World Journal of Advanced Research and Reviews

eISSN: 2581-9615 CODEN (USA): WJARAI

Cross Ref DOI: 10.30574/wjarr

Journal homepage: https://wjarr.com/

(Research Article)

Check for updates

Mitigating DNS Amplification Attacks at the DNS Server: Using BGP AS Paths and Ingress Filtering

Christian Bassey [1, *], Francis Jeremiah [1], Rustem Iuzlibaev [1], Opeyemi Oloruntola [2] and Success Imakuh [3]

[1] *Department of Security and Network Engineering, Innopolis University, Innopolis, Russia.*
[2] *School of Computing and Mathematical Sciences, University of Greenwich, London, U.K.*
[3] *Department of Computing, Teesside University, Middlesbrough, U.K.*

World Journal of Advanced Research and Reviews, 2024, 22(03), 331–335

Publication history: Received on 27 April 2024; revised on 03 June 2024; accepted on 05 June 2024

Article DOI: https://doi.org/10.30574/wjarr.2024.22.3.1716

Abstract

These days, quite a large number of application servers are being considered to be easily spoofed. Even though technologies like DNSSec, DNS over HTTPS/TLS, and DNSCurve have always been suitable for this type of problem, many developers need help to exercise the complete chain of trust. Implementing the mentioned protocols might be a matter of time, inexperience, or impossibility. In this paper, some workarounds that rely on BGP Autonomous System numbers (AS) are shown, and protocols therein are described by way of Unicast Reverse Path Forwarding (uRPF), its benefits and drawbacks from an analytical standpoint, as well as the primary flow to defend end systems, are presented. Our approach focuses on filtering malicious traffic closer to the source by identifying anomalies in BGP AS path information. The methodology is implemented and tested using Snort as an Intrusion Detection System (IDS) to capture and analyze DNS request patterns, then MikroTik router configurations are used for strict uRPF and ingress filtering, demonstrating the practical application of this solution proposed solution in real-world network environments.

Keywords: BGP; Security; Spoofing; DNS; Ddos; Ingress Filtering; Urpf; Network Security; Autonomous Systems.

1. Introduction

Spoofing tools are used to obfuscate that a particular server is the one it claims to be [1]. To put it another way, the host pretends to be some other host. It is essential to realize that unreliable networks (like the Internet) and most end devices mostly consider the destination or source IP address for further forwarding or replying, as in the case of packets requiring a response. With this in mind, end hosts/networks are responsible for watching the incoming traffic and verifying it.

With this possibility of spoofed packet characteristics, a Domain Name System server which is the primary mechanism for the resolution of domain names to IP addresses on the internet, is vulnerable to sending responses to spoofed source addresses as such, participating unwillingly in a distributed denial of service amplification attack [2]. This is because authoritative DNS servers are usually allowable for querying by third parties outside of the DNS server's network. These third-party queries have introduced a new DDoS attack vector in amplification attacks, allowing malicious actors to spoof source addresses and make DNS requests, causing unsolicited replies to the victim's address.

This attack heavily relies on the victim's bandwidth and other system resources being insufficient to process the huge amount of traffic the attacker sends to it via means of multiple DNS resolvers [3]. The DNS request is crafted to elicit the largest responses from the resolving server as a result, the victim whose IP address was spoofed receives an

[*] Corresponding author: Christian Bassey

amplification of the original traffic and in situations where there are multiple of such responses coming in at the same moment, the victim's network infrastructure is overwhelmed.

Identification of these malicious requests by the DNS servers involved in this amplification is low as multiple DNS servers are involved in the attack thus the regular methods of rate limiting multiple requests from one source address on the DNS server side do not apply.

These attacks happen over the Internet and would require the traffic to be routed first to the DNS server from the attacker and subsequently to the victim. Given that the attacker, amplifying DNS servers and victim are likely to be in different autonomous systems, there will be a routing protocol between these AS's this is the Border Gateway Protocol. The use of BGP AS paths in combination with ingress filtering on the amplifying server side has often been overlooked in studies for the mitigation of amplification attacks. This paper explores this option.

This paper is structured as follows, in section 2 is an overview of studies carried out with respect to mitigating amplification attacks and identifying spoofed IP packets, in section 3 there is a discussion of the goals for the research and the improvements over other studies, section 4 outlines the methodology for the detection of the spoofed packets and the setup of the research network. Sections 5 discusses the results of this research work and in section 6 conclusions and recommendations for further research is made.

2. Related Studies

Given the abundance of authoritative nameservers and the relative ease of effort required for carrying out an amplification attack, various research works exist in this domain which have explored mitigating these attacks from different dimensions. A few of these works are explored below.

Kambourakis et. al. [3] proposed a new method of detecting amplification attacks using a monitor to record outgoing request and incoming response pairs. A custom-made DNS Amplification Attacks Detector tool was implemented to process captured network packets and give out an alert in the situation that responses without requests (orphaned responses) are recorded.

Kim et. al. [4] proposed the use of software defined networking to store DNS queries histories to differentiate normal responses from attack responses. The study showed that malicious DNS responses can be blocked without incurring significant overhead.

Verma et. al. [5] proposed a query rate sharing between the victim and the DNS resolvers such that when the unsolicited packets begin arriving at the victim's network, the victim forwards the new DNS server from whom it has received traffic from and the query/response rate to other resolvers involved in the amplification attack. This way, all the resolvers build up an estimated consolidated query rate that is going towards that server and mitigate the attacks locally.

Vaidyanathan et. al. [6] in the study on use of BGP AS numbers to detect IP address spoofing proposed that a profile of BGP AS numbers and their prefixes be maintained. From this profile, an expected AS number and interface for each prefix will be generated and in situations where the AS numbers and prefixes do not match the EAS an appropriate notification is generated.

3. Background and Research goals

This section describes the typical architecture of an amplification attack, the focus points of mitigations in prior research and the proposed goals for this research paper.

The goal of an amplification attack is to use up the victim's network infrastructure and system resources. This is because a small DNS request can elicit a large response as such, one both making simultaneous requests to multiple resolvers with a victim's IP address can cause a denial of service on the victim. Below is what a typical amplification architecture looks like [2]:

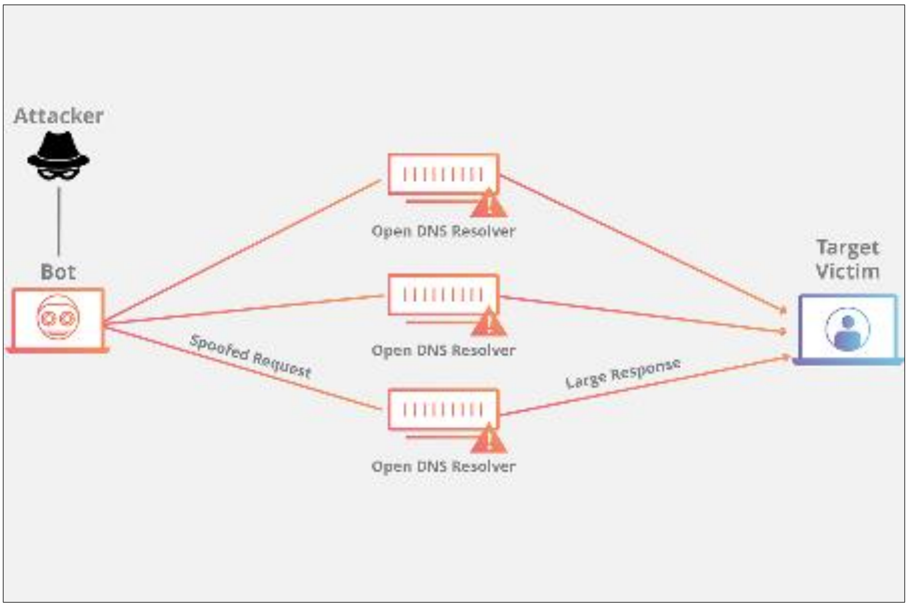

Figure 1 Architecture of a DNS Amplification Attack

In 2017, an amplification attack at Google IPs using an array of CLDAP, DNS, and SMTP servers peaked at 2.5TBps [7]. This research explored ways authoritative DNS resolvers can be prevented from being used in DDOS attacks without having the victim shoulder the full cost of absorbing malicious traffic. Thus, helping to reduce the possible amount of malicious traffic that ends up in the victim's infrastructure. We also identify the ways of detecting and stopping spoofed packets and carry out an analysis on them.

4. Methodology

Our solution makes the hypothesis that packets from the same AS transverse the same last hop AS before entering the destination AS [7]. Based on this, the following steps were taken:

Snort was set up as an IDS to detect incoming DNS request packets with the content type any and alert.

Source AS paths were mapped to prefixes of addresses gotten from snort. These AS numbers can be gotten locally from operators' routers or globally from Internet Routing Registry (IRR) databases (www.irr.net). In our case, they were gotten locally.

Mismatched AS numbers in traffic are detected and dropped.

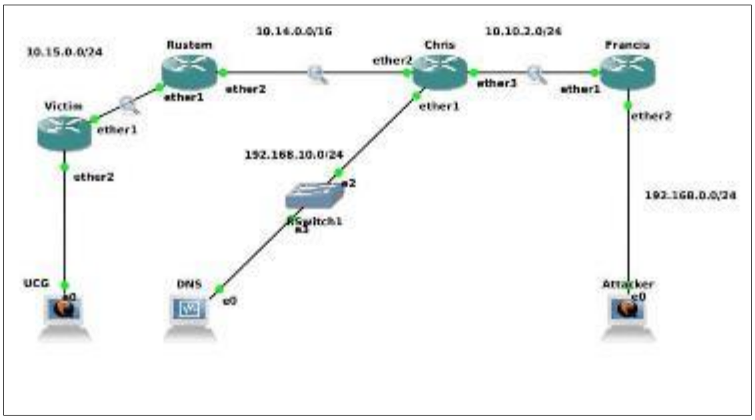

Figure 2 Network Topology

The following AS to prefix mapping was constructed using the sample network below for the expected entry and exit AS numbers for traffic coming to the DNS server.

Victim - AS64524

Attacker - AS64530

Intermediate ISP - AS64523

DNS Server - AS64516

Table 1 Autonomous System Mappings to Interfaces

Prefix	AS	Expected Entry AS/Interface for requests	Expected Exit AS/Interface for responses
10.15.0.0/24	AS64524	Via AS64523 on Ether2	Via AS64523 on Ether2
10.10.2.0/24	AS64530	Via AS64530 on Ether3	Via AS64530 on Ether3

Following the above and a series of packet captures [8], the transit route of a legitimate DNS traffic from the victim would be as follows:

Requesting endpoint in AS64524 -> AS64523 -> AS64516 ->DNS server

Response from DNS in AS64516 -> AS64523 -> AS64524 -> Requesting device

5. Results

Packet captures of legitimate and malicious requests were made [8]. From the packet captures, the table below represents what a legitimate DNS request was like:

Table 2 Legitimate DNS Request Route

Query Origin	DNS Server	Response Receiver
192.168.0.254(AS64530)	10.10.2.1 (AS64516)	192.168.0.254(AS64530)

From the capture of an unrestricted malicious packet capture [8], the transit route was as follows:

Malicious request in AS64530 -> AS64516 -> DNS server -> AS64523 -> AS64524 -> Victim

Table 3 Malicious DNS Request Route

Query Origin	DNS Server	Response Receiver
192.168.0.254(AS64530)	10.10.2.1 (AS64516)	10.15.0.1(AS64524)

These spoofed packets can be dropped by the edge devices of the AS where the DNS server is domiciled, causing one less device to be available for a DDOS attack on the victim. To improve the ability to detect such spoofed packets, the anti-spoofing measures should be considered and located as near to the source as possible; in our case, it is an ingress router that serves as ingress packet filtering. There are a great many precise guidelines to exclude the possibility of IP spoofing, and we have preferred one that relies on, in simple terms, checking if the entry route is not the same as the exit route, otherwise, it drops the packet.

The Mikrotik router solution provides a Strict uRPF (unicast Reverse-Path Forwarding) software feature. At its core (in Strict mode), the router goes through a two-step process: inspection, which involves looking for the source IP address in its routing table, and verification, which involves some checking whether the router uses the same interface to link up this source IP or not. These checks can be enabled by running the below on mikrotik routerOS 6.47:

```
ip settings set rp-filter=strict
```

Anti-spoofing can also be implemented using ingress packet filtering, where the packet's entry prefix is matched to the expected AS-Path for that prefix; failing that, it is dropped. A sample ingress filtering rule using the MikroTik router OS is as follows:

```
ip firewall filter add action=drop chain=forward src-address=!as-path in-interface=!as-interface
```

On enabling the strict uRPF, packets got to the border router of the DNS server but were dropped due to the mode of operation of the uRPF. Similarly, the ingress packet filtering also dropped packets arriving at the border router of the DNS network.

6. Conclusion

While using the BGP AS paths is a viable approach to detecting spoofed DNS packets, in situations where the attacker is in the same BGP as the victim, identifying the request as spoofed will fail. Implementing the strict uRPF mode for the rejection of spoofed packets is a better alternative. However, this would require rebooting the border routers. In both cases of uRPF strict mode filtering or ingress filtering using that AS-path, there is a need for the implementation to be done in tandem with the BGP provider. A recommendation for further research will be to implement automated learning of BGP as paths, their expected prefixes, and ingress interfaces.

Compliance with ethical standards

Disclosure of conflict of interest

No Conflict of interest to be disclosed.

References

[1] Arbor Networks (2020). Worldwide Infrastructure Security Report, Volume V”, http://www.arbornetworks.com/report (Accessed December 3, 2020). https://kapost-files-prod.s3.amazonaws.com/published/569e85ff426d9e582400000a/wisr-report.pdf https://www.cloudflare.com/learning/ddos/dns-amplification-ddos-attack/

[2] Kambourakis, G., Moschos, T., Geneiatakis, D., & Gritzalis, S. (2007). Detecting DNS Amplification Attacks. Accessed December 3, 2020 https://www.researchgate.net/publication/220816528_Detecting_DNS_Amplification_Attacks

[3] Kim, S., Lee, S., Cho, G., Ahmed, M.E., Jeong, J., & Kim, H. (2017). Preventing DNS Amplification Attacks Using the History of DNS Queries with SDN. In: Foley S., Gollmann D., & Snekkenes E. (eds) Computer Security – ESORICS 2017. ESORICS 2017. Lecture Notes in Computer Science, vol 10493. Springer, Cham. https://doi.org/10.1007/978-3-319-66399-9_8 (Accessed December 7, 2020)

[4] Verma, S., Hamieh, A., Huh, J., Holm, H., Rajagopalan, S., Korczynski, M., & Fefferman, N. (2020). Stopping Amplified DNS DDoS Attacks through Distributed Query Rate Sharing https://www.researchgate.net/publication/304778484_Stopping_Amplified_DNS_DDoS_Attacks_through_Distributed_Query_Rate_Sharing(Accessed December 7, 2020)

[5] Vaidyanathan, R., Ghosh, A., Cheng, Y., Yamada, A., & Miyake, Y. (2020). On the use of BGP AS numbers to detect spoofing. (Accessed December 7, 2020) http://intrusion-detection.org/papers/Ravi10BGPSpoofDetection.pdf

[6] Ghosh, A. (2017). InFilter: Predictive Ingress Filtering to Detect Spoofed IP Traffic”, https://patents.google.com/patent/US8925079

[7] Packet captures https://github.com/xrisbarney/rp1-iu.