

---

# Domain Trails

*Release 1.0.0*

**iAbdullahMughal**

**Dec 09, 2021**



# README

<b>1</b>	<b>Readme</b>	<b>3</b>
<b>2</b>	<b>Guide</b>	<b>5</b>
2.1	Getting Started . . . . .	5
2.2	domaintrails . . . . .	8
<b>3</b>	<b>Indices and tables</b>	<b>13</b>
	<b>Python Module Index</b>	<b>15</b>
	<b>Index</b>	<b>17</b>



Welcome to domain-trails docs. domain-trails is open source intelligence gathering tool. This tool collect information related to urls. Working as educational project and trying to complete the project with native python code.



---

CHAPTER  
**ONE**

---

**README**





## 2.1 Getting Started

Domain tails is python tool based on free available sources for domain information collection.

### 2.1.1 Installation

#### Preparing installation

Install python3 on system. Currently project has zero external dependency verify python version:

```
python -V  
Python 3.6.8
```

---

**Note:** Project is currently supported with python3+ versions.

---

#### Cloning Project

Clone repo on system / server:

```
git clone https://github.com/iAbdullahMughal/domain-trails.git
```

Your file system should now look similar to this:

```
domain-trails  
├── docs  
├── LICENSE  
├── README.md  
├── portal.py  
├── setup.py  
├── core  
│   ├── parser  
│   ├── recon  
│   ├── __init__.py  
│   └── resources.py
```

We have a top-level `portal.py` directory in the main project directory. .. warning:: Project directory structure may change.

## Executing Project

Project without parameters:

```
python .\portal.py
-----
/ _ _ \ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ ( ) _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ /
/ / / / / _ _ \ / _ _ \ _ _ \ / _ _ \ / / / / / _ _ \ / / / / / _ _ \ /
/ / / / / / / / / / / / / / / / / / / / / / / / / / / / / / / / / / / / /
/ _ _ _ / \ _ _ _ / / / / / / / / \ _ _ , / / / / / / / / / / / / / / / / / /
-----
Domain Trails - domains footprints, reconnaissance & information gathering
-----
usage: portal.py [-h] -d DOMAIN
portal.py: error: the following arguments are required: -d
```

Run project with parameter:

```
python .\portal.py -d www.google.com
-----
/ _ _ \ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ ( ) _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ /
/ / / / / _ _ \ / _ _ \ _ _ \ / _ _ \ / / / / / _ _ \ / / / / / _ _ \ /
/ / / / / / / / / / / / / / / / / / / / / / / / / / / / / / / / / / / / /
/ _ _ _ / \ _ _ _ / / / / / / / / \ _ _ , / / / / / / / / / / / / / / / / / /
-----
Domain Trails - domains footprints, reconnaissance & information gathering
-----
Printing Results for domain www.google.com
```

## 2.1.2 Supported Modules

---

**Note:** Actively adding more modules for more coverage.

---

Following modules are supported in domain trails

### Domain Availability Check

Output:

```
Domain Availability Result for www.google.com : Domain is Registered
```

## Domain Hosting History

Output:

### DNS History Records

Old Web Host	New Web Host	Month / Year	Zone Date	Transaction
google.com	n/a	december 2000		epoch

## Domain DNS Information

Output:

### IP v6 Records

name	type	class	ttl	endpoint
google.com.	AAAA	IN	300	2607:f8b0:4006:817::200e

### Mail eXchanger (MX) Records

name	type	class	ttl	endpoint
google.com.	MX	IN	600	30 alt2.aspmx.l.google.com.
google.com.	MX	IN	600	50 alt4.aspmx.l.google.com.
google.com.	MX	IN	600	10 aspmx.l.google.com.
google.com.	MX	IN	600	20 alt1.aspmx.l.google.com.
google.com.	MX	IN	600	40 alt3.aspmx.l.google.com.

### Parent Name Server records

name	type	class	ttl	endpoint
google.com.	A	IN	300	142.250.81.238

### Local Name Server Records

name	type	class	ttl	endpoint
google.com.	NS	IN	271654	ns4.google.com.
google.com.	NS	IN	271654	ns3.google.com.
google.com.	NS	IN	271654	ns1.google.com.
google.com.	NS	IN	271654	ns2.google.com.

### Start of Authority (SOA)

primary_nameserver	host_master_email	serial_number	refresh	retry	expire	↳
↳ minimum_ttl						
↳						
ns1.google.com.	dns-admin.google.com.	413628036	900	900	1800	60

(continues on next page)

(continued from previous page)

## Text Records

name	type	class	ttl	endpoint
google.com.	TXT	IN	3600	facebook-domain-verification=22rm551cu4k0ab0bxsw536tlds4h95
google.com.	TXT	IN	3600	google-site-verification=wD8N7i1JTNTkezJ49swvWW48f8_9xveREV4oB-0Hf5o
google.com.	TXT	IN	3600	docusign=1b0a6754-49b1-4db5-8540-d2c12664b289
google.com.	TXT	IN	3600	globalsign-smime-dv=CDYX+XFHUw2wml6/Gb8+59BsH31KzUr6c1l2BPvqKX8=
google.com.	TXT	IN	3600	google-site-verification=TV9-DBe4R80X4v0M4U_bd_J9cp0JM0nikft0jAgjmsQ
google.com.	TXT	IN	3600	"v=spf1 include:_spf.google.com ~all"
google.com.	TXT	IN	3600	docusign=05958488-4752-4ef2-95eb-aa7ba8a3bd0e
google.com.	TXT	IN	3600	MS=E4A68B9AB2BB9670BCE15412F62916164C0B20BB
google.com.	TXT	IN	3600	apple-domain-verification=30afIBcvSuDV2PLX

**Warning:** Domain name records may change there

## 2.2 domaintrails

### 2.2.1 domaintrails package

#### Subpackages

#### domaintrails.core package

#### Subpackages

#### domaintrails.core.parser package

#### Submodules

#### domaintrails.core.parser.html\_parser module

**class** domaintrails.core.parser.html\_parser.DnsHistoryParser(*html\_content: str*)

Bases: `object`

Custom parser written for extracting data from html.

**parse()** → `list`

This functions invokes parsing of html content :return: historical records against website :rtype: `object`

## Module contents

### class domaintrails.core.parser.ColorPrint

Bases: `object`

This class provides support for color printing text in console.

#### static print\_bold(message: str, end: str = '\n') → None

Print text in bold format in console :param message: message which needs to be printed on console. :type message: str :param end: new line :type end: str :return: None :rtype: None

#### static print\_fail(message: str, end: str = '\n') → None

Print text in red color in console :param message: message which needs to be printed on console. :type message: str :param end: new line :type end: str :return: None :rtype: None

#### static print\_info(message: str, end: str = '\n') → None

Print text in green color in console :param message: message which needs to be printed on console. :type message: str :param end: new line :type end: str :return: None :rtype: None

#### static print\_pass(message: str, end: str = '\n') → None

Print text in green color in console :param message: message which needs to be printed on console. :type message: str :param end: new line :type end: str :return: None :rtype: None

#### static print\_warn(message: str, end: str = '\n') → None

Print text in warning color in console :param message: message which needs to be printed on console. :type message: str :param end: new line :type end: str :return: None :rtype: None

### domaintrails.core.parser.domain\_for\_history(domain: str) → str

This function prepares domain for domain history checking service. :param domain: takes input in any shape of urls e.g. <http://www.example.com>, <https://www.example.com:8080> :type domain: str :return: simplified domain e.g. `example.com` :rtype: str

### domaintrails.core.parser.domain\_for\_md(domain: str) → str

This function prepares domain for domain availability checking service. :param domain: takes input in any shape of urls e.g. <http://www.example.com>, <https://www.example.com:8080> :type domain: str :return: root of domain e.g. `www.example.com` :rtype: str

## domaintrails.core.recon package

### Submodules

#### domaintrails.core.recon.dns\_history module

### class domaintrails.core.recon.dns\_history.DnsHistory(domain: Optional[str] = None)

Bases: `object`

This class is used to collect information from server related to dns history, we are using <http://www.hosterstats.com> service for grabbing records.

#### domain\_history() → list

This function download history from server for given domain name :return: function returns list of records as part of its output :rtype: list

## domaintrails.core.recon.dnslg module

**class** domaintrails.core.recon.dnslg.DnsLG(*domain: Optional[str] = None, get\_authority: bool = False*)  
Bases: `object`

Information: This class collect information about all the dns services currently available for domain. Following is the list of dns classes which are covered as part of this dns, Classes a -> Host Address (A records) cert -> Certificate (CERT records) dhcid -> DHCP Identifier (DHCID records) cname -> Canonical Name (CNAME records) aaaa -> Pv6 Host Address (AAAA records) dlv -> DNSSEC Lookaside Validation record (DLV records) dname -> Delegation name (DNAME records) dnskey -> DNS Key record (DNSKEY records) ds -> Delegation Signer (DS records) hinfo -> Host Information (HINFO records) hip -> Host Identity Protocol (HIP records) kx -> Key eXchanger record (KX records) loc -> Location record (LOC records) mx -> Mail Exchange record (MX records) naptr -> Name Authority Pointer (NAPTR records) ns -> Name Servers (NS records) nsec -> Next-Secure record (NSEC records) nsec3 -> NSEC record version 3 (NSEC3 records) nsec3param-> NSEC3 parameters (NSEC3PARAM records) opt -> Option record (OPT records) talink -> Trust Anchor LINK (TALINK records) tlsa -> TLSA records txt -> Text record (TXT records) ta -> DNSSEC Trust Authorities (TA records) rrsig -> Resource Records Signature (RRSIG records) soa -> Start of Authority (SOA record) spf -> Sender Policy Framework (SPF records) srv -> Service Locator (SRV records) sshfp -> SSH Public Key Fingerprint (SSHFP records) above classes are part of this service.

**download\_report()** → `dict`

This function create different threads by using multiprocessing, and it performs multiple api request from server, response is collected and parsed according usage. This function also slices data as per requirements we are ignoring different classes from api response. :return: a result dict is returned which contains records :rtype: dict

## domaintrails.core.recon.domain\_available module

**class** domaintrails.core.recon.domain\_available.DomainAvailable(*domain: Optional[str] = None*)  
Bases: `object`

This class check if a domain is available or not. We are using mdchecker.com for collecting information from server.

**property domain\_available:** `bool`

This function check domain availability from server, and return result. :return: True/ False / raise exception in case of failure :rtype: bool

## Module contents

### Submodules

## domaintrails.core.resources module

**class** domaintrails.core.resources.UserAgent  
Bases: `object`

Class contains multiple user agent for making different requests on the server.

**static user\_agent()**

this function reruns random user agent from list :return: random user agent from list :rtype: str

**class** domaintrails.core.resources.UserDomain(*domain: Optional[str] = None*)  
Bases: `object`

This class is used to share domain sting with other classes while performing analysis check on domain.

**property domain:** `str`

This is property used to get information about domain :return: domain address :rtype: str

`domaintrails.core.resources.ansi_escape(text: str) → str`

`domaintrails.core.resources.boxing(text: str, style: str = 'double', padding: int = 1, margin: int = 1) → str`

## Module contents

**class** `domaintrails.core.DomainBigDataWhois(domain: Optional[str] = None)`

Bases: `object`

**domain\_whois()**

`domaintrails.core.DomainHostHistory`

alias of `domaintrails.core.recon.dns_history.DnsHistory`

**class** `domaintrails.core.UserAgent`

Bases: `object`

Class contains multiple user agent for making different requests on the server.

**static user\_agent()**

this function reruns random user agent from list :return: random user agent from list :rtype: str

**class** `domaintrails.core.UserDomain(domain: Optional[str] = None)`

Bases: `object`

This class is used to share domain sting with other classes while performing analysis check on domain.

**property domain:** `str`

This is property used to get information about domain :return: domain address :rtype: str

`domaintrails.core.process_request(domain: Optional[str] = None)`

This function is responsible for calling all the configured module and services in project. Following services are called by the function, 1. DomainAvailability 2. DomainHostHistory 3. DomainDnsInformation

Function later calls printing function to display all content

## Submodules

### domaintrails.portal module

`domaintrails.portal.banner()`

banner function is showing ascii art for project name. It contains Domain Trails and tag line for project :return: strings for project banner. :rtype: basestring

`domaintrails.portal.main(domain: Optional[str] = None) → None`

## Module contents

`domaintrails.DomainHostHistory`

alias of `domaintrails.core.recon.dns_history.DnsHistory`

**class** `domaintrails.UserAgent`

Bases: `object`

Class contains multiple user agent for making different requests on the server.

**static** `user_agent()`

this function reruns random user agent from list :return: random user agent from list :rtype: str

**class** `domaintrails.UserDomain`(*domain: Optional[str] = None*)

Bases: `object`

This class is used to share domain sting with other classes while performing analysis check on domain.

**property** `domain: str`

This is property used to get information about domain :return: domain address :rtype: str

`domaintrails.process_request()`

This function is responsible for calling all the configured module and services in project. Following services are called by the function, 1. DomainAvailability 2. DomainHostHistory 3. DomainDnsInformation

Function later calls printing function to display all content



## INDICES AND TABLES

- genindex
- modindex
- search



## PYTHON MODULE INDEX

### d

- domaintrails, 12
- domaintrails.core, 11
- domaintrails.core.parser, 9
- domaintrails.core.parser.html\_parser, 8
- domaintrails.core.recon, 10
- domaintrails.core.recon.dns\_history, 9
- domaintrails.core.recon.dnslg, 10
- domaintrails.core.recon.domain\_available, 10
- domaintrails.core.resources, 10
- domaintrails.portal, 11



## A

`ansi_escape()` (in module `domaintrails.core.resources`), 11

## B

`banner()` (in module `domaintrails.portal`), 11

`boxing()` (in module `domaintrails.core.resources`), 11

## C

`ColorPrint` (class in `domaintrails.core.parser`), 9

## D

`DnsHistory` (class in `domaintrails.core.recon.dns_history`), 9

`DnsHistoryParser` (class in `domaintrails.core.parser.html_parser`), 8

`DnsLG` (class in `domaintrails.core.recon.dnslg`), 10

`domain` (`domaintrails.core.resources.UserDomain` property), 11

`domain` (`domaintrails.UserDomain` property), 12

`domain_available` (`domaintrails.core.recon.domain_available.DomainAvailable` property), 10

`domain_for_history()` (in module `domaintrails.core.parser`), 9

`domain_for_md()` (in module `domaintrails.core.parser`), 9

`domain_history()` (`domaintrails.core.recon.dns_history.DnsHistory` method), 9

`domain_whois()` (`domaintrails.core.DomainBigDataWhois` method), 11

`DomainAvailable` (class in `domaintrails.core.recon.domain_available`), 10

`DomainBigDataWhois` (class in `domaintrails.core`), 11

`DomainHostHistory` (in module `domaintrails`), 12

`DomainHostHistory` (in module `domaintrails.core`), 11

`domaintrails`  
module, 12

`domaintrails.core`

module, 11

`domaintrails.core.parser`  
module, 9

`domaintrails.core.parser.html_parser`  
module, 8

`domaintrails.core.recon`  
module, 10

`domaintrails.core.recon.dns_history`  
module, 9

`domaintrails.core.recon.dnslg`  
module, 10

`domaintrails.core.recon.domain_available`  
module, 10

`domaintrails.core.resources`  
module, 10

`domaintrails.portal`  
module, 11

`download_report()` (`domaintrails.core.recon.dnslg.DnsLG` method), 10

## M

`main()` (in module `domaintrails.portal`), 11

module

`domaintrails`, 12

`domaintrails.core`, 11

`domaintrails.core.parser`, 9

`domaintrails.core.parser.html_parser`, 8

`domaintrails.core.recon`, 10

`domaintrails.core.recon.dns_history`, 9

`domaintrails.core.recon.dnslg`, 10

`domaintrails.core.recon.domain_available`, 10

`domaintrails.core.resources`, 10

`domaintrails.portal`, 11

## P

`parse()` (`domaintrails.core.parser.html_parser.DnsHistoryParser` method), 8

`print_bold()` (`domaintrails.core.parser.ColorPrint` static method), 9

`print_fail()` (*domaintrails.core.parser.ColorPrint static method*), 9  
`print_info()` (*domaintrails.core.parser.ColorPrint static method*), 9  
`print_pass()` (*domaintrails.core.parser.ColorPrint static method*), 9  
`print_warn()` (*domaintrails.core.parser.ColorPrint static method*), 9  
`process_request()` (*in module domaintrails*), 12  
`process_request()` (*in module domaintrails.core*), 11

## U

`user_agent()` (*domaintrails.core.resources.UserAgent static method*), 10  
`user_agent()` (*domaintrails.core.UserAgent static method*), 11  
`user_agent()` (*domaintrails.UserAgent static method*), 12  
`UserAgent` (*class in domaintrails*), 12  
`UserAgent` (*class in domaintrails.core*), 11  
`UserAgent` (*class in domaintrails.core.resources*), 10  
`UserDomain` (*class in domaintrails*), 12  
`UserDomain` (*class in domaintrails.core*), 11  
`UserDomain` (*class in domaintrails.core.resources*), 10