Bilgisayar Bilimleri ve
Mühendisliği Dergisi

Cilt-16
Sayı-2
2023

A Study on the nDPI Deep Packet Inspection Tool A Study on the nDPI Deep Packet Inspection Tool

Zehra Nur ÖZBAY
Ege University
International Computer Institute
Izmir, Turkey
zehra.nur.ozbay@ege.edu.tr
ORCID: 0000-0001-5680-1227

Mehmet Emin DALKILIÇ
Ege University
International Computer Institute
Izmir, Turkey
mehmet.emin.dalkilic@ege.edu.tr
ORCID: 0000-0003-3932-5155

Abstract

Deep packet inspection (DPI) is an advanced packet identification method used to detect application protocols in the network by analyzing network packets up to the application layer in the five-layer network model. This study focuses on the open source nDPI library that performs packet identification using the DPI method. It is aimed to contribute to the field by detecting new application protocols and adding some missing protocols through this library. In addition, new rule definitions for network packets that were found to be miscategorized by nDPI were defined to correct such situations. For all these, network traffic was recorded with the Wireshark packet capture tool, packet contents were analyzed to create new rules and new application protocols were added to the nDPI library. Finally, a partial automation of the application protocol detection in nDPI was written.

Keywords: Deep packet inspection, DPI, nDPI

Abstract

Deep packet inspection (DPI) is an advanced packet identification method used to detect application protocols found in the network by analyzing network packets up to the application layer in the five-layer network model. In this study, the open source nDPI library, which uses DPI method to identify packets, is discussed. It is aimed to contribute to the field by detecting new application protocols and adding some missing protocols through this library. In addition, new rule definitions were made for network packets that were found to be miscategorized by nDPI, and such cases were corrected. For all these, network traffic was recorded with Wireshark packet capture tool, packet contents were analyzed, new rules were created and newly found application protocols were added to the nDPI library. Finally, a partial automation of the work done for the application protocol detection in nDPI was written.

Submission, correction and acceptance date: 20.02.2023 - 29.03.2023 - 20.05.2023

Type of article Research

Keywords*:* Deep packet inspection, DPI, nDPI

1. Introduction

Every day a new application takes its place in the network and the number of protocols that need to be defined and classified increases. In addition, the number of internet-connected devices is reaching enormous levels with the Internet of Things (IoT). By 2030, it is estimated that approximately 30 billion IoT devices will be in use [1]. This uncontrolled increase in the number of devices and applications brings along network security problems. Firewalls, Intrusion Detection Systems (IDS) and Intrusion Prevention Systems (IPS) are frequently preferred and used network security devices to counter network attacks. Firewalls, which are usually positioned at the border areas between the internal network and the external network, check the header information of network packets moving from the internal network to the external network or vice versa, giving the right of passage to packets that comply with the user's predefined rules, while dropping those that do not comply. Firewalls are used for classic packet filtering at network entrances and exits. However, infiltration through this filter is possible because it cannot detect "fine-grained" attack packets. Another network security device, the IDS, operates in the internal network by means of listening sensors. Using its signature database, the IDS detects suspicious and irregular traffic on the internal network and reports it to the network administrator. IPS, which is usually positioned right behind firewalls, takes an active role in the network, unlike the passive function of IDS on packets.

It is an extension of IDS devices, with the ability to intercept and drop packets. All these devices are undoubtedly the main elements of network security. However, when these devices are considered alone, their effectiveness against new types of network attacks will be low. For example, firewalls have limited processing power and cannot handle large volumes of packets. Moreover, even if a firewall can keep stateful information, it cannot see the information at the application layer, which is the last layer of the five-layer network model and contains the actual data. This is similar to the way we evaluate a book by reading only the title without being aware of the content [2]. DPI, on the other hand, decrypts network traffic and allows inspection at all layers, including the application layer, and thus reads the data contained in the packet, if it is not encrypted. On the other hand, attackers can break the packets they send to the target network into smaller pieces and prevent the attack signature from being contained in a single packet, making it impossible to detect by IDS. In DPI analysis, even if the attack signature is broken into many packets and the trace is attempted to be erased, these packets are put together and the signature is detected. DPI technology has therefore become an essential component of network security devices. The use of a DPI tool or library can be in conjunction with packet capture tools that eavesdrop on the network, or more commonly in the form of DPI-defined next generation firewall (NGFW) or IDS/IPS devices.

A review of publications shows that there are studies that use deep packet inspection for both application protocol detection and application blocking. In the study by Renals and Jacoby [3], it is shown how to block Skype, a P2P messaging application, using DPI. In the study, a rule set was created to identify Skype packets using the features that distinguish Skype from other applications. To build this rule set, we first collected data on different computers, at different times, and with different Skype accounts and versions, using the Wireshark network sniffing tool to account for all possible cases. The Skype traffic data was compared using an analysis program called Araxis Merge, which can compare up to three different sessions of traffic at the same time. This analysis resulted in three categories of Skype rules: keywords, port numbers and content. Keywords identify packets containing specific strings such as "Skype", port numbers identify TCP traffic with port 33033, and content identifies packets containing the ASCII strings "16 03 01 00 00 00" or "17 03 01 00 00 00". These rules are defined in the open source Snort, an IPS appliance with DPI defined in it. After applying these rules, Wireshark re-records the traffic and observes Skype's behavior. It is detected that Skype tries to establish a connection and uses new strings. Finally, these new strings are added to the rule table and when Snort is run, it is observed that Skype, which does not try a different path from its previous behavior, is blocked. In another study [4], the use of Onion Routing (TOR) software, which enables anonymous communication on the Internet, is detected in the network using DPI and how this application is blocked with the analysis results obtained. For this a Bro-IDS server with DPI, Wireshark and a Squid proxy, which can execute some commands when a new TCP connection is established (event-driven). In order to characterize the TOR traffic, as it is encrypted with the TLS protocol, we analyzed two processes necessary for TOR connection establishment, which take place before sending and receiving data. The first is the three-way handshake between the TOR user and the TOR network, also known as the transmission of SYN, SYN-ACK and ACK packets, and the second is the TLS session establishment. The three-way handshake was standard traffic compared to other connections outside TOR. However, in the TLS connection setup, also called the TLS handshake, some differences were found. In the "ClientHello" messages sent by the client to the server, two features that characterize TOR were observed: One is that the cipher suite combinations - that is, the encryption algorithms offered by the client (TOR browser) - are always the same, and the other is that the server name format is always www.<random string>.com or "www.<random_string>.net". Another difference is that the information in the certificate sent by the server after the "ClientHello" messages has the same string format as the "ClientHello" message. After these features of TOR traffic are written as rules in Bro-IDS, the blocking process starts with Bro-IDS analyzing a complete copy of the traffic between an Internet user and a proxy server. Packets with the TOR characteristic are extracted and written to the Bro-IDS log file. When this writing event occurs, Bro-IDS activates two functions: writing the destination IP addresses of the packets into an access list file and reloading the proxy with these addresses to be blocked. TOR traffic is thus blocked. Finally, in a study [5] on the detection of Adobe Creative Suite, Microsoft Sharepoint, Salesforce, Yammer and Zendesk applications in network traffic using nDPI, an automated script was written to browse the websites of the applications and the 20 most used IP addresses and domains were recorded. A whois tool was then used to verify who these IP addresses and domains actually belong to. In a case where some IP addresses were in use by two applications, they were not included in the rule set as they could not clearly identify a single application. On the other hand, if an IP address is hosted by another company but owned by the application in terms of usage rights, it is included.

Our work is based on a scenario in which the open source nDPI library [6] and the Wireshark [7] network sniffer are used together. The methodology used is similar to the above studies and is closer to Radityatama et al.'s research [5]. However, unlike this study, we aim to contribute to the open source nDPI library by detecting and adding new protocols without using IP addresses. The reason why IP addresses were not added is explained in the rule extraction sub-section.

2. Basic Definitions and Concepts

The concept of a flow refers to one or more packets with the same set of quintuple values that make up a TCP/IP connection. The elements of this quintet are the source IP address/port number, the destination IP address/port number, and the protocol information (TCP, UDP, ICMP, etc.) contained in the IP header information. This quintet uniquely identifies a TCP/IP connection. In other words, TCP/IP packets with the same quintet value belong to the same flow.

Network traffic classification is a fundamental method for identifying and analyzing the applications circulating in a network. This technique is frequently used by Internet Service Providers (ISPs) and network operators for network performance measurement, security, better management and traffic engineering. Thus, ISPs can identify, monitor and control traffic on an application or subscriber basis. In general, there are many other classes of network traffic, such as web, cloud, multimedia, messaging, email, gaming, music, database, file sharing and voice over IP (VoIP). For example, the application protocols in this study fall under the shopping and entertainment categories.

2.1 DPI

Deep packet inspection (also known as Full Packet Inspection or Information Extraction [8]) is a packet identification technique that is used for many different purposes such as network management, security, filtering, routing, censorship, statistics, quality of service (QoS) and quality of experience (QoE); it usually works as an important cog on network nodes such as firewalls, routers or distributors; it analyzes packets "in-depth" by looking not only at the header information but also at the payload content at the application layer. In short, deep packet inspection enables accurate classification of network traffic by performing context-aware processing.

DPI method is considered as two different concepts, narrow and broad [9]. Narrow refers to the pattern matching method. This method is divided into string and regular expression matching. This recognition process is implemented by matching the packet payload with predefined signatures. In the broad sense, statistical analysis, port-based matching and protocol analysis methods are considered. In this study, we use the string matching method, which is narrowly defined and is implemented in the nDPI using the file ndpi_content_match.c.inc.

The DPI method is based on learning the packet payload, which makes it highly accurate, but it can also be detrimental to user privacy and security. The transition to encrypted network traffic also plays a suppressive role on this method. Because as long as the password is not known in SSL/TLS traffic, only the key exchange in the handshake phase can be utilized. All of the application protocols in this study use encrypted connections. Therefore, only the data obtained from the key exchange in the SSL/TLS handshake phase is used for network traffic identification.

The processing of both encrypted and real-time network traffic is another topic that needs to be studied and remains a gap in the literature [10]. Due to this problem at the top of the network layers, perhaps one of the biggest challenges in the future of this field is to realize a network traffic classification that will allow direct monitoring of the traffic by reducing the classification process to the bits in the physical layer, which is the lowest step of the network layer [11].

DPI technology has two main functions. One is to make an identification and the other is to take an action based on the result of this identification [9]. Identification is to extract the characteristic of the network packet. This characteristic can indicate normal traffic flow such as application protocol, multimedia application, cloud software, or malicious content such as malware, virus, cyber attack. The action taken is the association with the security tools. Once the packet identification is complete, the IDS, IPS or firewall is activated to alert, block, redirect or log the packet. For example, a packet can be dropped by DPI if it is identified as "malware" on the firewall, a report can be sent to the network administrator if it is reported as "potential threat" on the IDS, or a connection can be blocked if it is identified as "suspicious" on the IPS. This paper addresses the identification part.

2.1.1 DPI Applications

DPI technology has many different uses. Some of them are network security, bandwidth management, quality of experience, surveillance and censorship. The study by Xu et al. [9] discusses the applications using DPI in more detail in terms of the use cases of three groups: government, ISP and enterprise. Among these use cases is the injection of advertisements through the user profile. Ad injection, first mentioned in Bendrath's study [12], is the creation of an ad profile of a user's footprint by tracking and analyzing the websites that a user browses or shops on the Internet and serving customized online advertisements. When the data collected from the websites of the seven companies mentioned in the framework of this research were analyzed, it was observed that the network traffic was largely filled with packages belonging to advertising and tracking software.

2.1.2 nDPI

nDPI is an LGPL-licensed DPI library originally developed in 2013 by ntop in C for characterizing network traffic. nDPI was inspired by the OpenDPI software, which is GPL-licensed but no longer receives updates. nDPI, similar to OpenDPI, can be used both in the Linux kernel and in user space where user programs reside. Besides Linux, it can also be installed on operating systems such as Windows, MacOS X and the BSD family. The latest version, nDPI 4.6 [13], was released in February 2023. The work in this research was done on Ubuntu, a Linux-based operating system, using the then-current version of nDPI 4.2. nDPI, with its latest version, has more than 300 application protocols.

identification, as well as TLS certificate, browser name and encryption
It also reports metadata associated with a flow, such as

its package. nDPI's main features are as follows:

- In nDPI, a protocol is usually detected using a traffic parser defined in the nDPI library and written in C. But protocols are not only found using the parser. They can also be found by port number, IP address and protocol characteristics. For example, Dropbox traffic is identified both by the parser used for local area network-based connections and by labeling HTTP traffic with the string ".dropbox.com" in the server name information as Dropbox [14]. Therefore, the lifecycle of a new streaming traffic starts with the parsing of the third and fourth layers of packets and the testing of parsers. Each parser is encoded in a separate c file to ensure modularity and extensibility, and the order in which the parsers are applied is based on the traffic type, starting with the one most likely to match the flow. For example, for TCP/80, the HTTP parser is tried first, while for TCP/UDP protocols with port 53, the DNS parser is tried first. Each flow stores the state information of the parsers that do not match in order to skip them in future iterations. The analysis ends until a match is found or after a certain number of iterations. In this study, we did not write a protocol parser since all the websites considered use TLS/HTTPS and nDPI already has an HTTP parser.

- In Deri et al.'s work [14], it is stated that the number of packets required to detect or label an application protocol as unknown in nDPI is at most eight. This means that the number of packets that need to be assembled to identify a streaming protocol signature is at most eight. It is also important when the work process is initiated. For example, if nDPI is run after the flow has started, the first packets of the flow are not analyzed and the flow may be marked as unclassified.

- In the nDPI library, each application protocol is identified by a unique protocol number and a symbolic protocol name. When a new protocol is added to the library, it must have a number and a name that has never been used before. In the nDPI language, not only basic network protocols such as TLS, DNS, but also applications such as Skype, Facebook or Youtube are called protocols. So a protocol is actually defined in two types: network and application. These protocol types can also be referred to as major and minor respectively. For example, a TCP packet with network protocol TLS and application protocol Facebook is reported as proto: 91.119/TLS.Facebook. Where 91 is the TLS-specific protocol number and 119 is the Facebook-specific protocol number. Likewise, the most common number 5 belongs to DNS.

- In an SSL connection between two end systems, nDPI has a decoder for the initial key exchange part of the SSL connection before the encrypted communication starts, which is done with the help of a public key. This allows it to extract the server name of the connected end system and determine which encrypted packets are encrypted, even if it cannot read their contents. The server name information is stored in the stream metadata of the nDPI, just as the hostname is retrieved from the HTTP header information for HTTP connections. This analyzer can both identify protocols based on server names and find self-signed SSL certificates. Self-signed certificates are public key certificates issued by users on their own behalf, not validated by any certificate authority. This information is valuable because a connection with this type of signature is not considered secure.

- The nDPI has a file ndpi_typedefs.h that categorizes protocols into types such as safe, acceptable, potentially dangerous, dangerous, etc. The same file also defines abstract categories for grouping protocols in a meaningful way, such as shopping, gaming, file sharing, advertisements, online cloud services, social networks, instant messaging applications, malware and banned websites.

- The ndpiReader demo application in nDPI/example is a test tool that runs from the command line and demonstrates some features of the nDPI library. To use this application, data can be provided either by ndpiReader capturing live traffic for a certain period of time or by giving as input ready-made pcap files previously recorded with network listening devices such as Wireshark. In this way, ndpiReader also works as a network sniffer. ndpiReader has several command line options that can be used as needed. For example, the -i flag allows the ndpiReader to specify the name of a pre-recorded pcap file to be given as input to the ndpiReader tool, or the name of the device or interface to be used if live traffic capture is to be performed, while the -v <1|2|3|4> flag allows the display of normal detail for number 1, more detail for 2, port statistics for 3 and hash statistics for 4. All other options and some nDPI features can be obtained using the -h flag, which means help.

- The results of the flow analysis can be saved to a csv file using the -C flag. This csv file contains flow number, source/destination IP number, source/destination port number, unique protocol number, server name, TLS version, number of packets and bytes in both directions and many more fields. In this way, the data in csv format can be organized and sorted, which can be useful for traffic analysis processes. To get an output in this format, type the following command from the command line:

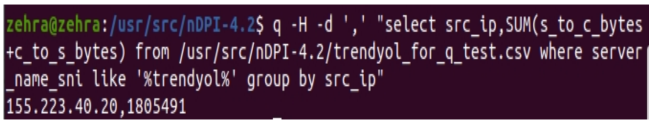
```
ndpiReader -i <file_name.pcapng> -C <file_name.csv>
```

The csv file obtained in this way is sent to the Linux command line

If used as input to the q-text-as-data application [15], which provides SQL querying capabilities, it allows us to make some inferences about traffic. For example, suppose we want to know which source IP number or client spends the most time on a particular website. With the help of the following code, we can sort all IP numbers communicating with the site by the total number of bytes in outgoing and incoming traffic using q:

```
q -H -d ',' "select src_ip,
SUM(s_to_c_bytes+c_to_s_bytes)
from <csv_file_road> where
server_name_sni like
'<server_name>' group by src_ip"
```

Here the -H flag indicates that the file contains a header line, which is used to name the columns. The -d flag works as an input delimiter. The output of an example query for packets with "trendyol" in the server name is given in Figure-2. Since the data is collected on a personal desktop computer, there is only one corresponding IP address, 155.223.40.20. In this way, network traffic analysis can be performed for the targeted purpose with different queries different from the other data in the csv file.



```
zehra@zehra:/usr/src/nDPI-4.2$ q -H -d ',' "select src_ip,SUM(s_to_c_bytes
+c_to_s_bytes) from /usr/src/nDPI-4.2/trendyol_for_q_test.csv where server
_name_sni like '%trendyol%' group by src_ip"
155.223.40.20,1805491
```

Figure-2: Example output of a q-text-as-data query

3. Materials and Methods

The general framework of the study is given in Figure 1.

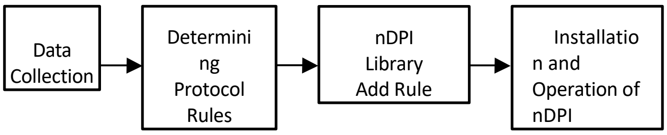

Data Collection
Determining Protocol Rules
nDPI Library Add Rule
Installation and Operation of nDPI

Figure-1: General framework of the study

In this context, the packets generated in the network as a result of the real-time use of the application we want to detect are captured with the help of the Wireshark tool and saved with the pcap or pcapng extension and given as input to the nDPI tool installed for detailed packet content analysis. Using both the results obtained from the nDPI tool and other methods that will be mentioned in the rule extraction sub-heading, a list of rules obtained from domain names was created. In the last step, these application protocols missing from the nDPI tool were added to the library as rules to increase the number of protocols that the tool can detect. These protocols include the websites used by the top seven companies on the list of the 100 largest internet companies in Turkey [16], which were published by Fast Company magazine in February 2020. These companies are Trendyol, Hepsiburada, Nesine.com, N11, Bilyoner, GittiGidiyor and Yemeksepeti in order of size. Choosing the protocol from this list The reason for this is that the interaction of these companies with the user is only over the internet and the detection of network traffic is considered to be important due to the intense data flow on the relevant websites. Finally, the study was partially automated using the Selenium web driver and the Python programming language.

3.1 Data Collection

Data were collected using two different hosts: One is a 64-bit desktop computer with dual operating systems, Windows 10 and Ubuntu 20.04.5 LTS, connected to the Internet via Ethernet cable, and a quad-core 3.60 GHz Intel Core i7-4790 processor. The other is a 64-bit laptop with a Windows Education series operating system, connected to the Internet wirelessly and equipped with a dual-core 2.50 GHz Intel Core i5-3210M processor. Both Chrome and Firefox browsers were used on both computers for web page browsing for data collection using Wireshark. The purpose of using two different computers, operating systems, connections and browsers here is to account for the possibility that different combinations of systems may yield new data. However, we did not observe a case where one of these choices yielded more data than the other. In order to determine the rules to be added to the nDPI library, first, Wireshark was opened on the websites of Trendyol, Hepsiburada, Nesine.com, N11, Bilyoner, GittiGidiyor, and Yemeksepeti, and packet capture was started by opening Wireshark and browsing for a while on each of these websites. In order to capture all the different domains of a web page, we tried to access as many page elements and types as possible. The resulting files were given various names according to protocol, browser and connection type.
It is saved in a folder with the .pcapng extension according to the operating system. For example, n11_chrome_ethernet.pcapng or trendyol_firefox_wifi.pcapng. One of these companies, GittiGidiyor, owned by eBay, announced its withdrawal from the Turkish market on June 20, 2022 and closed its online shopping site on July 18, 2022. For this reason, although the protocol of this site was studied and rules were extracted at the beginning of the study, it was not included in the automation process written later.

3.2 Rule Extraction

At the beginning of the study, the first step in analyzing the traffic data was to access the IP addresses of the domain names of the seven companies, which are clearly visible in the browser and well known to users. For this purpose, the registrars who registered these domains and their respective web addresses were first found through a whois database query. Afterwards, whois queries were made again on the web addresses of the registrars to obtain the authorized name server information and the list of IP addresses of the web pages offered by the companies to the users. In the queries made, the authorized name servers of the registrant organizations, the registered

Although it returned various information such as the name of the organization, the name and contact details of the system administrator and the address of the organization, it did not return IP addresses. Therefore, another method was tried to obtain these addresses in the form of IP address ranges. For this purpose, whois queries were made in the database of ARIN, APNIC and RIPE NCC organizations. However, no reliable results were obtained here either. For example, in the query of Trendyol company in the RIPE database, the IP ranges 159.20.112.0/24, 159.20.113.0/24, 159.20.114.0/24, 159.20.115.0/24 and 159.20.116.0/24
address ranges were obtained. However, despite listening to the network with Wireshark and applying many filters such as ip.addr == 159.20.116.0/16, which checks for the presence of a very wide IP range, none of the above IP addresses obtained from the whois query were detected in network traffic. However, an IP address-location query at https://dnschecker.org/ip-location.php shows that these IP addresses belong to various districts of Istanbul and that the name of the organization is DSM Grup Danışmanlık İletişim ve Satış Ticaret Anonim Şirketi [17], the owner of Trendyol. Therefore, as a third option, a longer but more reliable and accurate solution was to analyze all DNS queries and answers in the traffic data collected by Wireshark. Wireshark filtering methods were used to access DNS queries and responses easily and quickly. For example, the dns.qry.class filter was used to retrieve all DNS queries and answers in a flow, while the dns contains "n11" filter was applied to retrieve DNS queries and answers containing the string "n11" among packets. As a result of such queries, the data contained in the packets listed as output in Wireshark were analyzed and both the domain names and the corresponding IP addresses were recorded. In addition, there were different domain names and IP addresses obtained with the nslookup method, although they did not appear in the Wireshark results. However, since this method does not provide all IP addresses, it was used when there was a domain name that was highly likely to exist but could not be found with Wireshark. For example, Wireshark for Trendyol via collect.trendyol.com, collect2.trendyol.com, collect3.trendyol.com and collect5.trendyol.com were detected. Therefore, since it was possible that domains such as collect1.trendyol.com and collect4.trendyol.com also existed, we queried nslookup collect1.trendyol.com from the command line and found that they did indeed exist. All the domain names of Trendyol and other companies were found and registered for inclusion in the nDPI-4.2 library, using the DNS query and retrieval method described above. On the other hand, for Trendyol alone, a list of 2095 IP addresses was generated, including those obtained from the RIPE database. All IP addresses were then subjected to a second check at https://dnschecker.org/ip-whois-lookup.php, where it is possible to query who is the real owner of the IP address. Since the query tool at this address also performs a whois query, it is possible to determine which registrar the IP address belongs to.
-ARIN, APNIC, RIPE NCC, etc.- as well. Therefore, the who the addresses belong to is finally determined by performing a whois query again at the registrars. clarified. At the end of this whole inquiry process, all IP addresses in the list of IP addresses used by Trendyol obtained from Wireshark - except for those obtained from RIPE - belonged to companies such as Alibaba Cloud and CloudFlare, which are known for providing content delivery network (CDN) services. Similarly, Hepsiburada, N11 and Bilyoner were found to be using IP addresses belonging to Akamai, a company known for its CDN service. Since CDN services are now preferred by many companies, it is possible to find cases where multiple applications use different IP addresses in the same address range. An example that illustrates this situation most clearly is given in Table-1. Accordingly, the IP addresses used as the identifier of the TOR protocol defined in the nDPI-4.2 library and the IP address information obtained from Wireshark for the domain name collect.trendyol.com used by Trendyol all correspond to the IP address range of the Alibaba Cloud cloud service provided by Alibaba company in whois queries in the APNIC registry. Therefore, this suggests that the e-commerce sites we considered may use multiple IP addresses and that these IP addresses are subject to change. This was the main reason why IP addresses were not chosen as the determinant of the application protocols we wanted to identify.

A subsequent review of version 4.4 of nDPI, which was released just after 4.2, revealed that the IP address of cloud service providers such as Alibaba Cloud and Amazon, which TOR was using, had been removed.

Table-1: Two different applications using the same cloud service: TOR and Trendyol

Trendyol Addresses	TOR Addresses
8.209.80.30 8.209.81.21 8.209.88.204 8.209.89.108	8.209.79.125 8.209.93.160 8.209.94.85 8.210.144.170
APNIC Results IP address space range: 8.209.64.0-8.209.127.255 Network Name: ALICLOUD-DE	APNIC Results IP address space range: 8.209.64.0-8.209.127.255 Network Name: ALICLOUD-DE

A concrete problem encountered while analyzing the packets and caused by IP addresses was the observation that many domains were labeled under the wrong category headings in the output of the ndpiReader tool. This is because the domain names of organizations that make use of the IP addresses of cloud service providers such as Cloudflare, Amazon AWS, Google and Microsoft Azure are not included in the nDPI-4.2 library, while the IP addresses of these cloud service providers are. A few examples of this situation are given in Table-2.

Table-2: Domain names miscategorized in nDPI-4.2 and some of the cloud service applications that cause this

Domain Name and IP Address	ndpiReader Output	Name and IP Information of the Registered nDPI Protocol

sync.srv.stackadapt.com 54.87.192.123	Proto: 91.265/TLS.Amazon AWS Cat: Cloud/13	NDPI_PROTOCOL_AMAZON_AWS 54.87.0.0/16

js.appboycdn.com 104.18.22.230	Proto: 91.220/TLS.Cloudflare Cat: Web/5	NDPI_PROTOCOL_CLOUDFLARE 104.16.0.0/12
api-js.mixpanel.com 107.178.240.159	Proto: 91.126/TLS.Google Cat: Web/5	NDPI_PROTOCOL_GOOGLE 107.178.192.0/18

According to this table, domains starting with .stackadapt.com belong to the StackAdapt software company, which collects data for advertising purposes,
Domains starting with .appboycdn.com were found to belong to the cloud-based software company formerly known as AppBoy but now known as Braze, and domains starting with .mixpanel.com were found to belong to Mixpanel, a company that collects and reports data by tracking user interactions. nDPI 4.2, as in this example, found that once all the other miscategorized domains were correctly added to the library, network packets were categorized as cat: Advertisement/101 or cat: Media/1.

A second way to find domain names is to use the ndpiReader application itself. When the verbose mode of nDPI is selected, all server names serving with the internet server name -v2 are available. This method was used in the automation process.

Third and lastly, the E-company company information portal, which provides access to information such as the trade registry number, title, address information and internet address of companies in Turkey
[18] and when the information from the about or who we are sections of the relevant company's web address is examined and compared, it may be possible that there are several other domain names belonging to that organization in the network traffic. For example, although not found in searches for other companies, an examination of Trendyol reveals the company's trade name as DSM Grup Danışmanlık İletişim ve Satış Ticaret Anonim Şirketi. Since the DSM group only owns the Trendyol e-commerce website, the following domains, which were obtained when we filtered dns contains "dsm" in Wireshark, were also registered for Trendyol:

- cdn.dsmcdn.com
- img-dsmncdn.mncdn.com
- dsmgrup.com
- static.dsmcdn.com

3.3 Rule Writing and nDPI Setup

There are two sets of rules that need to be added to the library for the seven major companies before the nDPI installation:

- In the ndpi_protocol_ids.h file in the /src/include folder, a previously obsolete identification number is written for each protocol. Since the last version of nDPI 4.2 used the number 281, the protocol name and identification number from 282 to 288 are written for the protocols of the seven sites, corresponding respectively.
- Second, the respective domains of each target protocol ndpi_content_match.c.inc in /src/lib

file with the field name, how it will be named in ndpiReader if detected, protocol name, protocol category and protocol type, respectively. A few examples of these rules are given below:

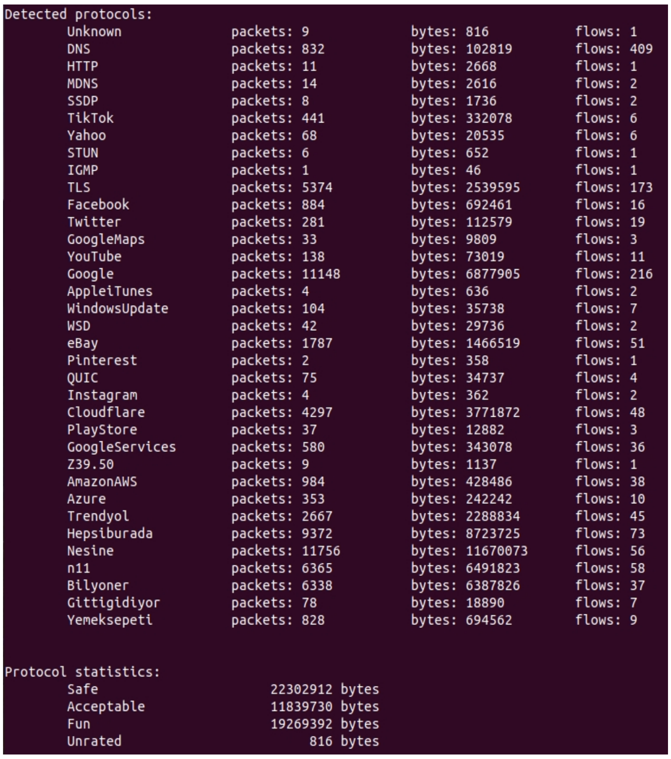
- {"trendyol.com", "Trendyol", NDPI_PROTOCOL_TRENDYOL, NDPI_PROTOCOL_CATEGORY_SHOPPING, NDPI_PROTOCOL_SAFE, NDPI_PROTOCOL_DEFAULT_LEVEL}
- {"nesine.com", "Nesine", NDPI_PROTOCOL_NESINE, NDPI_PROTOCOL_CATEGORY_GAME, NDPI_PROTOCOL_FUN, NDPI_PROTOCOL_DEFAULT_LEVEL}
- {"yemeksepeti.com", "Yemeksepeti", NDPI_PROTOCOL_YEMEKSEPETI, NDPI_PROTOCOL_CATEGORY_SHOPPING, NDPI_PROTOCOL_SAFE, NDPI_PROTOCOL_DEFAULT_LEVEL}

Once all rule entries are complete, the nDPI setup can now be performed by running the following commands in order:

```
cd
<nDPI_folder_path>
sudo ./autogen.sh
sudo ./configure
sudo make
sudo make install
```

A screenshot of the output of the application protocols detected by the ndpiReader tool after the rules are added and the nDPI is installed is given in Figure-3.

Figure-3: Detection of added protocols by nDPI



```
Detected protocols:
        Unknown              packets: 9           bytes: 816          flows: 1
        DNS                  packets: 832         bytes: 102819       flows: 409
        HTTP                 packets: 11          bytes: 2668         flows: 1
        MDNS                 packets: 14          bytes: 2616         flows: 2
        SSDP                 packets: 8           bytes: 1736         flows: 2
        TikTok               packets: 441         bytes: 332078       flows: 6
        Yahoo                packets: 68          bytes: 20535        flows: 6
        STUN                 packets: 6           bytes: 652          flows: 1
        IGMP                 packets: 1           bytes: 46           flows: 1
        TLS                  packets: 5374        bytes: 2539595      flows: 173
        Facebook             packets: 884         bytes: 692461       flows: 16
        Twitter              packets: 281         bytes: 112579       flows: 19
        GoogleMaps           packets: 33          bytes: 9809         flows: 3
        YouTube              packets: 138         bytes: 73019        flows: 11
        Google               packets: 11148       bytes: 6877905      flows: 216
        AppleiTunes          packets: 4           bytes: 636          flows: 2
        WindowsUpdate        packets: 104         bytes: 35738        flows: 7
        WSD                  packets: 42          bytes: 29736        flows: 2
        eBay                 packets: 1787        bytes: 1466519      flows: 51
        Pinterest            packets: 2           bytes: 358          flows: 1
        QUIC                 packets: 75          bytes: 34737        flows: 4
        Instagram            packets: 4           bytes: 362          flows: 2
        Cloudflare           packets: 4297        bytes: 3771872      flows: 48
        PlayStore            packets: 37          bytes: 12882        flows: 3
        GoogleServices       packets: 580         bytes: 343078       flows: 36
        Z39.50               packets: 9           bytes: 1137         flows: 1
        AmazonAWS            packets: 984         bytes: 428486       flows: 38
        Azure                packets: 353         bytes: 242242       flows: 10
        Trendyol             packets: 2667        bytes: 2288834      flows: 45
        Hepsiburada          packets: 9372        bytes: 8723725      flows: 73
        Nesine               packets: 11756       bytes: 11670073     flows: 56
        n11                  packets: 6365        bytes: 6491823      flows: 58
        Bilyoner             packets: 6338        bytes: 6387826      flows: 37
        Gittigidiyor         packets: 78          bytes: 18890        flows: 7
        Yemeksepeti          packets: 828         bytes: 694562       flows: 9


Protocol statistics:
        Safe                      22302912 bytes
        Acceptable                11839730 bytes
        Fun                       19269392 bytes
        Unrated                        816 bytes
```

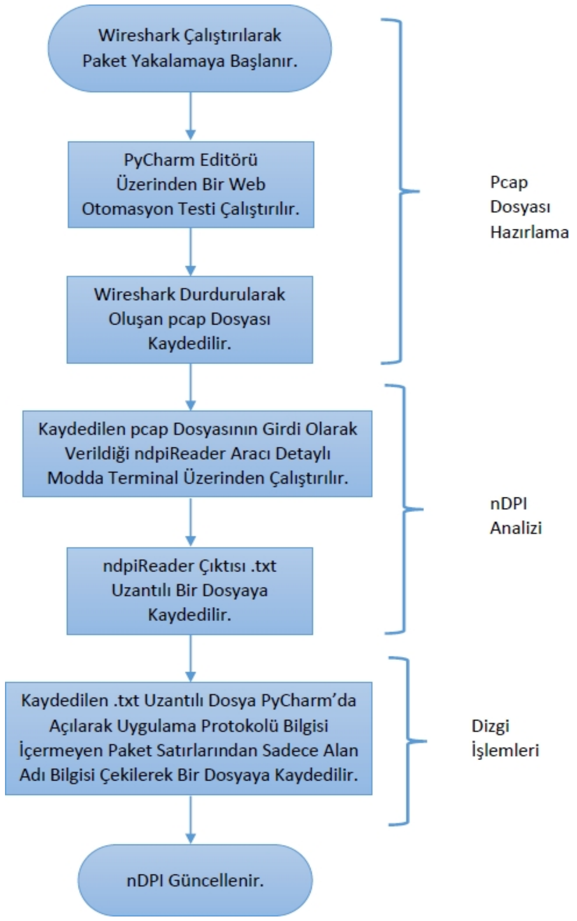

Wireshark Çalıştırılarak Paket Yakalamaya Başlanır.
PyCharm Editörü Üzerinden Bir Web Otomasyon Testi Çalıştırılır.
Wireshark Durdurularak Oluşan pcap Dosyası Kaydedilir.
Pcap Dosyası Hazırlama
Kaydedilen pcap Dosyasının Girdi Olarak Verildiği ndpiReader Aracı Detaylı Modda Terminal Üzerinden Çalıştırılır.
ndpiReader Çıktısı .txt Uzantılı Bir Dosyaya Kaydedilir.
nDPI Analizi
Kaydedilen .txt Uzantılı Dosya PyCharm'da Açılarak Uygulama Protokolü Bilgisi İçermeyen Paket Satırlarından Sadece Alan Adı Bilgisi Çekilerek Bir Dosyaya Kaydedilir.
Dizgi İşlemleri
nDPI Güncellenir.

Figure-4: Automation flow

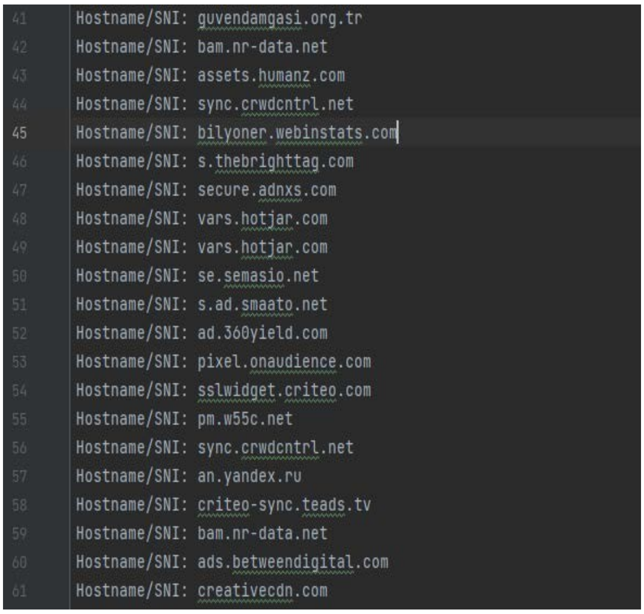
4. Automation

The flowchart of the automation process is given in Figure 4. Accordingly, automation consists of three parts; pcap file preparation, nDPI analysis and typesetting. The first one is the realization of the pcap or pcapng files obtained from manual navigation of each web page using web automation tests. For this purpose, a Selenium web automation framework was created using the open source Selenium web driver and the Python programming language, with one test file and html report for each web page. After running the web automation tests, Wireshark is stopped and the resulting pcap file is saved. Thus, the first stage of the automation flow, preparing the pcap file, is finished.

The second stage is the nDPI analysis. The pcap file is given as input to the ndpiReader application run from the command line, using the -i switch. The detailed mode, -v2 switch, is used by nDPI to get detailed packet results. When the nDPI analysis is finished, the terminal output is saved in a .txt file. The following command line is sufficient for all second phase operations:

```
ndpiReader -i <filename>.pcapng -v2 >> <filename>.txt
```

The third stage is typesetting. The text document saved in the second phase is opened using any Python IDE. All package lines are extracted and saved in a text file. Then, this text document containing only package information and detect packets for which the application protocol was not found by the nDPI. For this, the packet lines are scanned for TLS.<application_protocol> either or DNS.<application_protocol>, and instead contains the strings [proto: 91/TLS] or [proto: 5/DNS]



```
41  Hostname/SNI: guvendamgasi.org.tr
42  Hostname/SNI: bam.nr-data.net
43  Hostname/SNI: assets.humanz.com
44  Hostname/SNI: sync.crwdcntrl.net
45  Hostname/SNI: bilyoner.webinstats.com
46  Hostname/SNI: s.thebrighttag.com
47  Hostname/SNI: secure.adnxs.com
48  Hostname/SNI: vars.hotjar.com
49  Hostname/SNI: vars.hotjar.com
50  Hostname/SNI: se.semasio.net
51  Hostname/SNI: s.ad.smaato.net
52  Hostname/SNI: ad.360yield.com
53  Hostname/SNI: pixel.onaudience.com
54  Hostname/SNI: sslwidget.criteo.com
55  Hostname/SNI: pm.w55c.net
56  Hostname/SNI: sync.crwdcntrl.net
57  Hostname/SNI: an.yandex.ru
58  Hostname/SNI: criteo-sync.teads.tv
59  Hostname/SNI: bam.nr-data.net
60  Hostname/SNI: ads.betweendigital.com
61  Hostname/SNI: creativecdn.com
```

If there are any lines, the "Hostname/SNI:<domainname>" strings in these lines are extracted. Thus, packets whose application protocol is not detected by nDPI are found and the new domain names are saved in a separate text file. A sample output of the script written for string operations is given in Figure-5. According to this figure, the domain name bilyoner.webinstats.com on line 45 appears as a new domain name belonging to Bilyoner from seven protocols.

Figure-5: An example of the output obtained as a result of typesetting operations

Finally, nDPI is updated by determining whether the newly obtained domains should be added as rules or not. It is important to note that the new domains include domains belonging to application protocols that do not have a unique number defined in the ndpi_protocol_ids.h file, such as advertising or CDN networks. This is because such protocols are defined by category, so the strings [proto: 91/TLS] or [proto: 5/DNS] still appear in the packet contents. If these domains are not present in the nDPI, the nDPI abstract categories CUSTOM_CATEGORY_ADVERTISEMENT and

It can be added using NDPI_PROTOCOL_CATEGORY_MEDIA. In this way, 6 domains belonging to CDN networks and 74 domains belonging to advertising and tracking software were found. These domains are not included in the latest version of nDPI, nDPI-4.6, released in February 2023. Therefore, there is still a need to add the domains found in this study that contain advertising and tracking software to the nDPI library in order to avoid the problem of miscategorized domains. In nDPI-4.6, unlike previous versions 4.2 and 4.4, domains belonging to companies engaged in advertising, tracking or data analytics are not treated as an abstract category. A protocol name and a corresponding identification number have been defined that can be used specifically for these types of domains. Therefore, in the new version

Using NDPI_PROTOCOL_ADS_ANALYTICS_TRACK protocol name and NDPI_PROTOCOL_TRACKER_ADS protocol type, the following rule template should be used:

```
{ "<alan_name>", "ADS_Analytic_Track",
NDPI_PROTOCOL_ADS_ANALYTICS_TRACK,
CUSTOM_CATEGORY_ADVERTISEMENT,
NDPI_PROTOCOL_TRACKER_ADS,
NDPI_PROTOCOL_DEFAULT_LEVEL }
```

5. Discussion

In this study, we consider highly popular e-commerce websites with a high number of users in Turkey. Detecting the presence of such applications in the network is important in three ways.

First, for more effective bandwidth management, end systems on a local area network may need to be scaled by system administrators to access such sites during certain periods or times. An example of this would be a campus network that experiences high traffic during enrollment renewal application or results learning periods.

Second, statistical information on individual or regional usage rates of e-commerce sites can be useful in various ways. For example, the volume of application protocols in use on an ISP's CDN network could be taken into account to implement prioritization tariffs and thus improve the quality of service. Indeed, an examination of four pcap files revealed that the flow sequence of the trunk connections shows that a CDN network is connected to after a three-way handshake starting with a SYN at the beginning of the TCP connection setup, followed by a TLS connection setup.

Third and finally, for privacy and security reasons, these sites may need to be restricted or even blocked on some networks with high security requirements. One of the most concrete examples of this is the cyber-attack on March 18, 2021, which was carried out by accessing a web application server of Yemeksepeti. The compromised data included first name, last name, date of birth, phone number, email addresses, address information and passwords summarized with the SHA-256 algorithm. Even if the password itself is unknown, such data is actually a valuable resource for malicious actors. For example, according to a study conducted by Keeper Security on 1000 full-time employees in the United States who accessed their work-related online accounts with the help of a password, 44% of these employees had the same password for both their personal and work-related accounts [19]. Furthermore, experts estimate that about 50% of all internet users still use the same password for all their online accounts [20]. Returning to the Yemeksepeti data theft incident, an attacker who holds the hashed passwords can perform a Pass the Hash (PtH) attack on the Yemeksepeti application if there is a weakness in the authentication protocol such as keeping the hashed passwords constant from session to session. A PtH attack allows an attacker to use the hashed password obtained from a system to authenticate his or her identity to the authentication protocol in that system, as if he or she were starting a new session. to infiltrate the system. In a weak system where password digests remain constant across different sessions, it is possible for the attacker to gain time by moving around the system undetected until the password is changed, since the digests will not change unless the password is changed [21]. Thus, by trying the summarized passwords, the attacker will be able to access the accounts, and from there to the public version of the password, and perhaps from there to the accounts used by critical people at work or to the servers containing sensitive data belonging to the company. On the other hand, it is possible to launch an attack to find passwords just from date of birth and name information. According to a study [22], internet users in the United States

For nearly 60%, the password for their online accounts is a string containing a name or date of birth. Considering all these attack methods, the number of people affected by the attack - 21 million 504 thousand 83, which is a quarter of Turkey's population - officially reported by Yemeksepeti to KVKK, and the fact that this data theft was detected only a week after the attack - on March 25, 2021 - it is possible to say that this constitutes a very serious security vulnerability. On the other hand, the websites of Trendyol, N11, and Hepsiburada contain not only data such as name, surname, date of birth, phone number, email addresses, and address information, as in the Yemeksepeti example, but also data such as product search, location information, and purchase information, which, when combined together, can enable many more inferences to be made about individuals (such as their preferences, tendencies, and health status). Since this data, which can also be collected electronically for surveillance and tracking purposes, can also be shared with third-party companies, restrictive measures may need to be taken for people working in important positions and the organizations they work for.

6. Conclusion

In this research, deep packet inspection method, which is a structure in which application protocols can be detected by analyzing not only the header information of the packets but also the payload content in the application layer, which is the top layer in the five-layer network model, is discussed. In this study, new protocols were added to the open source nDPI tool using this model. Since the protocols used in this study use encrypted connections, only the server name and hostname information - before the connection becomes encrypted in the SSL handshake phase - was used to identify the websites used by Trendyol, Hepsiburada, Nesine.com, N11, Bilyoner, GittiGidiyor and Yemeksepeti in the network traffic. A partial automation has been written to retrieve new and updated domain names that may arise in the future. While collecting data on these websites, 6 domains belonging to CDN networks and 74 domains belonging to advertising and tracking software were found. This software runs immediately after connecting to the relevant website and the interaction continues until the end of the connection. For this reason, there may be a need to restrict the use of such sites, which have the potential to process user data over the internet and perhaps transmit it to third party companies, for institutions that contain sensitive data that need confidentiality or authorized persons in senior management positions in these institutions. As a result of this restriction, possible attack situations such as Yemeksepeti or security vulnerabilities that may arise due to direct use by the e-commerce site itself or sharing with third party companies can be prevented. In this study, new rule definitions for network packets that were found to be miscategorized by nDPI were defined to correct such situations.

In future studies, it is aimed to contribute to the field by detecting fake sites and link traps using DPI methods.

Source

[1] Vailshery, L. S., "*IoT connected devices worldwide 2030*", https://www.statista.com/statistics/1183457/iot-connected-devicesworldwide/ , 2021.

[2] Brook, C., "*What is Deep Packet Inspection? How it Works, Use Cases for DPI, and More*", https://digitalguardian.com/blog/what-deep-packet- inspection-how-it-works-use-cases-dpi-and-more, 2018.

[3] Renals, P. and Jacoby, G. A., *Blocking skype through deep packet inspection*, in 42nd Hawaii International Conference on System Sciences, 2009, pp. 1-5, doi: 10.1109/HICSS.2009.90.

[4] Saputra, F. A., Nadhori, I. U. and Barry, B. F., *Detecting and blocking onion router traffic using deep packet inspection*,, 2016 International Electronics Symposium (IES), 2016, pp. 283-288, doi: 10.1109/ELECSYM.2016.7861018.

[5] Radityatama, G. A., Lim, C. and Ipung, H. P., *Toward full enterprise software support on nDPI*, 6th International Conference on Information and Communication Technology (ICoICT), 2018, pp. 1-6, doi: 10.1109/ICoICT.2018.8528792.

[6] ntop, "*nDPI*", https://github.com/ntop/nDPI.git, 2023.

[7] Wireshark, "*About Wireshark*", https://www.wireshark.org/about.html, 2023.

[8] Wikibooks, "*Intellectual Property and the Internet/Deep packet inspection*", https://en.wikibooks.org/wiki/Intellectual_Property_and_the_Internet/Deep_packet_inspection, 2023.

[9] Xu, C., Chen, S., Su, J., Yiu, S. M. and Hui, L. C. K., *A survey on regular expression matching for deep packet inspection: Applications, algorithms, and hardware platforms*, IEEE Communications Surveys & Tutorials, 18(4), 2016, pp. 2991-3029, doi: 10.1109/COMST.2016.2566669.

[10] Papadogiannaki, E. and Ioannidis, S., *A survey on encrypted network traffic analysis applications, techniques, and countermeasures*, ACM Computing Surveys, 2021, 54(6), pp. 1-35, https://doi.org/10.1145/3457904

[11] Mellia, M., Pescapè, A. and Salgarelli, L., *Traffic classification and its applications to modern networks*, Computer Networks (Vol. 53), 2009, pp. 759-760, 10.1016/j.comnet.2008.12.007.

[12] Bendrath, R., Global technology trends and national regulation: Explaining variation in the governance of deep packet inspection, International Studies Annual Convention, New York, 2009, pp. 15-18.

[13] ntop, "nDPI 4.6 (Feb 2023)", https://github.com/ntop/nDPI/releases/tag/4.6, 2023.

[14] Deri, L., Martinelli, M., Bujlow, T. and Cardigliano, A., *nDPI: Open-source high-speed deep packet inspection*, 2014 International Wireless Communications and Mobile Computing Conference, 2014, pp. 617-622, doi: 10.1109/IWCMC.2014.6906427.

[15] Attia H. B., "*q - Run SQL directly on CSV or TSV files*", http://harelba.github.io/q/.

[16] Fast Company Turkey, "*The 100 biggest internet companies*", https://fastcompany.com.tr/calisma-hayati/en-buyuk-100-internet-sirketi/ , 2020.

[17] DSM Grup Danışmanlık İletişim ve Satış Ticaret A.Ş, https://www.dsmgrup.com/.

[18] E-Company, https://e-sirket.mkk.com.tr/esir/.

[19] Whitney, L., "*How poor password habits put your organization at risk*", https://www.techrepublic.com/article/how-poor-password-habits-put-your-organization-at-risk/, 2021.

[20] Crafford, L., "*7 Bad Password Habits to Break Now*", https://blog.lastpass.com/2021/01/7-bad-password-habits-to-break-now-2/, 2021.

[21] Delinea, " *What is a Pass-the-Hash attack?*", https://delinea.com/what-is/pass-the-hash-attack-pth, 2022.

[22] Google and The Harris Poll, "*The United States of P@ssw0rd$*", https://storage.googleapis.com/gweb-uniblog-publish-prod/documents/PasswordCheckup-HarrisPoll-InfographicFINAL.pdf, 2019.