

# D4 Project

Open and collaborative network monitoring

Team CIRCL

<https://www.d4-project.org/>

2019/05/21

Jean-Louis Huynen



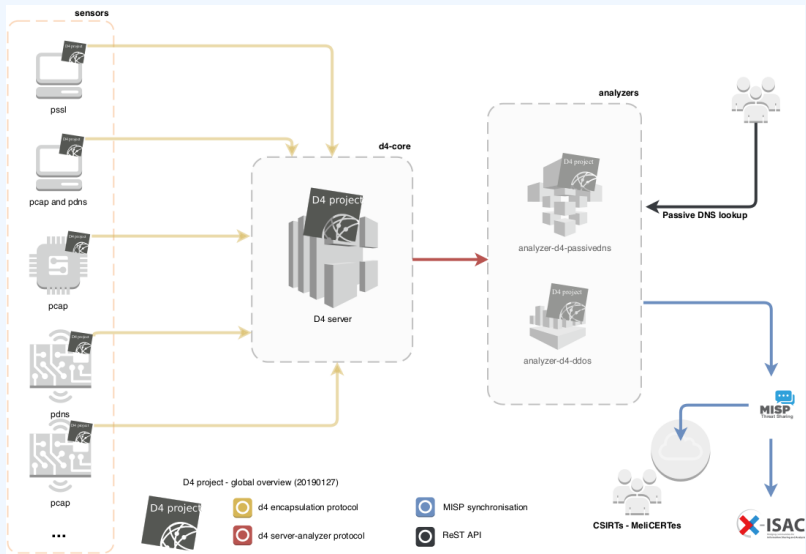
- CSIRTs (or private organisations) build their **own honeypot, honeynet or blackhole monitoring network**
- Designing, managing and operating such infrastructure is a tedious and resource intensive task
- **Automatic sharing** between monitoring networks from different organisations is missing
- Sensors and processing are often seen as blackbox or difficult to audit

- Based on our experience with MISP<sup>1</sup> where sharing played an important role, we transpose the model in D4 project
- Keeping the protocol and code base **simple and minimal**
- Allowing every organisation to **control and audit their own sensor network**
- Extending D4 or **encapsulating legacy monitoring protocols** must be as simple as possible
- Ensuring that the sensor server has **no control on the sensor** (unidirectional streaming)
- Don't force users to use dedicated sensors and allow **flexibility of sensor support** (software, hardware, virtual)

---

<sup>1</sup><https://github.com/MISP/MISP>

# D4 OVERVIEW



- D4 Project (co-funded under INEA CEF EU program) started - **1st November 2018**
- D4 encapsulation protocol version 1 published - **1st December 2018**
- vo.1 release of the D4 core<sup>2</sup> including a server and simple D4 C client - **21st January 2019**
- First version of a golang D4 client<sup>3</sup> running on ARM, MIPS, PPC and x86 - **14th February 2019**

---

<sup>2</sup><https://www.github.com/D4-project/d4-core>

<sup>3</sup><https://www.github.com/D4-project/d4-goclient/>

## (SHORT) HISTORY

Release	Date
analyzer-d4-passivedns-vo.1	Apr. 5, 2019
analyzer-d4-passivessl-0.1	Apr. 25, 2019
analyzer-d4-pibs-vo.1	Apr. 8, 2019
BGP-Ranking-1.0	Apr. 25, 2019
d4-core-vo.1	Jan. 25, 2019
d4-core-vo.2	Feb. 14, 2019
d4-core-vo.3	Apr. 8, 2019
d4-goclient-vo.1	Feb. 14, 2019
d4-goclient-vo.2	Apr. 8, 2019
d4-server-packer-0.1	Apr. 25, 2019
IPASN-History-1.0	Apr. 25, 2019
sensor-d4-tls-fingerprinting-0.1	Apr. 25, 2019

see <https://github.com/D4-Project>

CIRCL will host a server instance for organisations willing to contribute to a public dataset without running their own D4 server:

- ✓ Blackhole DDoS
- ✓ Passive DNS
- ✓ Passive SSL
- BGP mapping
- egress filtering mapping
- Radio-Spectrum monitoring: 802.11, BLE, etc.
- ...

# D4 ENCAPSULATION PROTOCOL

stream of information  
(text or binary)

```
010111010100
100010101101
010100101011
010100100100
011010100101

01011101010010
10001010110101
01010010101111
01010010010100
01101010010101

01011101010010
10001010110101
01010010101111
01010010010100
01101010010101
```



D4 encapsulation protocol version 1



*version* (8) - Version of the header  
*type* (8) - Data encapsulated type  
*uuid* (128) - Sensor UUID  
*timestamp* (64) - Encapsulation time  
*hmac* (256) - Header authentication  
(HMAC-SHA256-128)  
*size* (32) - Payload size



<https://www.d4-project.org>



Name	bit size	Description
version	uint 8	Version of the header
type	uint 8	Data encapsulated type
uuid	uint 128	Sensor UUID
timestamp	uint 64	Encapsulation time
hmac	uint 256	Authentication header (HMAC-SHA-256-128)
size	uint 32	Payload size

Type	Description
0	Reserved
1	pcap (libpcap 2.4)
2	meta header (JSON)
3	generic log line
4	dnscap output
5	pcapng (diagnostic)
6	generic NDJSON or JSON Lines
7	generic YAF (Yet Another Flowmeter)
8	passivedns CSV stream
254	type defined by meta header (type 2)

D4 header includes an easy way to **extend the protocol** (via type 2) without altering the format. Within a D4 session, the initial D4 packet(s) type 2 defines the custom headers and then the following packets with type 254 is the custom data encapsulated.

```
{
  "type": "ja3-jl",
  "encoding": "utf-8",
  "tags": [
    "tlp:white"
  ],
  "misp:org": "5b642239-4db4-4580-adf4-4ebd950d210f"
}
```

- D4 core server<sup>4</sup> is a complete server to handle clients (sensors) including the decapsulation of the D4 protocol, control of sensor registrations, management of decoding protocols and dispatching to adequate decoders/analysers.
- D4 server is written in Python 3.6 and runs on standard GNU/Linux distribution.

---

<sup>4</sup><https://github.com/D4-project/d4-core>

The D4 server provides a **web interface** to manage D4 sensors, sessions and analyzer.

- Get Sensors status, errors and statistics
- Get all connected sensors
- Manage Sensors (stream size limit, secret key, ...)
- Manage Accepted types
- UUID/IP blocklist
- Create Analyzer Queues

# D4 SERVER - MAIN INTERFACE


The screenshot displays the D4 Server Main Interface. At the top, there is a navigation bar with the D4 project logo and links for Home, Sensors Status, and Server Management. The main content is divided into two panels: 'UUID' and 'Types'. The 'UUID' panel shows a list of sensors with their IDs and UUIDs. The 'Types' panel shows the types of sensors, including 'pcap (libpcap 2.4)' and 'passivedns CSV stream'. The date '2019/05/20' is displayed at the bottom of both panels.


UUID	
4019794	c0bb49e788964718af4dfea4c0ab898c
47820	bbbcf7a43aed47aa84badc50262f5aba
27183	37d2f040fc074aaab2caf49059667525
8401	1b06b4ab8a754ef9ae3dd4d073b38f0e5
1022	de1df62d862b494a830f1f78ec27fca5



Types	
4046981	1: pcap (libpcap 2.4)
57243	8: passivedns CSV stream

2019/05/20

2019/05/20

 CIRCL  
Computer Incident  
Response Center  
Luxembourg

 Co-financed by the Connecting Europe  
Facility of the European Union

# D4 SERVER - SERVER MANAGEMENT

The screenshot displays the 'Server Management' section of the D4 project interface. It features a navigation bar with 'Home', 'Sensor Status', and 'Server Management'. The main content is divided into two columns: 'Blacklist IP' and 'Blacklist UUID'. Each column contains three panels: 'Blacklist IP' with an 'IP Address' input and a 'Blacklist IP' button; 'Manage IP Blacklist' with a 'Show Blacklisted IP' button; and 'Unblacklist IP' with an 'IP Address' input and an 'Unblacklist IP' button. The 'Blacklist UUID' column has a similar structure with 'UUID' input and 'Blacklist UUID', 'Manage UUID Blacklist' with 'Show Blacklisted UUID', and 'Unblacklist UUID' with 'Unblacklist UUID' buttons.

Below the blacklist management is the 'Header Accepted Types' section. It includes a search bar and a table of accepted types. The table has columns for 'Type', 'Description', and 'Remove Type'. The entries are:

Type	Description	Remove Type
1	pcap (libcap 2.4)	Remove Type
2	meta header (JSON)	Remove Type
4	dnscap output	Remove Type
8	passwdns CSV stream	Remove Type
254	type defined by meta header (type2)	Remove Type

Navigation controls for this table include 'Showing 1 to 5 of 5 entries', 'Previous', '1', and 'Next'. To the right of the table is an 'Add New Types' panel with a text input containing '1' and an 'Add New Type' button.

Below the first table is a second table for 'Extended Types' with columns for 'Type Name', 'Description', and 'Remove Type'. It contains one entry:

Type Name	Description	Remove Type
ja3-f		Remove Extended Type

This table also has navigation controls: 'Showing 1 to 1 of 1 entries', 'Previous', '1', and 'Next'.

# D4 SERVER - SERVER MANAGEMENT

Analyzer Management

Show 10 ▾ entries Search

Type	uuid	last updated	Change max size limit	Analyzer Queue
1	f72ea760-370b-4f99-bb93-b6c6e645a32	2019-05-20 14:14:23	10000 <input type="text"/> Change Max Size	<input type="text" value="10001"/>
8	4072e072-bfaa-4395-9bb1-ccb3e470d715	2019-05-20 14:14:57	10000 <input type="text"/> Change Max Size	<input type="text" value="0"/>

Showing 1 to 2 of 2 entries Previous 1 Next

Show 10 ▾ entries Search

Type Name	uuid	last updated	Change max size limit	Analyzer Queue
ja3-f	8d8b724c71b64d6c942bffc2b6d761ac <small>This analyzer pushes TLS sessions into a postgres database for passiveSSL.</small>	2019-05-14 08:50:31	100000 <input type="text"/> Change Max Size	<input type="text" value="18036"/>

Showing 1 to 1 of 1 entries Previous 1 Next

Add New Analyzer Queue

1

Optional Description



# D4 SERVER - SENSOR OVERVIEW

Active Connection			Search
UID: 041803062376450779607608			
First Seen	Last Seen	Status	
2019-09-20 11:05:05 - (1504081940)	2019-09-20 13:55:23 - (150406366)	OK ● Connected	
UID: 1006060607644766060607000606			
First Seen	Last Seen	Status	
2019-04-09 12:27:42 - (1504703402)	2019-05-20 14:19:08 - (1500041940)	OK ● Connected	
UID: 01020606076606060606060606			
First Seen	Last Seen	Status	
2019-04-01 11:46:31 - (1504118190)	2019-05-20 14:17:35 - (1500041875)	OK ● Connected	
UID: 00060766060606060606060606			
First Seen	Last Seen	Status	
2019-04-02 07:16:49 - (1504080402)	2019-05-20 14:17:35 - (1500041875)	OK ● Connected	
UID: 03060607660606060606060606			
First Seen	Last Seen	Status	
2019-04-09 13:06:12 - (1504702952)	2019-05-20 14:17:35 - (1500041875)	OK ● Connected	

# D4 SERVER - SENSOR MANAGEMENT

D4 project



[Home](#)

[Sensors Status](#)

[Server Management](#)

UUID: de1df62d862b494a830f178ec27fca5

First Seen	Last Seen	Status
2019-03-31 11:03:05 - (1554030185)	2019-05-20 13:56:23 - (1558360583)	OK <input checked="" type="checkbox"/> Connected <a href="#">Kick UUID</a>

### Change Stream Max Size

10000

[Change Max Size](#)

### UUID Blacklist

[Blacklist UUID](#)

### Blacklist IP Using This UUID

[Blacklist IP](#)

### Change UUID Key

private key to change

[Change UUID Key](#)

### Types Used:

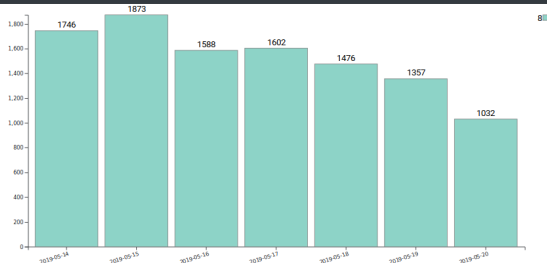
Show 10 entries

Search:

Type	first seen	last seen
8	2019-04-04 12:46:43	2019-05-20 13:56:23

Showing 1 to 1 of 1 entries

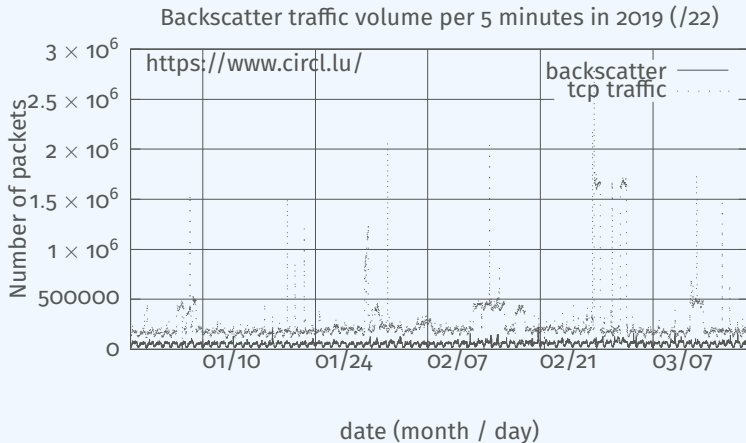
[Previous](#) [1](#) [Next](#)



## A distributed Network telescope to observe DDoS attacks



DDoS Attacks produce an observable side-effect:



- External point of view on ongoing Denial of Service attacks:
  - ▶ **Confirm** if there is a DDoS attack
  - ▶ **Recover** time line of attacked targets
  - ▶ **Confirm** which services (DNS, webserver, ...)
  - ▶ **Observe** Infrastructure changes
- **Assess the state of an infrastructure under denial of service attack**
  - ▶ **Detect** failure/addition of intermediate network equipments, firewalls, proxy servers etc
  - ▶ **Detect** DDoS mitigation devices
- Create models of DoS/DDoS attacks

D4 - for data collection and processing:

- **provide** various points of observation in non contiguous address space,
- **aggregate** and **mix** backscatter traffic collected from D4 sensors,
- **perform** analysis on big amount of data.

D4 - from a end-user perspective:

- **provide** backscatter analysis results,
- **provide** daily updates,
- **provide** additional relevant (or pivotal) information (DNS, BGP, etc.),
- **provide** an API and search capabilities.

- ✓ analyzer-d4-pibs<sup>5</sup>, an analyzer for a D4 network sensor:
  - ▶ **processes** data produced by D4 sensors (pcaps),
  - ▶ **displays** potential backscatter traffic on standard output.

---

<sup>5</sup><https://github.com/D4-project/analyzer-d4-pibs>

## Passive DNS



- CIRCL (and other CSIRTs) have their own passive DNS<sup>6</sup> collection mechanisms
- Current **collection models** are affected with DoH<sup>7</sup> and centralised DNS services
- DNS answers collection is a tedious process
- **Sharing Passive DNS stream** between organisation is challenging due to privacy

---

<sup>6</sup><https://www.circl.lu/services/passive-dns/>

<sup>7</sup>DNS over HTTPS

- Improve **Passive DNS collection diversity** by being closer to the source and limit impact of DoH (e.g. at the OS resolver level)
- Increasing diversity and **mixing models** before sharing/storing Passive DNS records
- Simplify process and tools to install for **Passive DNS collection by relying on D4 sensors** instead of custom mechanisms
- Provide a distributed infrastructure for mixing streams and filtering out the sharing to the validated partners

- ✓ analyzer-d4-passivedns<sup>8</sup>, an analyzer for a D4 network sensor:
  - ▶ **processes** data produced by D4 sensors (in passivedns CSV format<sup>9</sup>),
  - ▶ **ingests** these into a **Passive DNS server** which can be queried later to search for the Passive DNS records,
  - ▶ **provides** a lookup server (using on redis-compatible backend) that is a Passive DNS REST server compliant to the Common Output Format<sup>10</sup>.

---

<sup>8</sup><https://github.com/D4-project/analyzer-d4-passivedns>

<sup>9</sup><https://github.com/gamlinux/passivedns>

<sup>10</sup><https://tools.ietf.org/html/draft-dulaunoy-dnsop-passive-dns-cof-04>

## Passive SSL revamping

CSIRT's rationale for collecting TLS handshakes:

- **pivot** on additional data points,
- **find** owners of IP addresses,
- **detect** usage of CIDR blocks,
- **detect** vulnerable systems,
- **detect** compromised services,
- **detect** key material reuse,
- **detect** weak keys.

**Keeping** a log of links between:

- x509 certificates,
- ports,
- IP address,
- client (ja3),
- server (ja3s),

*“JA3 is a method for creating SSL/TLS client fingerprints that should be easy to produce on any platform and can be easily shared for threat intelligence.”<sup>11</sup>*

---

<sup>11</sup><https://github.com/salesforce/ja3>

**Collect** and **store** x509 certificates and TLS sessions:

- Public keys type and size,
- moduli and exponents,
- curves parameters.

**Detect** broken crypto:

- Public Key reuse,
- Moduli that share one prime factor,
- Moduli that share both prime factor,
- Small factors,
- Nonces reuse / common prefix or suffix, etc.

- ✓ sensor-d4-tls-fingerprinting<sup>12</sup>: **Extracts** and **fingerprints** certificates, and **computes** TLSH fuzzy hash.
- ✓ analyzer-d4-passivessl<sup>13</sup>: **Stores** Certificates / PK details in a PostgreSQL DB.
- lookup-d4-passivessl<sup>14</sup>: **Exposes** the DB through a public REST API.

---

<sup>12</sup>[github.com/D4-project/sensor-d4-tls-fingerprinting](https://github.com/D4-project/sensor-d4-tls-fingerprinting)

<sup>13</sup>[github.com/D4-project/analyzer-d4-passivessl](https://github.com/D4-project/analyzer-d4-passivessl)

<sup>14</sup>[github.com/D4-project/lookup-d4-passivessl](https://github.com/D4-project/lookup-d4-passivessl)



- **Mixing models for passive collection streams** (for privacy) in next version of D4 core server
- **Interconnecting private D4 sensor networks** with other D4 sensor networks (sharing to partners filtered stream)
- **Previewing datasets** collected in D4 sensor network and providing **open data stream** (if contributor agrees to share under specific conditions)
- **Leverage MISP sharing communities** to augment Threat Intelligence, and provide accurate metrology.

# GET IN TOUCH IF YOU WANT TO JOIN THE PROJECT, HOST A SENSOR OR CONTRIBUTE

- Collaboration can include research partnership, sharing of collected streams or improving the software.
- Contact: [info@circl.lu](mailto:info@circl.lu)
- <https://github.com/D4-Project>
- [https://twitter.com/d4\\_project](https://twitter.com/d4_project)
- <https://d4-project.org>