into AIL:
  - For optimal processing and efficiency, it is recommended to keep each file around 3 MB in size
  - This balance between processing capabilities and file size is crucial, as certain modules perform various computations, such as regexp matching, which has a default timeout of 30 seconds
  - If you need to process a large file, it is advisable to split it into multiple smaller files. The AIL leak feeder tool[13] can assist you in this task.

[13] https://github.com/ail-project/ail-feeder-leak

Via the UI (1)



○ Submit a file ● Submit a text

Optional Tags:

Add Taxonomies Tags

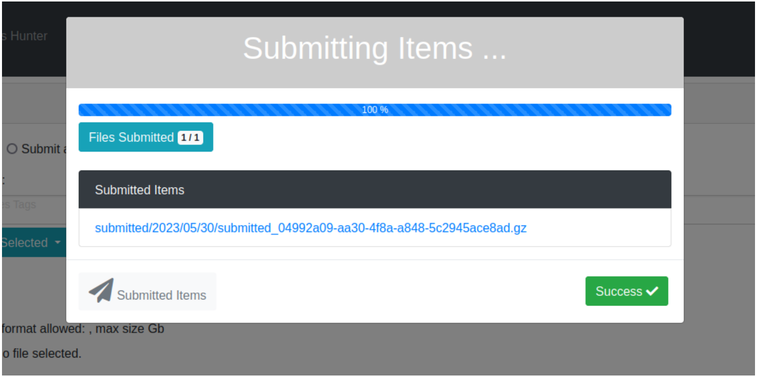
Add Galaxies Tags

Submit a text, max size 1.0 Mb

Source

test text to submit

Via the UI (2)



submitted/2023/05/30/submitted_04992a09-aa30-4f8a-a848-5c2945ace8ad.gz

format allowed: , max size Gb

o file selected.

API - Feeding AIL with your own data

api/v1/import/item

```
{
  "type": "text",
  "tags": [
    "infoleak:analyst-detection=\"private-key\""
  ],
  "text": "text to import"
}
```

Importers

- Importers are located in the `/bin/importer` directory
- They are used to import different types of data into AIL
- Adding new Importers is straightforward.
- Available Importers:
  - AIL Feeders
  - ZMQ
  - pystemon
  - Files

File Importer

- `importer/FileImporter.py`

Import File

```
. ./AILENV/bin/activate
cd tools/
./file_dir_importer.py -f MY_FILE_PATH
```

Import Dir

```
. ./AILENV/bin/activate
cd tools/
./file_dir_importer.py -d MY_DIR_PATH
```

AIL feeders Importers

- **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 push JSON metadata
- Feeder are then pushing the generated JSON to AIL API

Certificate transparency feeder for AIL

- ail-feeder-cti[14] 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

[14] https://github.com/ail-project/ail-feeder-ct

GitHub archive and GitHub repository

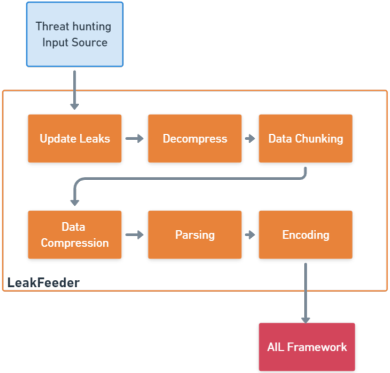
- ail-feeder-gharchive[15] is a generic software to extract informations from **GHArchive**, collect and feed AIL via AIL ReST API
- ail-feeder-github-repo[16] is collecting from a GitHub repository and push everything to AIL
- For monitoring a set of **suspicious git repositories** or finding leaks on existing or managed git repositories, it's a simple way to feed AIL with such source.

[15] https://github.com/ail-project/ail-feeder-gharchive

[16] https://github.com/ail-project/ail-feeder-github-repo

AIL LeakFeeder

- ail-feeder-leak[17] automates the process to feed leaked large files automatically to AIL



[17] https://github.com/ail-project/ail-feeder-leak

AIL feeder ActivityPub

- ail-feeder-activity-pub[18] is feeder for the ActivityPub standard used in distributed social networks (e.g. Mastodon)
- Accounts are required on the ActivityPub instance to get the stream

[18] https://github.com/ail-project/ail-feeder-activity-pub

AIL feeder telegram

- ail-feeder-telegram[19] is a **Telegram feeder**
- An API ID/hash for Telegram is required and linked to your Telegram phone number

[19] https://github.com/ail-project/ail-feeder-telegram

More feeders

- ail-feeder-discord[20] is a generic **Discord** feeder for AIL
- ail-feeder-atom-rss[21] is an **Atom and RSS reader** and feeder for AIL
- ail-feeder-jsonlogs[22] is a **JSON aggregator** to submit generic JSON input into AIL

[20] https://github.com/ail-project/ail-feeder-discord

[21] https://github.com/ail-project/ail-feeder-atom-rss

[22] https://github.com/ail-project/ail-feeder-jsonlogs

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!

〈(^.^)ノ

Ongoing developments

- MISP Importer
- Bloom filter filtering
- Data retention and lifetime management of objects
- MISP modules expansion
- 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

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.

Contact

- CIRCL has developed a range of open-source tools for intelligence analysts and incident responders.
- We welcome partnerships and collaboration discussions. Feel free to contact us[23].

[23] `mailto:info@circl.lu`