On Collecting Onion Server Fingerprints and Identification of Their Operators

Yuuki Kimura[1], Soramichi Akiyama[2], Atsuo Inomata[3], and Tetsutaro Uehara[2*]
[1]Graduate School of Information Science and Engineering, Ritsumeikan University, Kusatsu–shi, Shiga, Japan
[2]College of Information Science and Engineering, Ritsumeikan University, Kusatsu–shi, Shiga, Japan
[3]Research Organization of Science and Technology, Ritsumeikan University, Kusatsu–shi, Shiga, Japan
ykimura@cysec.cs.ritsumei.ac.jp, s-akym@fc.ritsumei.ac.jp, inomata.atsuo.cysec@osaka-u.ac.jp, t-uehara@fc.ritsumei.ac.jp
*corresponding author

Abstract—Anonymous network technology exists to ensure anonymous communication. Among the many anonymous networks that have been developed, Tor (The Onion Router) is the most well-known and has the largest usage share. Tor can anonymize not only the client but also the server, which is a function called Onion Service. Although Onion Service makes it possible to operate websites while keeping IP addresses secret, it is sometimes used to post illegal content. To identify the operators of Onion Services containing illegal content, we propose combining and comparing fingerprints obtained through multiple methods. In addition, we also report on the rate and examples of information exposure by fingerprints. In this research, we collected approximately 40,000 Onion Domains and analyzed their fingerprints. From the results, we evaluate the possibility of identifying the operators of Onion Services.

Keywords–Tor; Dark Web; Onion Service

1. Introduction

Onion Service [1] [2] is a method of managing a web service while keeping its IP address secret, allowing web service operators to keep their IP addresses private. Therefore, a website set up using Onion Service (referred to as *Onion Site*) can be used to buy and sell goods and post content that is illegal in many countries. It can be used for additional nefarious purposes, such as operating a website to leak information stolen by ransomware. These illegal activities have become a world-wide problem and judicial authorities in various countries have attempted to identify the operators of Onion Sites and shut them down. The closure of illegal Onion Sites has achieved some success through large-scale efforts.

The first challenge in the effort to close illegal Onion Sites is to identify the operators. However, these operators are behind the wall of Tor's anonymity, which is, by design, difficult to break. This mandates a reliable workaround method to identify the operator of an Onion Site. However, a reliable method for identifying the operator of an Onion Site has not been established and it remains difficult, as mentioned, to directly break the anonymity of Tor itself. On the other hand, IP addresses or other unique information may be exposed due to improper configuration or the vulnerabilities of servers [3]. In addition, server fingerprints are known to vary from server to server, even for Onion Sites [4]. To date, no research has combined these two approaches. Therefore, in this research, we propose combining and comparing fingerprints obtained by multiple methods.

The second challenge in the effort to close illegal Onion Sites is to collect onion domains. By its very nature, Tor does not have an exhaustive search engine like Google, making it difficult to understand the Onion Site itself. Therefore, it is necessary to collect Onion Site domains (referred to as *Onion Domains*). In this research, since Onion Sites are used by people, we assume that functioning Onion Sites are always linked to other Onion Sites by links and collect Onion Domains by crawling Onion Sites.

Therefore, we first conducted an extensive collection of Onion Domains. Then, to identify the source of operator of Onion Sites, we examined the fingerprint that can be obtained from the Onion Site with the aim of exposing IP addresses by exploiting inappropriate settings. In the domain collection, about 40,000 Onion Domains were collected and valid responses were obtained from about 90%. As a result of trying to identify the operation source using multiple attack techniques, we confirmed the exposure of information leading to an IP address or IP address identification at multiple Onion Sites.

2. Research Background

2.1. Dark Web

The dark web is a collection of anonymous websites built on the Internet with an overlay network that requires specific software for browsing. In general, a server cannot identify client information when a client accesses the dark web. Therefore, the characteristics of the dark web can be exploited to post illegal content. In addition, Kaur's report [5] indicates that 6% of web content on the Internet is on the dark web. One notable software that enables the dark web is Tor.

2.1.1. Tor: Tor is a standard for realizing anonymous communication over TCP/IP and it is also used as the name of software and networks that implement the standard. While Tor is able to offer a client side IP address hidden feature by using the Onion Routing, on the server side, it can also provide the capability of hidden the IP address of the server side by using the Onion Service.

2.1.2. Onion Service: Onion Service is a Tor service that provides TCP services while keeping the IP address of the server secret. Onion Service ends with ".onion" and has a pseudo-domain of 56 characters encoded in Base 32 [6] (e.g., abcdefghijklmnopqrstuvwxyz234567abcdefghijklmnopqrstuvwx.onion). Services

provided on Onion Service can only be accessed via the Tor network. By using Onion Service, server operators can provide TCP services only to Tor users without revealing the IP address of the server.

Onion Sites can be used for illicit purposes, such as buying and selling goods, posting content that is illegal in many countries, or operating a website to leak information that was stolen by ransomware. Sites that deal with illegal goods are called "dark markets", a prominent one of which was "Silk Road". "Silk Road" was a hotbed of crime, with $1.2 billion US worth of illegal drugs and firearms traded over two and a half years between 2011 and 2013 [7] [8]. In addition, more than 30,000 child pornography images were posted on "PlayPen" [9] [10]. Illegal sites such as "Silk Road" and "PlayPen" are regarded as world-wide problems and attempts to shut down illegal Onion Sites have been made by judicial bodies around the world. Notably, "Silk Road" and "PlayPen" were shut down by the FBI.

However, to close an Onion Site, it is first necessary to identify the source of the Onion Site's operation; no reliable method has been established for this identification. It is also difficult– by design–to directly break the anonymity of Onion Service.

2.2. Server Fingerprint

Server fingerprints that can uniquely identify an IP address or the server from which a web site operates may be exposed due to improper configuration or server vulnerabilities. A server fingerprint refers to publicly available information, such as server software version or type, that does not identify a server by itself but can identify a server by combining multiple pieces of the server fingerprint. As an example, the OWASP Testing Guide 4.0 [11] shows that the order of the HTTP response headers varies depending on the web server software. In this research, information that can be used to identify the source of the server operation, such as the contents of the HTTP response headers, their order, and error messages, is collectively treated as a server fingerprint.

2.3. Related Research

Wang et al. [12] focused on the characteristics of TCP packets and proposed a method to identify an onion site being accessed by local users. Their aim was to identify which Onion Site a user is visiting by collecting packets from when the user is visiting the site and comparing them to a log of pre-collected packets. Their proposal uses the fingerprint to attack anonymity but, unlike our research, it does not directly break the anonymity of the Onion Service. We aim to use the fingerprint to attack anonymity and identify the Onion Site operators.

Cernica [3] presented several techniques for deanonymizing and identifying the operators of an Onion Site. However, they did not show to what extent deanonymization actually succeeds. Therefore, in this research, we conducted experiments using the method originally introduced by Cernica.

Arai et al. [4] [13] collected about 6,000 Onion Domains and classified Onion Sites by focusing on the HTTP header information. They crawled Onion Sites in July 2019, obtained and analyzed the HTTP response headers, and claimed to have obtained 4,340 Onion Domains, which is 94% of all Onion Domains that they had collected. However, the Tor Project reports [14] that there were 72,204 Onion Domains on July 14, 2019 when the experiment was conducted. Therefore, their collection numbers do not cover the entire Onion Service. In this research, we aim to collect Onion Domains on a larger scale by crawling the URLs of Onion Sites using parallel processing.

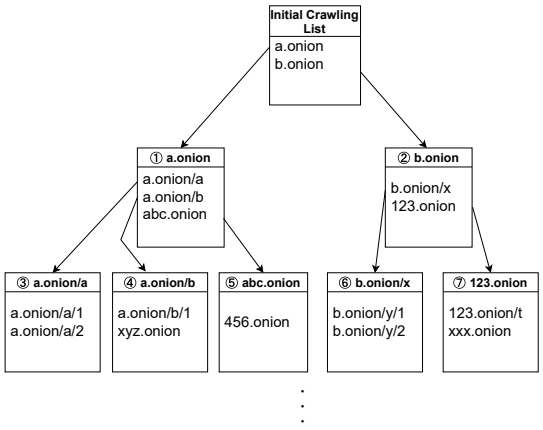

Figure 1. Crawling order of breadth-first search for Onion Sites

Klijnsma [15] points out that, on a server running an Onion Service, if the server is listening to anything other than 127.0.0.1, IP addresses of the server may be exposed due to secure sockets layer (SSL) certificate matching. In addition, a report by Talos Japan [16] shows an example of exposing IP addresses by applying a method similar to Klijnsma's method to favicons. Therefore, in this research, we examine the possibility of IP address exposure by utilizing and applying previous research.

3. Survey Method

In this research, the investigation is divided into the collection of Onion Domains and the analysis of the collected Onion Domains. In this Section, we explain the survey methods separately for collection and analysis.

3.1. Collecting Onion Domains

To collect an Onion Domain, we crawl the URL of the Onion Site. We create an initial crawling list and each URL is accessed by a breadth-first searchs from the initial crawling list to obtain an HTTP response body. Then, we collect the URL containing ".onion" from the response body. We also collect and record the Onion Domain containing ".onion" from some collected URLs. The collected URL is accessed and these procedures are repeated. The order in which crawls are performed is shown in Figure 1.

In order to speed up the crawling, we parallelize requests to the Onion Site URL. The crawler crawls at most 100 threads in parallel. The implementation is carried out using

TABLE I
SITES FOR SEARCHING ONION SITES

Site name	Onion Domain
Torch	torchdeedp3i2jigzjdmfpn5ttjhthh5wbmda2rr3jvqjg5p77c54dqd.onion
Torch	4rfotrat64q6lssi4ztqbmlbtan6edgego4wa2idd7tl7pc4d5nkfwqd.onion
ourrealm	orealmvxooetglfeguv2vp65a3rig2baq2ljc7jxxs4hsqsrcemkxcad.onion
Ahmia	juhanurmihxlp77nkq76byazcldy2hlmovfu2epvl5ankdibsot4csyd.onion
visitor	uzowkytjk4da724gizttttfly4rugfnbqkexecotfp5wjc2uhpykrpryd.onion
tor66	tor66sewebgixwhcqfnp5inzp5x5uohhdy3kvtnyfxc2e5mxiuh34iid.onion

```
GET / HTTP/1.0
Connection:Close
```

Figure 2. HTTP request for HTTP 1.0 Attack

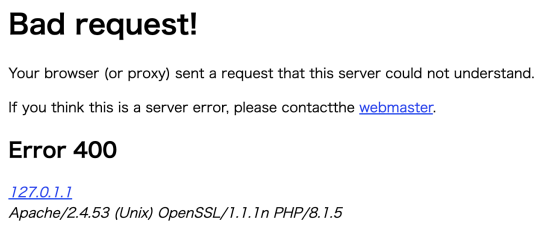

Figure 3. Example of IP address exposure in NoHost Requests Attack

Go language and the collected information is recorded in the database server.

In the initial crawling list of crawlers, we used search result pages of prominent onion site-search sites. The Onion Sites used are shown in Table I.

3.2. Analysis of Onion Domains

The following three methods are used to analyze the Onion Domains collected by the methods described in the previous section.

3.2.1. HTTP 1.0 Attack: In an environment where Virtual Hosting is configured, if a request is made with HTTP 1.0 for which the Host specification is not defined, the web server software may return unintended content. The method that compares returned content with other information using the above features to estimate the origin of the Onion Site is referred to as an HTTP 1.0 Attack in this research.

An example of an HTTP request for which an HTTP 1.0 Attack is successful is shown in Figure 2.

3.2.2. NoHost Requests Attack: In the Apache HTTP Server, when a GET request is made without specifying a Host, the IP address may be exposed on the error screen. An example of exposing an IP address is shown in Figure 3. However, Figure 3 example is when the local loopback address (127.0.0.0/8) is exposed.

In this research, we call the attack method that uses the above features to estimate the origin of an Onion Site from its exposed IP address the NoHost Requests Attack.

```
GET / HTTP/1.1
Host:
Accept: */*
```

Figure 4. HTTP request for NoHost Requests Attack

TABLE II
RESULTS OF COLLECTING ONION DOMAINS

Period of investigation	2023/01/21-2023/01/30（10 days）
URLs discovered	278,075,522
URLs investigated	14,363,466
Onion Domains discovered	39,118

An example of an HTTP request in which a NoHost Requests Attack is successful is shown in Figure 4.

3.3. Analysis of the Server Fingerprint

In an HTTP response, the HTTP response header holds additional information about the response. The HTTP response header can add any header field on the server side. Therefore, a server's own header may be added, which may lead to the identification of the server. Thus, in this research, we collect header names with infrequent occurrences and their header field values.

We also collect the number of headers because the number of headers varies from server to server.

4. FINDINGS

In this section, we describe the survey results separately for collection and analysis.

4.1. Onion Domain Collection Results

The results of the collection based on the proposed method in Section 3-A are shown in Table II.

According to the Tor Project report [14], the number of Onion Domains was 760,899 on average from January 21, 2023 to January 30, 2023. Therefore, in this research, we can say that we collected 5.14% of all Onion Domains.

4.2. Results of Onion Domain Analysis

The research and analysis results for each analysis method indicated in Section 3.2 are shown below.

4.2.1. HTTP 1.0 Attack: The HTTP 1.0 Attack was performed on all collected domains, resulting in 29,789 responses. The top 5 most frequently occurring HTTP response status codes from the responses obtained by performing the HTTP 1.0 Attack are shown in Table III.

Among the response status codes obtained by the HTTP 1.0 Attack, 5,873 were "200 OK". For the 5,873 responses that resulted in "200 OK", differences from normal responses were visually confirmed and classified into the following four patterns.

1) Displaying content identical to the normal response

TABLE III
FREQUENT HTTP RESPONSE STATUS CODES IN HTTP 1.0 ATTACK RESPONSES

Response code	Number of cases
400	22,997
200	5,873
403	278
301	252
302	145

TABLE IV
HTTP 1.0 ATTACK RESPONSE CLASSIFICATION

Pattern	Number of cases
(1)	4,685
(2)	869
(3)	60
(4)	259

TABLE V
IP ADDRESS EXPOSURE RESULTS OF NOHOST REQUESTS ATTACK

	Number of cases
No Exposure	1,908
Local loopback address	76
Local IP address	7
Global IP address	5

2) Displaying a default page, for example of the web server software, that is different from the normal response
3) Displaying an error page, for example of the web server software, that is different from the normal response
4) Displaying content that is different from the normal response and is neither a default page nor an error page

The number of cases of each pattern are shown in Table IV. According to Table IV, there were 259 cases where the content was different from the normal response and was neither the default page nor the error page. As a result, we can confirm that there were real cases where the web server returned content that was not the Onion Site specified in the HTTP 1.0 Attack request.

4.2.2. NoHost Requests Attack: For the collected domains, the survey was limited to domains that were found to use the Apache HTTP Server for their web server software according to the survey results in the Section 4.2.1. 1,996 domains were surveyed. The results of exposed IP addresses are shown in Table V.

From Table V, we can confirm that the exposure of IP addresses can lead to the identification of the operator of the Onion Site. The assigned countries of the exposed IP addresses were surveyed using GeoIP2 [17]. These were the People's Republic of China, the Republic of Seychelles, Canada, the Republic of Bulgaria, and Japan, respectively.

4.2.3. Server Fingerprint: GET requests were made to 39,118 collected Onion Domains, resulting in 36,079 HTTP responses. This section analyzes the HTTP response headers among the HTTP responses. We do not distinguish between response headers and entity headers in this section.

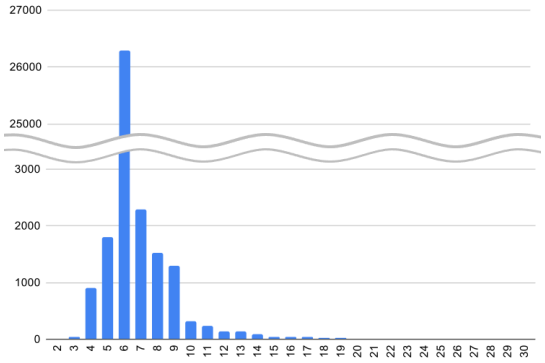

Figure 5. Distribution of response header lines

TABLE VI
TOP 10 RESPONSE HEADER OCCURRENCES

Header name	Number of occurrences
Server	36,046
Content-Type	36,039
Transfer-Encoding	35,512
Connection	33,023
Cache-Control	30,872
Date	28,754
Set-Cookie	5,184
Vary	4,839
Content-Length	3,261
Last-Modified	3,100

The distribution of the number of response header lines included in each HTTP response is shown in Figure 5. The x axis shows the number of response header lines and the y axis shows the number of Onion Domains.

The number of lines in the response header was highest in the HTTP response with six lines, followed by seven, five, eight, and nine lines. The minimum number of response header lines was 2 and the maximum was 30.

We next describe the frequency of each header's appearance. As a result of investigation, 206 kinds of response headers were obtained. The top 10 types of HTTP response headers in terms of the number of occurrences are shown in Table VI.

Among the collected response headers, 86 types of response headers appeared only once in the domains investigated in this research. For each of these response headers, the distribution of the number of IP addresses obtained as a result of searching with Shodan [18] is shown in Figure 6. Shodan indexes the banner information, such as HTTP response headers, of Internet IP addresses. The x axis shows the number of IP addresses obtained by the Shodan search and the y axis shows the number of types of response headers to which the search was applied.

As shown in Figure 6, among the obtained response headers,

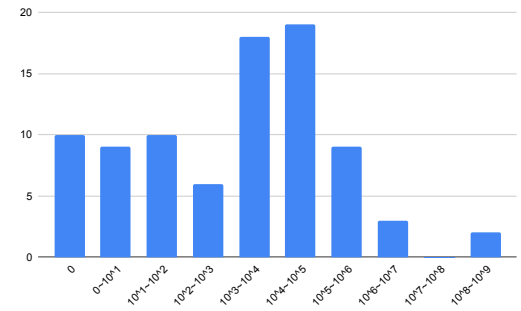

Figure 6. Results in Shodan for response headers used only once

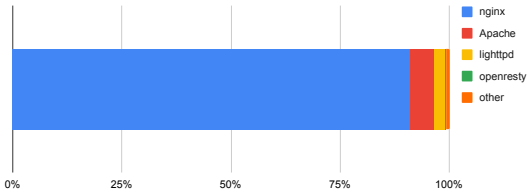

Figure 7. Web server software information contained in the server header

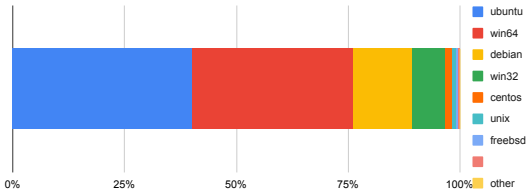

Figure 8. OS information contained in the server header

some response headers that appeared only once appeared less frequently even on the Surface Web. On the other hand, in some cases there are more than 1000, which does not necessarily lead to the identification of the Onion Site operator.

Next, we describe the contents of the server header that appeared most frequently among the obtained response headers. We received 36,046 server header responses. The server header describes the software used by the server that generated the response. In this survey, 1,306 server headers included the name of the operating system and 35,275 included the name of the web server software. The names of the operating systems and web server software included in the server header are shown in Figure 7 and 8, respectively.

From Figure 7, nginx accounts for 90% of the web server software running Onion Sites, followed by Apache and lighttpd. Another significant feature of the web server operating the Onion Site is that, while UNIX-based operating systems account for 60% of the OS, 64-bit Windows accounts for 35%, as shown in Figure 8.

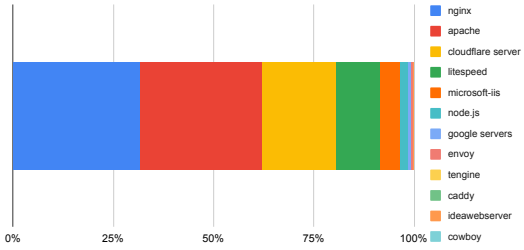

Figure 9. Share of web server software on the Surface Web [21]

5. Discussion

5.1. Evaluation of the Results of the Onion Domain Collection

In this research, we collected 39,118 Onion Domains. As described in 4.1, this is estimated to be 5.14% of all Onion Domains. On the other hand, Aoki et al. [19] estimates that the number of Onion Services are between 14,509 and 96,034. It also estimates that there are an average of 40,848 Onion Services in the two-year period from June 2018 to December 2020. This is about 20%˜50% of the value of the Tor Project reports [14]. Furthermore, Cilleruelo et al. [20] state that it is difficult to collect all Onion Domains in a given moment in the Tor network. This is because not all Onion Domains are running http servers, and domains can go offline at any time. Therefore, there are various arguments for estimating the size of the number of Onion Services at a given instant. Similarly, the calculation of coverage is not certain.

In addition, in this study, the collection of Onion Domains by crawling only collects Onion Domains that are connected by hyperlink, and it is difficult to collect all Onion Domain that are not linked to any other Onion site. As a solution, it is necessary to further parallelize and speed up the crawlers.

However, due to the nature of hyperlink crawling, it is difficult to cover everything. Therefore, it is necessary to generate a completely random string as an Onion Domain and try to access it, or try other new ideas.

5.2. Evaluation of the Results of the Onion Domain Analysis

In this section, we discuss the implications of the results of each survey described in Section 4.

In this research, the Onion Domain analysis shows that nginx accounts for more than 90% of the web server software in the Onion Service. On the other hand, on the Surface Web, nginx's share is said to be around 30% [21]. The share of web server software on the Surface Web is shown in Figure 9.

One possible reason for the larger share of nginx on the dark web compared to the Surface Web is the use of the same copy of the boilerplate server image and the same hosting service. In addition, a possible factor that accounts for more than 95 % of the share in the Apache HTTP Server and nginx is the Tor Project's "Set Up Your Onion Service". The Apache HTTP Server and nginx are the examples introduced in the installation guide [22].

Focusing on the OS, in contrast to previous studies, which showed Windows holding about 3% among Onion Sites, the

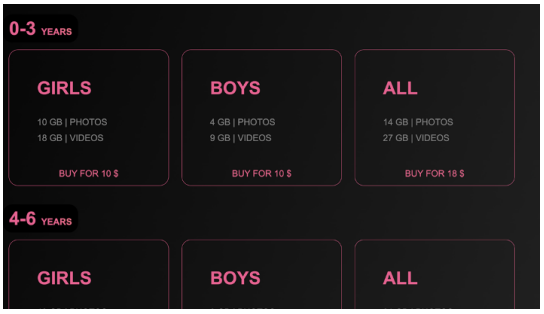

Figure 10. “ChildrenXXX”

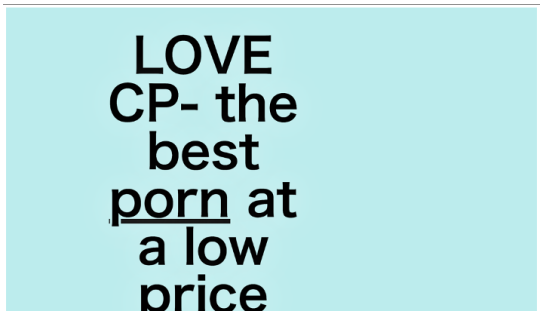

Figure 11. “Love CP”

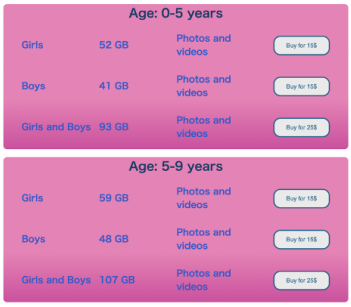

Figure 12. HTML content returned by “ChildrenXXX” and “Love CP” in HTTP 1.0 Attack

results show Windows holding a larger share. However, a closer look at 32-bit Windows in particular shows that of the 96 Onion Sites that returned a server header containing the string “Win 32”, 74 completely matched each other both in the contents of the server header and the response body of the HTTP 1.0 Attack. From this, it is highly likely that these 74 Onion Sites are operated by the same server or the same group of operators. Next is a more concrete example.

Figures 10 and 11 are the top pages of the Onion Sites titled “ChildrenXXX” and “Love CP”, respectively.

Both sites in Figure 10 and 11 are Onion Sites containing child pornography. However, each site has different content. As a result of the HTTP 1.0 Attack, both sites returned the same HTML content. The HTML content returned by both sites is shown in Figure 12.

The response headers of the sites of “ChildrenXXX” and “Love CP” are shown in Figure 13 and 14, respectively.

```
Server : Apache/2.4.38 (Win32) PHP/7.1.26
X-Powered-By : PHP/7.1.26
Transfer-Encoding : chunked
Content-Type : text/html; charset=UTF-8
```

Figure 13. A response header of “ChildrenXXX”

```
Server : Apache/2.4.38 (Win32) PHP/7.1.26
ETag : "2804-5ece4a13b2eb9"
Accept-Ranges : bytes
Content-Length : 10244
Content-Type : text/html
```

Figure 14. A response header of “Love CP”

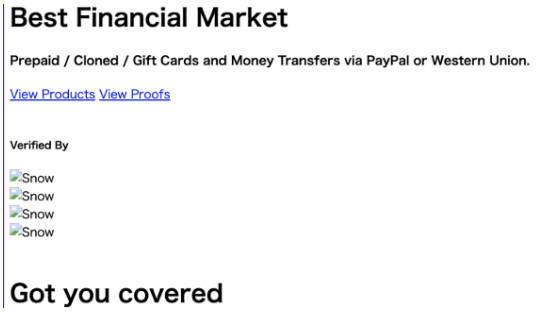

Figure 15. “Cardzilla - Best Financial Market”

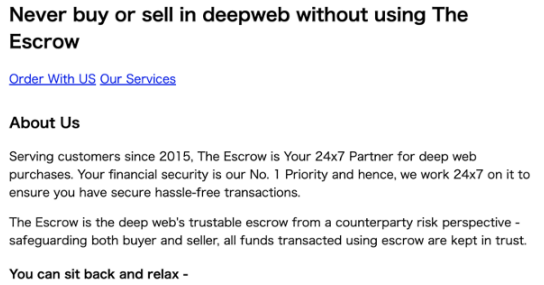

Figure 16. “The Escrow - Dark Web Escrow Service”

Of the response headers in Figure 13 and 14, the server header shows that both are “Apache/2.4.38 (Win 32) PHP/7.1.26”. Since both of the similarities described above are considered to be rare similarities, it is conceivable that the sites in Figure 10 and 11 are operated by the same operator or server.

Below are further examples of identifying the operator of an Onion Site by combining multiple fingerprints.

Figures 15 and 16 are the top pages of the Onion Sites “Cardzilla - Best Financial Market” and “The Escrow - Dark Web Escrow Service,” respectively. “Cardzilla” is a site that sells credit card information and “The Escrow” is a site that provides escrow services.

Figure 17 shows the response header of “Cardzilla - Best Financial Market”.

```
Server : nginx
Content-Type : text/html; charset=UTF-8
Transfer-Encoding : chunked
Connection : keep-alive
Keep-Alive : timeout=60
Vary : Accept-Encoding
X-Powered-By : PHP/7.4.5
```

Figure 17. A response header of “Cardzilla - Best Financial Market”

Figure 18. HTML content returned by “Cardzilla - Best Financial Market” and “The Escrow - Dark Web Escrow Service” in the HTTP 1.0 Attack

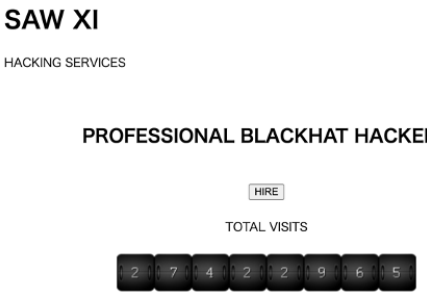

Figure 19. “SAW XI”

In this case, the contents of the response header of “Cardzilla” and the response header of “Escrow” were exactly same, as was the response body of the HTTP 1.0 Attack. The HTML content returned by both sites is shown in Figure 18.

In the example of Figure 15 and 16, the contents and order of the server header are exactly the same and the results of the HTTP 1.0 Attack are the same, so it appears that they are operated by the same operator or server.

Finally, we show an example of Onion Site with a slightly unusual similarity.

Figures 19 and 20 are the top pages of the Onion Sites “SAW XI” and “CLAY,” respectively. “SAW XI” and “CLAY” both claim to be hacking services.

Furthermore, Figure 21 shows a screenshot of the bottom of the top page of “CLAY”.

Figures 22 and 23 show the response headers of the “SAW XI” and “CLAY” Onion Sites, respectively.

In this case, the contents of the server headers of “SAW XI”

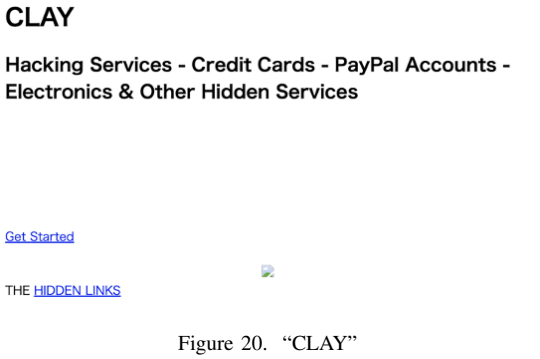

Figure 20. “CLAY”

Figure 21. The bottom of the top page of “CLAY”

```
Server : Apache/2.4.51 (Unix)
OpenSSL/1.1.1l
PHP/8.0.11 mod_perl/2.0.11 Perl/v5.32.1
X-Powered-By : PHP/8.0.11
Content-Length : 1703
Content-Type : text/html; charset=UTF-8
```

Figure 22. A response header of “SAW XI”

```
Server : Apache/2.4.48 (Unix)
OpenSSL/1.1.1k
PHP/7.3.30 mod_perl/2.0.11 Perl/v5.32.1
Last-Modified : Sun, 01 Jan 2023 04:13:25
GMT
ETag : "2186-5f12c0e5931d8"
Accept-Ranges : bytes
Content-Length : 8582
Content-Type : text/html
```

Figure 23. A response header of “CLAY”

and “CLAY” differ. However, they have a common point in that “mod_perl/2.0.11 Perl/v5.32.1” is included. Furthermore, looking at Figure 19 and 21, we note that both sites use the same access counter. In this way, we see that there is some common points for the operator of these Onion Sites even in the example of Figure 19 and 20.

We can therefore say that similarities between Onion Sites can be investigated by using multiple survey results as a server fingerprint.

In this way, it is possible to identify the server-by-server fingerprint technology when web server software other than nginx is used, when OS names other than ubuntu, 64-bit Windows, and Debian are specified in the server header, when a unique response header is provided, or when a response other than the Onion Site specified in the request is exposed. For these Onion Sites, if a web site with a similar fingerprint can be found on the dark web, it is possible to show the possibility of multiple Onion Sites being operated by the same operator. In addition, the fingerprint can be used in conjunction with other methods to find more consistent Onion Sites. Furthermore, if a web site with a similar fingerprint can be found on the Surface Web, it will lead to the identification of the source of the Onion Site operation. Additionally, this research was able to show an example of IP address exposure by the NoHost Requests Attack. However, it is not certain whether the server operator intended for the IP addresses exposed by the NoHost Requests Attack to be exposed or not, so we must pay attention to this evaluation.

6. Conclusion

In this research, we examined the possibility of identifying the source of an Onion Service and collected Onion Domains for this purpose. In the phase of Onion Domain collection, the crawler was implemented by parallel processing using Go language and the number of Onion Domains collected successfully exceeded the previous research. We also showed that, by using the server fingerprint, it is possible to identify the origin of an Onion Sites without directly breaking Tor’s encryption. Server fingerprint identification is an effective means for identifying operators who operate Onion Sites with malicious intent. On the other hand, from the operator’s point of view, in order to ensure the secrecy of onion site operator information, using OS and server software that many operators have adopted is recommended. It is important not to add information to the response header and to ensure error handling.

As a challenge, there are currently about 700,000 Onion Domains and, since we can only access IP address identification of a small number of Onion Sites at this time, further research and development is required for both collection and analysis. In the future, we will speed up the Onion Domain collection, devise and implement new methods for analysis, and aim to improve the number of operators of Onion Services. In addition, in order to identify the source of an onion site operation, it is necessary to examine a unique server fingerprint that cannot be concealed by the mechanism.

References

[1] “Tor project — anonymity online,” https://www.torproject.org/, (Accessed on 01/14/2023).

[2] “Tor project — onion services,” https://community.torproject.org/onion-services/, (Accessed on 02/05/2023).

[3] I. Cernica, “Deanonymization of tor http hidden services - def con forums,” https://media.defcon.org/DEF%20CON%2030/DEF%20CON%2030%20presentations/Ionut%20Cernica%20-%20Deanonymization%20of%20TOR%20HTTP%20hidden%20services.pdf, (Accessed on 08/05/2023).

[4] Y. Arai, K. Yoshioka, and T. Matsumoto, “A study on the features of middleware for illegal goods trading sites in the tor hidden service,” in *Computer Security Symposium*, vol. 2019, oct 2019, pp. 482–487.

[5] K. Raminder, “Why dark web threat intelligence feed is a key component of siem,” https://www.linkedin.com/pulse/why-dark-web-threat-intelligence-feed-key-component-siem-cetarkcorp, (Accessed on 08/03/2023).

[6] “Rfc 4648: The base16, base32, and base64 data encodings,” https://www.rfc-editor.org/rfc/rfc4648, (Accessed on 08/03/2023).

[7] A. Gayathri, “From marijuana to lsd, now illegal drugs delivered on your doorstep,” https://www.ibtimes.com/marijuana-lsd-now-illegal-drugs-delivered-your-doorstep-290021, (Accessed on 08/03/2023).

[8] A. Chen, “Now you can buy guns on the online underground marketplace,” https://www.gawker.com/5879924/now-you-can-buy-guns-on-the-online-underground-marketplace, (Accessed on 08/03/2023).

[9] C. Joseph, “The fbi’s ’unprecedented’ hacking campaign targeted over a thousand computers,” https://www.vice.com/en/article/qkj8vv/the-fbis-unprecedented-hacking-campaign-targeted-over-a-thousand-computers, (Accessed on 08/03/2023).

[10] “ ‘ playpen ’ creator sentenced to 30 years — fbi,” https://www.fbi.gov/news/stories/playpen-creator-sentenced-to-30-years, (Accessed on 07/31/2023).

[11] E. Saad, M. Meucci, and R. Mitchell, “Owasp testing guide, v4,” *OWASP Foundation*, vol. 4, pp. 66–80, 2013.

[12] T. Wang and I. Goldberg, “Improved website fingerprinting on tor,” in *Proceedings of the 12th ACM workshop on Workshop on privacy in the electronic society*, 2013, pp. 201–212.

[13] Y. Arai, K. Yoshioka, and T. Matsumoto, “Automatic illegal-goods-handling site detection by using http headers,” *IPSJ Journal*, vol. 61, no. 9, pp. 1388–1396, sep 2020.

[14] “Onion services – tor metrics,” https://metrics.torproject.org/hidserv-dir-onions-seen.html, (Accessed on 08/03/2023).

[15] L. Abrams, “Public ip addresses of tor sites exposed via ssl certificates,” https://www.bleepingcomputer.com/news/security/public-ip-addresses-of-tor-sites-exposed-via-ssl-certificates/, (Accessed on 08/03/2023).
[16] E. Paul, “De-anonymizing ransomware domains on the dark web,” https://blog.talosintelligence.com/de-anonymizing-ransomware-domains-on/, (Accessed on 08/03/2023).
[17] “Geoip2 databases demo — maxmind,” https://www.maxmind.com/en/geoip-demo, (Accessed on 08/03/2023).
[18] “Shodan search engine,” https://www.shodan.io/, (Accessed on 08/03/2023).
[19] T. Aoki and A. Goto, “Graph visualization of dark web hyperlinks and their feature analysis,” *International Journal of Networking and Computing*, vol. 11, no. 2, pp. 354–382, 2021. [Online]. Available: http://www.ijnc.org/index.php/ijnc/article/view/259
[20] C. Cilleruelo, L. de Marcos, J. Junquera-Sánchez, and J.-J. Martínez-Herráiz, “Interconnection between darknets,” *IEEE Internet Computing*, vol. 25, no. 3, pp. 61–70, 2021.
[21] “Historical trends in the usage statistics of web servers, august 2023,” https://w3techs.com/technologies/history_overview/web_server, (Accessed on 08/03/2023).
[22] “Tor project — set up your onion service,” https://community.torproject.org/onion-services/setup/, (Accessed on 08/03/2023).