ASSURED

SECURITY CONSULTANTS

Report

Mullvad DNS over HTTPS server audit

Wictor Olsson, Emilie Barse, Benjamin Svensson, Johanna Abrahamsson

Project	Version	Date
MUL001	v1.0	2021-02-23

ASSURED
SECURITY CONSULTANTS

REPORT

Project	**Version**
MUL001	v1.0

Date
2021-02-23

Executive summary

Assured was tasked to perform a whitebox system audit of Mullvads DNS over HTTPS servers. The audit focused on configuration in regards to privacy, attack surface reduction and security best practices. The server deployment and configuration displayed a good level of security in general. At the time of the audit, the exposed services were running at a good patch level, with no known vulnerabilities. The most notable findings during the audit was related to a misconfiguration of the DNS service (Unbound), NTP service and iptables egress/ingress configuration, these issues were promptly resolved by the Mullvad team and verified during the audit period.

REPORT

Project	**Version**
MUL001	v1.0

Date
2021-02-23

Contents

ASSURED
SECURITY CONSULTANTS

REPORT

Project	**Version**
MUL001	v1.0

Date
2021-02-23

1 Introduction

1.1 Background

Assured AB (Assured) was contracted by Mullvad to perform a audit of their new servers running DNS over HTTPS.

1.2 Constraints and disclaimer

This report contains a summary of the findings found during the project period. This report should not be considered as a complete list of all possible vulnerabilities, security flaws and/or misconfigurations.

1.3 Project period and staffing

Assured started the project on 2021-02-08 and finished on 2021-02-18.

This report was last reviewed on 2021-02-23.

Involved in the audit was Assured consultants Wictor Olsson, Emilie Barse, Benjamin Svensson and Johanna Abrahamsson.

1.4 Risk rating

In this report we have assessed the severity of issues and identified vulnerabilities. The levels of severity are rated according to the OWASP Risk Rating Methodology [1].

Table 1: OWASP Risk Rating overall severity model

Overall risk severity				
Impact	HIGH	Medium	High	Critical
	MEDIUM	Low	Medium	High
	LOW	Note	Low	Medium
		LOW	MEDIUM	HIGH
	Likelihood			

As Table 1 visualizes, the overall risk assessment is determined from a combined likelihood and impact of an identified vulnerability or security issue. A value from 0 to 9 is assessed for each variable, where 0-2 is determined LOW, 3-5 is MEDIUM and 6-9 is HIGH.

Likelihood is dependant on attributes related to threat actors and the identified vulnerability, with factors such as: the skill level and motivations of the threat agents; how easily the vulnerability can be found and exploited, and; how likely an exploit may be detected.

Impact depends on technical and business factors, such as: level of loss of confidentiality, integrity, availability and accountability; potential financial damage; potential brand damage, and; potential violations of privacy.

Please note that the severity assessment is made by Assured consultants and ratings may differ from the resource owners' ratings.

2 Scope and methodology

2.1 Scope

2.1.1 Audit of Mullvad DNS over HTTPS servers

The scope was to perform mainly a whitebox configuration review of the targeted system(s).

The main areas of interest were the following:

- No logs or information leakage that exposes the clients using the server
- No known vulnerabilities in the exposed services
- Service and system configuration should follow best security practices
- No unnecessary services or applications running

2.2 Methodology

2.2.1 System audit

Access to two systems and their deployment scripts were made available to Assured consultants.

The servers were running the DNS service Unbound [2], the intention was to only expose the DNS over TCP/TLS and DNS over HTTPS protocol stacks. The servers were also running Quagga for BGP which was restricted through iptables to only accept ingress/egress traffic to or from specific hosts. Other services running were a SSH server (OpenSSH) and an NTP server (default Ubuntu NTP server).

The deployment scripts were reviewed for issues and to get a general idea of the target systems. The systems were then accessed through SSH and analyzed primarily manually with the aid of some auditing scripts.

The following areas were audited:

- Users and groups, file permissions
- Filesystem structure, installed and running services, packages installed
- Listening sockets and iptables configuration
- Patch level of exposed services and system
- Exposed services configurations (unbound, ntpd, quagga, ssh)
- Logs and logging configuration
- Linux kernel settings
- TLS stack configuration of Unbound (DoT/DoH)

Dynamic testing was done using a lab setup of Unbound with an almost identical configuration as in the target system:

- Check content of upstream DNS TLS traffic using sslsplit for different kinds of DoT and DoH DNS requests (including edns0 client subnet requests).
- Limited blackbox robustness testing (fuzzing) was executed on the Unbound DNS over HTTPS stack implementation with the tool erlamsa.

2.3 Limitations

Only remote access to the target systems and a limited period of time.

3 Observations

3.1 Mullvad DNS over HTTPS servers

These are the findings regarding the Mullvad DNS over HTTPS servers configuration.

3.1.1 Low MITIGATED Unbound listening socket misconfiguration

Current status

The Unbound DNS server still listens to DNS over UDP on port (443/udp and 853/udp), but the local firewall was changed after reporting this finding. The local firewall is now configured to block requests to the UDP ports and the finding is considered to be mitigated.

Original finding

Unbound is a validating, recursive, caching DNS resolver. To help increase online privacy, Unbound supports DNS-over-TLS and DNS-over-HTTPS which allows clients to encrypt their communication. The Mullvad Unbound setup is intended to make use of and expose the DoT and DoH protocol stacks for clients to securely perform queries. This means exposing only the intended listening sockets, in this case 443 (DoH) and 853 (DoT).

However, during active system analysis it was discovered that Unbound was exposing its regular UDP based DNS stack on ports 443 and 853 as well, a misconfiguration. This was later confirmed with Mullvad to not be the intended scenario.

This will open up the Unbound resolver to attack and misuse. Examples of such misuse would be a DNS amplification attack. Another would be the potential threat of an attacker exploiting unknown implementation issues in Unbound's UDP DNS protocol stack.

It seems to be a feature limitation currently in how Unbound binds its sockets relative to specified interface and if it should be UDP or TCP, see: https://github.com/NLnetLabs/unbound/issues/143

One solution would be to utilize the ”do-tcp” or ”do-udp” configuration parameters, but this would disable one of the stacks altogether.

To minimize the attack surface, its currently recommended to apply iptables to block the UDP based DNS stack sockets (443/udp and 853/udp) until Unbound implements a feature that makes it possible to choose if a listening socket should be UDP or TCP without completely turning off either stack for selected interfaces.

3.1.2 Low FIXED Iptables should be more restrictive

Current status

The iptables local firewall rules were reconfigured and verified during the test, after reporting this finding. The new iptables local firewall rules are configured strictly to block any unnecessary traffic, both for ingress and egress traffic, and both for ipv4 and ipv6. The finding is considered to be fixed.

Original finding

The iptables local firewall rules for the servers are not following best practices for a hardened server. The iptables configuration is using default policy of ACCEPT on both the INPUT and OUTPUT chain and only has a few blacklisting rules for ssh and bgpd. Hence, Mullvad has used a blacklisting policy instead of whitelisting.

For a hardened server, the iptables rules should be configured to have policy DENY and allow only the necessary traffic both inbound and outbound. A whitelisting firewall policy would have prevented ntpd from listening on external interfaces, which was not intended according to Mullvad.

The server exposes the external services ssh (1022/tcp), quagga-bgpd (179/tcp),ntpd (123/udp) and doh/dot from unbound (443/udp, 443/tcp, 853/udp, 853/tcp). Only bgpd and ssh are covered by the iptables rules currently. Access to bgpd is filtered based on IP address. Access to ssh is only filtered to check the incoming interface. However, connections to ssh is filtered on IP address in a separate firewall.

The recommendation is to configure the local firewall with a stricter ruleset. In server deployments like this, with a very limited amount of services, a whitelisting policy should be fairly easy to implement. Only allow unrestricted ingress connections for unbound DoT/DoH, and possibly IP restricted ingress connections for ssh, bgpd and ansible. Only allow egress connections for related traffic and to necessary servers (or services), such as external DNS servers, NTP servers, bgpd servers, and software updating servers.

3.1.3 Note FIXED Ntpd listening on all interfaces

Current status

The ntpd configuration was changed and verified during the test. The ntpd service now only listens on the intended interfaces. The finding is considered to be fixed.

Original finding

During active analysis of the systems, it was found that the NTP service running on the system was listening on all interfaces (a number of different public IP addresses). The NTP service was supposed to be limited to listen on one specific public IP address.

Exposing a service on every interface will in many cases lead to an increased risk of the application being compromised or abused. For example, if firewall rules in the local firewall or an external firewall is configured assuming that it only listens on one IP, it may expose the service in an unintended manner.

Based on the running configuration, attempts were made to limit the NTP service exposure by listening on a specific public IP address (interface IP obfuscated with X).

Example 1: ntpd configuration

```
interface listen 127.0.0.1
interface listen ::1
interface listen X.X.X.X
```

However, without the ”interface ignore wildcard” setting configured, ntpd will still listen on all available interfaces.

To avoid unnecessary exposure of the service its recommended to change the configuration accordingly.

3.1.4 Note Apparmor for exposed services

Neither Unbound nor Quagga is running Apparmor profiles. For a hardened system, Apparmor can be used for an extra layer of security to protect the exposed services. This may protect the service and the server from exploitation of unknown vulnerabilities.

Custom Apparmor profiles can be created for the exposed services to restrict their permissions to the filesystem and network. Especially, the Unbound service is not supposed to write logs to the filesystem and could benefit from a rather strict Apparmor profile for file access.

The recommendation is to create a custom Apparmor profile for Unbound and possibly also for Quagga.

3.1.5 Note Unnecessary installed software

There are some installed packages, such as tcpdump, netcat and nmap, on the server(s), which are not necessary for the functionality and also can be useful for an attacker who gets code execution on a server. There is also compilation software, such as gcc installed. This is used for building the unbound package. For a hardened production server, it is considered best practice to remove this kind of software.

In this case, tcpdump may be the most dangerous software, because it can be used to view and log client connections and reveals the client IP. However, tcpdump requires special privileges to be run.

Removing the compilation software would require using a separate build system, and only install the compiled binaries on the server. This also has other advantages, such as central control of the installed software and possibility to use security scanning tools at build time.

It is recommended to remove the unnecessary software and to use security monitoring, to alert if software such as tcpdump is installed.

4 Conclusions and recommendations

The conclusion of the audit is that the system has a good security level, and that the Unbound DNS service is properly configured for protecting the privacy of the clients using it. Unbound is configured to avoid creating or saving any logs and is running on an encrypted filesystem.

The Unbound configuration was examined in detail for security issues and dynamic testing was done to double-check that client information was not logged or forwarded to the upstream DNS server. The TLS stack configuration was also reviewed and tested. The tests did not reveal any issues, except that it was discovered that the Unbound service listens to DNS over UDP on port 443 and 853, which was not intended and expose an unnecessary attack surface. This was mitigated during the test after reporting the finding, by configuring the local firewall.

The Unbound services has been audited for security issues previously [3], but not with focus on the DNS over HTTPS (DoH) stack which is a recently implemented feature. Limited blackbox fuzzing of the DoH stack was done during this test, and did not show any issues. A thorough analysis of the Unbound DoH implementation was not covered by this audit but would be recommended in the future.

The server exposes the external services ssh (1022/tcp), quagga-bgpd (179/tcp), ntpd (123/udp) and doh/dot from unbound (443/udp, 443/tcp, 853/udp, 853/tcp). However, access to ssh and ntpd is restricted by an external firewall, and quagga-bgpd is restricted by local firewall rules. On a hardened server, it is best practice to configure the local firewall rules strictly for both inbound and outbound access, which was not the case. After reporting this finding, the local firewall rules were reconfigured during the test to block any unnecessary traffic, and the finding is considered to be fixed.

The analysis of the configuration of the external services did not reveal any issues, except that the NTP service was misconfigured to listen on all interfaces. This was also fixed during the test.

The server itself is relatively well limited in functionality and installed software. However, some further hardening can be done by removing unnessesary software, such as tcpdump. Also, it may be beneficial to use a separate build system and to avoid building the software on the production server. Further hardening can be done by using Apparmor profiles for the exposed services.

Assureds recommmendations are:

- (Fixed) Configure the local iptables firewall to restrict inbound and outbound traffic to only the necessary ports and servers.
- (Fixed) Change the configuration of ntpd to listen to only the intended IP by using the ”interface ignore wildard” setting.
- Consider using Apparmor to protect the Unbound and Quagga services
- Remove unnecessary software on the servers
- Consider using a separate build system with build time security scanning
- Check local iptables rules also for other systems and servers to see if stricter rules can be implemented

References

[1] OWASP, "OWASP Risk Rating Methodology." https://www.owasp.org/index.php/OWASP_Risk_Rating_Methodology, 2019.

[2] NLNet Labs, "NLnet Labs Unbound DNS server." https://www.nlnetlabs.nl/projects/unbound/about/, 2021.

[3] X41 D-SEC GmbH, "Source Code Audit on Unbound DNS Server for NLnet Labs." https://ostif.org/wp-content/uploads/2019/12/X41-Unbound-Security-Audit-2019-Final-Report.pdf, 2019.