Secure Consensus Generation with Distributed DoH

Philipp Jeitner[*], Haya Shulman[†] and Michael Waidner[*][†]

[*]Technical University of Darmstadt, philipp.jeitner@sit.tu-darmstadt.de

[†]Fraunhofer Institute for Secure Information Technology SIT, {haya.shulman,michael.waidner}@sit.fraunhofer.de

Abstract—**Many applications and protocols depend on the ability to generate a pool of servers to conduct majority-based consensus mechanisms and often this is done by doing plain DNS queries. A recent off-path attack [1] against NTP and security enhanced NTP with Chronos [2] showed that relying on DNS for generating the pool of NTP servers introduces a weak link. In this work, we propose a secure, backward-compatible address pool generation method using distributed DNS-over-HTTPS (DoH) resolvers which is aimed to prevent such attacks against server pool generation.**

I. INTRODUCTION

Majority-based consensus mechanisms ensure security by comparing the responses from a set of servers taken from a larger pool. The security guarantee is given by statistical analysis and assumptions on the distribution of benign and malicious servers in that pool. To generate the pool, some applications rely on lists of servers signed by central authorities (e.g., Tor [3]) but others like Chronos [2] and most Cryptocurrencies [4] just rely on the DNS. Recently [5], [6], [1] showed that this reliance on DNS introduces a weak link and allows an attacker to attack DNS instead of the application protocol. Specifically [1] showed that Chronos is vulnerable to off-path attacks, by subverting DNS security, while Chronos [2] guarantees security against strong on-path attackers. In this work we propose an approach for enhancing the DNS layer by distributing the queries for Network Time Protocol (NTP) servers to a set of resolvers. Our proposal, in tandem with Chronos, guarantees security to the NTP ecosystem.

Previous proposals on distributing DNS [7], [8], focused on the DNS nameservers, our goal in this work is to distribute the DNS resolvers. We propose a majority based protocol to securely generate a pool of servers via DNS which does not require any changes to the existing protocols nor infrastructure. Our proposal leverages majority decision supported by secure channels to a list of trusted DNS-over-HTTPS (DoH) [9] resolvers. The security is guaranteed since distributing the resolvers ensures that the queries are sent from different locations in the Internet. Hence even if some of the paths or servers are corrupted or controlled by the attacker, the requests from the paths that are not under attacker's control guarantee security. The security of our proposal assumes that the realistic attackers are limited in their capabilities: the attacker can control some of the servers and some of the links in the Internet but not all the Internet servers and links. This is a realistic assumption and applies even to the strong government sponsored attackers.

Our proposed system is shown in Figure 1. The mechanism works by querying the NTP pool domain through a list of distributed DoH resolvers (Step 2 in Figure 1) and combines the results to generate a server address pool containing a fraction of at least x (e.g. $\frac{1}{2}$) benign servers assuming that an attacker can only attack a fraction of x of the DoH resolvers.

Fig. 1. System overview

II. SECURE SERVER POOL GENERATION

The operation mode targeted in this work is that of applications aiming to generate a trustworthy pool of servers, the application can connect to. For the application to be secure, this pool must include a fraction of at least x benign servers. This is the operation mode Chronos [2] (and to some degree, traditional Network Time Protocol (NTP) clients) operate in. In this operation mode, the application is aware that not all servers in the generated pool are trustworthy, but the number of these is limited to be lower than the fraction of $1 - x$ of the total servers which addresses are included in the DNS response. As this operation mode is specific to server pool generation, it does only support address lookups[1]. Assuming a fraction x of secure DoH resolvers, this property can be fulfilled by querying all DoH resolvers, truncating the list of results from each resolver to the number of results from the resolver with the shortest list and then returning the the combination of these truncated lists to the application. This process is shown in Algorithm 1.

Ensuring that *all* of the servers in a returned DNS query are benign can be performed via a classic majority-vote on each of the returned addresses, e.g., the majority DNS resolver only includes an address in the final response, if it is given by a

---

[1]If Dual-stack operation needs to be supported, it depends on the application, whether the property of a honest majority of servers needs to be fulfilled for the union of A and AAAA records or for both sets individually.

Algorithm 1: Secure server pool generation lookup

Input: domain *(query domain)*
Input: resolvers *(list of DoH resolvers)*
Input: x *(Fraction of assumed non-attacked resolvers, eg. $\frac{1}{2}$)*
results = [] , lengths = [] , addresspool = []
for *res in resolvers* **do**
    r = query(res, domain)
    results.append(r)
    lengths.append($\|r\|$)
end
truncatelength = min(lengths[])
for *r in results* **do**
    addresspool.add(truncate(r, truncatelength))
end
return addresspool

---

majority of the DoH resolvers. This is not required for Chronos since it can handle a minority of malicious servers by itself.

We propose to deploy our mechanism without changing the DNS infrastructure, offering a standard-compatible DNS-resolver interface. The design is in Algorithm 1.

III. SECURITY ANALYSIS

In this section we give an analysis of the probability that responses from a fraction of x DoH resolvers can be trusted assuming that (1) all resolvers are benign and (2) a probability that an attacker can successfully perform an attack against any of the resolvers independently with a probability of p_{attack}.

a) Fraction of DoH resolvers x required for a successful attack: For a successful attack against the DNS-using application, we assume the attacker needs to control at least y (e.g., $\frac{1}{2}$) of the addresses in the server pool used by that application. Assuming a number of used DoH resolvers N and the length of the shortest[2] response list K, we get a total number of $N \cdot K$ addresses, where every resolver controls only K. To control more than y addresses the attacker therefore needs to attack at least x DoH resolvers fulfilling $yK \leq xK$, which gives us $x \geq y$. The attacker therefore is required to successfully attack at least $x = y$ of the DoH resolvers used.

b) Probability of attacking at least x of DoH resolvers: Assuming a number of used DoH resolvers N, the probability of attacking a fraction of at least x is p_{attack}^M with $M \leq \lceil xN \rceil$. Even when the only 3 DoH resolvers are used, this means that the probability of a successful attack which requires a malicious majority ($x \geq \frac{2}{3}$) is reduced significantly (p_{attack}^2). Furthermore, by increasing the number of used DoH resolvers, a successful attack becomes exponentially less probable, effectively giving the same type of asymptotic advantage over an attacker which is achieved by increasing the key size in a traditional cryptosystem.

IV. DISCUSSION AND FUTURE WORK

For our approach to work, the application must handle multiple instances of the same address in the response as

individual servers. Without this assumption, an attacker might overwhelm a majority of DoH resolvers if those resolvers do not give mutually exclusive sets of responses. Furthermore, our proposal is only meant to provide security on the DNS layer, namely we guarantee that even if some of the responses from the DNS resolvers can be attacked, the majority response is correct, i.e., the NTP servers returned in a response are benign. This may however not always be the case: attackers can try to join the NTP pool themselves and operate malicious NTP servers. Hence, for the overall NTP ecosystem to maintain security a distributed mechanism on the NTP layer should also be used, such as the Chronos proposal [2].

V. CONCLUSION

In this work we propose a distributed generation of NTP server pool using multiple DoH DNS resolvers. We show that using our proposal mitigates the off-path attacks against plain NTP as well as against Chronos enhanced NTP [1], and guarantees security even against realistic on-path Man-in-the-Middle (MitM) attackers that control some (but not all) of the Internet paths and DNS resolvers. Our proposal is easy to integrate and does not require changes to the DNS infrastructure, it is backward compatible with standard DNS nameservers.

REFERENCES

[1] P. Jeitner, H. Shulman, and M. Waidner, "The Impact of DNS Insecurity on Time," in *2020 50th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, 2020, (Forthcoming).

[2] O. Deutsch, N. R. Schiff, D. Dolev, and M. Schapira, "Preventing (Network) Time Travel with Chronos," in *Proceedings 2018 Network and Distributed System Security Symposium*, San Diego, CA, 2018.

[3] The Tor Project. Tor directory protocol, version 3. https://github.com/torproject/torspec/blob/master/dir-spec.txt, accessed 2020-02-10.

[4] A. F. Loe and E. A. Quaglia, "You shall not join: A measurement study of cryptocurrency peer-to-peer bootstrapping techniques," in *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, ser. CCS '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 2231–2247. [Online]. Available: https://doi.org/10.1145/3319535.3345649

[5] A. Herzberg and H. Shulman, "Fragmentation Considered Poisonous: or one-domain-to-rule-them-all.org," in *IEEE CNS 2013. The Conference on Communications and Network Security, Washington, D.C., U.S.* IEEE, 2013.

[6] M. Brandt, T. Dai, A. Klein, H. Shulman, and M. Waidner, "Domain Validation++ For MitM-Resilient PKI," in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2018, pp. 2060–2076.

[7] C. Cachin and A. Samar, "Secure distributed dns," in *International Conference on Dependable Systems and Networks, 2004*. IEEE, 2004, pp. 423–432.

[8] E. S.-J. Swildens, R. D. Day *et al.*, "Domain name resolution using a distributed dns network," May 25 2010, uS Patent 7,725,602.

[9] P. Hoffman and P. McManus, "Dns queries over https (doh)," Internet Requests for Comments, RFC Editor, RFC 8484, October 2018.

---

[2]We use the shortest list, because this prevents attacks where the attacker seeks to overwhelm resolvers by including more responses than usual (See attack against Chronos [1]). This comes at the cost of allowing DoS attacks when the attacker includes no responses at all in his poisonous response.