STUDENT SESSION

CYBER SECURITY AND DOMAIN NAME SYSTEMS DEPLOY AND PROTECT NETWORK WITH DNS SINKHOLE BLACKHOLE

Aleksandar Jokić (Student)*,
Marko Šarac (Mentor)

University Singidunum,
Belgrade Serbia

Abstract:

DNS is a protocol within the standards for how computers exchange data on the Internet, known as TCP/IP protocol suite. A DNS server, also called a name server, handles a massive database, which maps domain names to IP addresses. The Domain Name System (DNS) [1] is central to the operation of modern networks, translating human-readable domain names into machine-usable Internet Protocol (IP) addresses. DNS makes navigating to a website, sending an email, or making a secure shell connection more accessible and is a crucial component of the Internet's resilience. As with many Internet protocols, DNS is one of them which not withstand abuse from wrong actions intent on causing harm. "Protective DNS" [2] is different from earlier security-related changes to DNS. It is a security service – not a protocol – that analyses DNS queries and takes action to mitigate threats, leveraging the existing DNS protocol and architecture.

Keywords:

DNS, Protective DNS, DNS Sinkhole, protection from Ads.

INTRODUCTION

The domain name system is indisputably one of the most important and overlooked parts of the Internet. Since it is difficult to memorise many IP addresses, which are strings of numbers, DNS came into existence. Domain name system (DNS) manages a considerable database mapping IP address against domain names. DNS takes the URLs we enter in our web browsers as input, finds the IP addresses of the web servers hosting those sites, and returns those IP addresses [3].

Phishing is the attempt to obtain confidential information such as usernames, passwords, and details of credit and debit cards, often for malicious reasons, by tricking the user. There are some approaches to prevent phishing. DNS sinkhole is one among them. DNS sinkhole, also called blackhole DNS, is used to spoof DNS servers to prevent resolving hostnames of specified URLs. A sinkhole is a way of redirecting malicious Internet traffic so that it can be captured and analysed by security analysts. That can be done by configuring the DNS forwarder to return a false IP address to a particular URL.

Correspondence:
Aleksandar Jokić

e-mail:
aleksandar.jokic.10.dls@singimail.rs

DOI: 10.15308/Sinteza-2022-421-426

DNS sinkhole can restrict access to specific sites that violate corporate policies, including abusive social networking content. In DNS sinkhole, we create two lists called whitelist and blacklist. Malicious URLs can be collected from already known servers through the open-source sites providing malicious IP details. The known malicious URLs will be placed on a blacklist, whereas the whitelist contains essential URLs. The URLs present on a blacklist can never be accessed. The URLs present in the whitelist are safe for sure. DNS sinkhole verifies the input DNS query with the elements present in the whitelist. If the system query finds a match on the list, the user will have full access to the IP address. Otherwise, it verifies with those present on the blacklist, and the user gets information that access is forbidden [4].

When a new domain is on the blacklist, the domain ultimately falls under the control of the sinkhole administrator. After this, it is no longer possible to access the original host. The blacklists must be updated constantly by the administrators of the DNS sinkhole. Open-source lists of known adware sites, malware sites, and information from other sources can combine with organisation-specific information from DNS resource record queries from affected clients' analysis of malware found on compromised clients. The mentioned information can add to the blacklist of a sinkhole. The DNS sinkhole can also control some domains that are not malicious or fraudulent but contravene the policies. DNS sinkhole can be a part of security [5].

2. WHERE IS THE PROBLEM?

It is the age of bots. Botnet traffic is increasing daily, exploiting computer systems through various infection vectors and establishing command and control channels for sale or lease to the highest bidder. Attacks are seen as a form of cybercrime and other illegal activities. Security analysts and administrators must respond daily to malware that forces users to unknowingly download suspicious files from websites that they have no reason to access. Available applications that are dangerous must also be blocked. The following figure shows that botnet activity has been increasing day by day.

It is the duty of security analysts and anti-malware engineers to control and prevent bots and other unwanted traffic. DNS sinkhole can play a significant role in preventing access to known malware sites as a part of security [6].

3. DOMAIN CLASSIFICATION

A core capability of Protective DNS is the ability to categorise domain names based on threat intelligence. Protective DNS services typically leverage known malicious domains' open source, commercial, and governmental information feeds. These feeds enable coverage of domain names found at numerous points of the network exploitation lifecycle. Some solutions may also detect novel malicious domains based on pattern recognition. The types of domains typically addressed by a Protective DNS system include the following:

- Phishing: Sites known to host applications that maliciously collect personal or organisational information, including credential harvesting scams. Protective DNS can protect users from accidentally connecting to a potentially malicious link [7].
- Malware distribution sites are known to serve malicious content or used by threat actors to command-and-control malware. For example, these may include sites hosting malicious JavaScript® files or domains that host advertisements that collect information for undesired profiling. Protective DNS can block and alert on known malicious connection attempts.
- Advanced malware – including some botnets – depends on communicating with command and control (C2) infrastructure. Cyber threat actors use domain generation algorithms (DGAs) for malware to circumvent static blocking – either by domain name or IP – through programmatically generating domain names according to a preset speed. Protective DNS can offer protection from malware DGAs by analysing every domain's textual attribute and tagging those associated with known DGA attributes, such as high entropy.
- Content filtering: Sites whose content is in specific categories against an organisation's access policies. Although an ancillary benefit to malware protection, Protective DNS can use a categorisation of various domains' use cases (e.g., "gambling") and warn or block on those that are a risk for a given environment [8].

4. RESPONSE TO DNS AND FUNCTIONALITIES

The DNS sinkhole bypasses the DNS request and provides the response configured by the DNS sinkhole administrator. It does not allow the domain to resolve requests by its authoritative owner. Instead, the DNS sinkhole intercepts the DNS request and responds with an authoritative answer configured by the organisation.

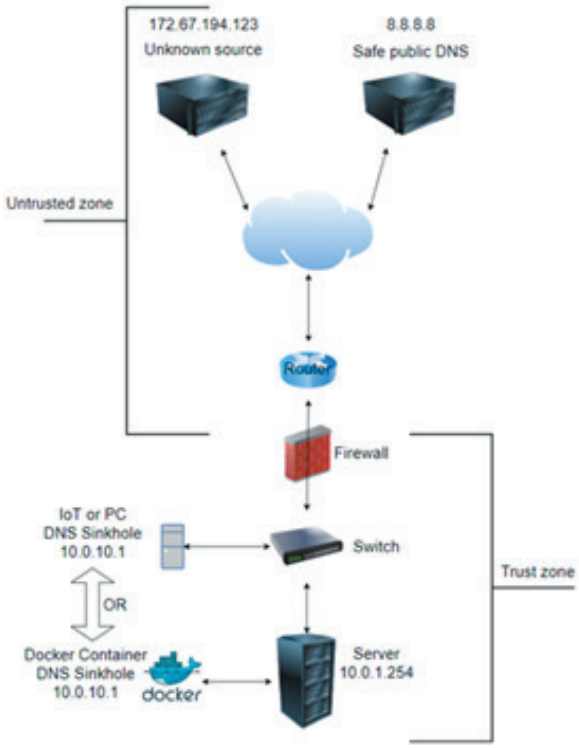

Figure 1 - An example workflow of a sinkhole on the primary network

With the basic sinkhole functionality, the malware on the infected machine attempts to initiate a connection to a system hosted on a URL with a known malicious domain configured in the DNS sinkhole. However, the request does not pass to the malicious URL. Instead, all DNS requests from the client redirect to the sinkhole, which is the localhost IP address. The client cannot contact the malicious site, and the command-and-control connection with the botnet can not establish the connection. The botmaster will be unaware that the compromise has occurred.

After this step, the preparation, detection, and partial containment are finished. Containment is partial because the compromised computer may still attempt to attack internal computers.

Therefore, the corresponding teams should carry out additional analysis and eradication steps.

1. DNS sinkhole workflow: The client hosts on the server, and sinkhole IP addresses must be in different zones, so sessions pass through the firewall. The sinkhole IP address does not have to be an active host, just an unused IP address.
2. DNS sinkhole setup overview (IoT or Docker): For demonstration, we will use popular PiHole DNS Sinkhole as Network-wide ad blocking and block in-app advertisements, improving network performance and, in the end, coming with complete monitor statistics with API for extending stats.

For this configuration, the process is used Raspberry Pi Zero W with a microSD memory card of 8GB size, USB to UDP connection, and another experiment was with Docker container on the server, VM with double CPU core and 8GB size of RAM and the exact size of HDD [9].

5. INSTALLATION, CONFIGURATION, AND TESTING NETWORK

5.1. FIRST TEST CONFIGURATION:

For that matter, Raspberry Pi Zero, or any other Raspberry Pi model, needs an operating system before running any software or application. While there are different ways to go about this, we recommend using Raspberry Pi Imager. The Imager is an easy-to-use utility that lets conveniently flash an image onto a microSD card. In advanced settings, it is essential to set up an SSH connection with the user and password for, and if not using LAN connection, in the same settings, configure a Wi-Fi connection.

Plug-in microSD card either using an adapter or a card reader to the computer. Now, in Imager, choose which OS to install which memory card and then Write to start flashing OS on the memory card.

When the flashing process finishes, remove the microSD card, and plug it into Raspberry Pi Zero. Raspberry Pi Zero should now boot up. Start Putty and connect to Raspberry Pi by SSH connection. Default login credentials in the terminal window are - username: pi; password: raspberry.

Lastly, we need to assign a static IP (Internet Protocol) address to the Raspberry Pi to prevent your router from assigning a new IP address to the Pi every time it connects. To do this, first, get the static IP address currently assigned to Raspberry Pi Zero. The easiest way to do this is to type in arp -a in the terminal window, which presents the interface, the IP address, and the MAC address of Pi. Alternatively, head to the router's configuration page to view these details.

And hit Enter\return. The above command may seem odd to some — piping curl to bash — if we are doing it over HTTPS and aware of the source installing the software, we should be fine. Using that command makes the installation procedure efficient and quick.

The Pi-hole installer should start now, and all we have to do from hereon is follow the on-screen instructions. We need to particularly pay attention to a few screens for static IP, upstream DNS provider, and ad-services blacklist. For static IP, make sure it matches the one we added to the configuration file, and for upstream DNS, select the DNS service we prefer (Google, Cloudflare, AdGuard).

In addition to the above screens, we must also carefully attend to the screens that ask for web admin interface installation and web server installation. Likewise, we can select to log queries and set privacy mode depending on preference on the screens that follow. If we set Pi-hole on a home network, we can enable both options. However, it is necessary to select an option in an office setting sensibly.

Finally, once the installation is complete, we are presented with a screen that contains all the necessary settings for our Pi-hole. Note these settings or take a snapshot as we will need them later [10].

Once have the IP address, enter the following command in the terminal:

```
sudo nano /etc/dhcpcd.conf
```

Furthermore, enter/modify the following details:

```
[hostname]interface eth0
static ip_address=[IP address with subnet mask]
static routers=[router gateway address]
static domain_name_servers=[router DNS or other preferred DNS server]
```

Alternatively, a Static IP address we can be set up by a MAC address on the DHCP server. The only thing needed is to make an Address reservation by MAC address.

We are now on to the final step in the process, which is to install Pi-hole on the Raspberry Pi. For this, enter the following command in the terminal window:

```
curl -SSL <https://install.pi-hole.net> | bash
```

5.2. SECOND TEST CONFIGURATION

How to set up pi-hole and Docker?
Open a terminal and Start Docker

```
sudo systemctl start docker
```

Enter the command to download PiHole from docker hub:

```
sudo docker pull PiHole/PiHole
```

Change the DNS to something else like google:

```
set DNS 8.8.8.8
```

Open a File

```
sudo nano /etc/resolv.conf
```

Furthermore, copy/paste the below code:

```
version: "3"
services:
  PiHole:
    container_name: PiHole
    image: PiHole/PiHole:latest
    ports:
      - "53:53/tcp"
      - "53:53/udp"
      - "67:67/udp"
      - "80:80/tcp"
      - "443:443/tcp"
    volumes:
       - './etc-PiHole/:/etc/PiHole/'
       - './etc-dnsmasq.d/:/etc/dnsmasq.d/'
    dns:
      - 127.0.0.1
      - 1.1.1.1
    cap_add:
      - NET_ADMIN
    restart: unless-stopped
```

Run the compose file to launch PiHole:

```
sudo docker-compose up -d
```

Move inside PiHole container:

```
sudo docker exec -it PiHole bash
```

Change PiHole password:

```
PiHole -a -p
exit
```

5.3. HOW TO USE PI-HOLE?

With Pi-hole running on the local network, all we must do to use it and block ads is configure the DNS. There are two ways to do this: we can either make Pi-hole's IP the default DNS of the entire network or point all devices to this IP address to route their traffic.

The former approach requires changing the DNS client for the network from the router's settings, while the latter only requires changing the DNS server on each device from their Wi-Fi settings. Now use the PiHole as DNS instead of any other:

Go to the DNS setting in Windows…

Settings –> Network and Internet –> Ethernet (if we want to connect to ethernet) or Wi-Fi (if your laptop connected to Wi-Fi) –> change adapter options –>right click on Wi-Fi or ethernet and go to properties –> select IPv4 –> properties –> change the DNS to the IP of PiHole.

Select IPv4 –> Change the DNS from obtaining DNS automatically to Use following DNS server and write the IP the box. In second write any DNS like 8.8.8.8

6. CONCLUSION

To summarise, deception as an intelligent sinkhole in the organisation is easy to configure and yields much information about the adversaries' activities. We can gather intelligence and information about the attacker's goals and learn how to detect and prevent current and future attacks.

DNS always was seen as an attacker's target. The concept of Protective DNS came that it is critical to leverage DNS to protect our network from threats. DNS sinkhole technique is the best method for defending from malware.

The sinkhole is OS and protocol-independent, which we can configure as a shield and real-time reporting notification system to improve our response. A complete analysis of logs in the DNS sinkhole helps us research threats in the network deeply.

7. REFERENCES

[1] V. Mladen and J. Aleksandar, Uvod u računarske mreže., Belgrade: Univerzitet Singidunum, Fakultet za poslovnu informatiku, 2007.

[2] P. Sector, "National Cyber Security Centre," 17 August 2017. [Online]. Available: https://ncsc.gov.uk/information/pdns. [Accessed 18 March 2022].

[3] A. K. Jain, "A novel approach to protect against phishing attacks at client side using an auto-updated whitelist," 6 May 2016. [Online]. Available: https://link.springer.com/article/10.1186/s13635-016-0034-3#Abs1. [Accessed 17 March 2022].

[4] S. Ji, C. Im, M. Kim and H. Jeong, "Botnet Detection and Response Architecture for O*ffering Secure Internet Services*," 2009. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/4725354. [Accessed 17 March 2022].

[5] L. HG., C. SS., L. YS. and P. HS, Enhanced Sinkhole System by Improving Post-processing Mechanism, Berlin: Springer, Berlin, Heidelberg, 2010.

[6] D. Zhao, I. Traore, B. Sayed, W. Lu, S. Saad, A. Ghorbani and D. Garant, "Computers & Security," in Botnet detection based on traffic behaviour analysis and flow intervals, Elsevier Ltd, 2013, pp. 2-16.

[7] K. Thomas, C. Grier, J. Ma, V. Paxson and D. Song, "Design and Evaluation of a Real-Time URL Spam Filtering Service," in *Symposium on Security and Privacy*, Oakland, CA, USA, 2011.

[8] T.-D. Nguyen, T.-D. CAO and L.-G. Nguyen, "DGA Botnet detection using Collaborative Filtering and Density-based Clustering," in *Symposium on Information and Communication Technology*, New York, NY, USA, 2015.

[9] Taib, A. Mat and M. F. H. Ishak, "Securing network using Raspberry Pi by implementing VPN, Pi-hole, and IPS (VPiSec)," International Journal 9.1.3, 2020.

[10] L. Pounder, "tom's HARDWARE," 1 August 2021. [Online]. Available: https://tomshardware.com/how-to/set-up-pi-hole-raspberry-pi. [Accessed 16 March 2022].