Pentest-Report Mullvad VPN Relay-Infrastructure 06.2024

Cure53, Dr.-Ing. M. Heiderich, J. Larsson, M. Elrod

Index

Introduction

> *"Mullvad VPN AB is owned by parent company Amagicom AB. The name Amagicom is derived from the Sumerian word ama-gi – the oldest word for "freedom" or, literally, "back to mother" in the context of slavery – and the abbreviation for communication. Amagicom stands for "free communication"."*

From https://mullvad.net/en/about

This report describes the results of a penetration test and source code audit conducted by Cure53 against the Mullvad WireGuard and OpenVPN relay infrastructure.

To give some context regarding the assignment's origination and composition, Mullvad VPN AB contacted Cure53 in January 2024. The test execution was scheduled for June 2024, namely in CW23 / CW24. A total of sixteen days were invested to reach the coverage expected for this project, and a team of three senior testers was assigned to its preparation, execution, and finalization.

The methodology conformed to a white-box strategy, whereby assistive materials including sources and all further means of access required to complete the tests were provided to facilitate the undertakings.

The work was split into two separate work packages (WPs), defined as:

- **WP1**: Penetration-tests & source code audits against Mullvad WireGuard relays
- **WP2**: Penetration-tests & source code audits against Mullvad OpenVPN relays

Note that this was not the first time the Mullvad VPN infrastructure complex was tested by Cure53. It was already the focus of several preceding audits, the most recent of which was held in November 2020 (see MUL-03).

All preparations were completed in late May and early June 2024, specifically during CW22, to ensure a smooth start for Cure53. Communication throughout the test was conducted through a dedicated and shared Slack channel, established to combine the teams of Mullvad and Cure53. All personnel involved from both parties were invited to participate in this channel. Communications were smooth, with few questions requiring clarification, and the scope was well-defined and clear. No significant roadblocks were encountered during the test. Cure53 provided frequent status updates, shared their findings, and offered live reporting through the aforementioned Slack channel.

The Cure53 team achieved good coverage over the scope item, and identified a total of five findings. Of the five security-related discoveries, two were classified as security vulnerabilities, and three were categorized as general weaknesses with lower exploitation potential.

The overall number of findings made during this testing can be seen as a small amount, and this can be interpreted as a positive sign. In addition, this audit showed a significant decrease in findings compared to the preceding iteration (MUL-03), which showcases the commitment of the Mullvad team to properly securing and strengthening the underlying infrastructure of its VPN application.

All in all the Cure53 team concluded that the Mullvad system appears to be well-designed, well set up, and generally in exemplary condition. Nevertheless, it is recommended to swiftly resolve all of the issues detailed herein, as well as to continue to improve and test the application and its infrastructure, in order to ensure that a good level of security is maintained.

The report will now shed more light on the scope and testing setup, and will provide a comprehensive breakdown of the available materials. Next, the report will detail the *Test Methodology* used in this exercise. This chapter will show which areas of the software in scope have been covered, and what tests have been executed, despite the limited number of findings made during the course of the exercise. Following this, the report will list all findings identified in chronological order, starting with the *Identified Vulnerabilities*, and followed by the *Miscellaneous Issues* unearthed. Each finding will be accompanied by a technical description, Proof-of-Concepts (PoCs) where applicable, plus any fix or preventative advice to action.

In summation, the report will finalize with a *Conclusions* chapter in which the Cure53 team will elaborate on the impressions gained toward the general security posture of the Mullvad WireGuard and OpenVPN relay infrastructure.

Scope

- **Penetration-tests and source code audits against Mullvad VPN relay-infrastructure**
  - **WP1:** Penetration-tests & source code audits against Mullvad WireGuard relays
    - **Host:**
      - *se-got-wg-999.relays.stagemole.eu*
      - SSH: 1022/tcp
    - **Repository:**
      - *infrastructure-ansible-release-2024.3*
  - **WP2:** Penetration-tests & source code audits against Mullvad OpenVPN relays
    - **Host:**
      - *se-got-ovpn-999.relays.stagemole.eu*
      - SSH: 1022/tcp
    - **Repository:**
      - *infrastructure-ansible-release-2024.3*
  - **Test-supporting material was shared with Cure53**
  - **All relevant sources were shared with Cure53**

Test Methodology & Coverage

This section outlines testing methodology and coverage, focusing on the steps taken to evaluate the current security posture of Mullvad's relay infrastructure. The following section highlights significant tasks performed during the assessment of the components and configuration of Mullvad's relay infrastructure. This emphasizes the areas Cure53 testers focused on, particularly concerning the relay nodes, and clarifies the coverage and methodology applied during this phase of the assessment.

- This assessment began with evaluating the Ansible repository provided by Mullvad. This repository contained numerous playbooks and associated roles used to configure and deploy VPN components to Mullvad's infrastructure. By analyzing the deployment process configurations within the supplied repository, Cure53 quickly gained insight into the specific configurations and components used by Mullvad.
- The Ansible configuration, roles, and procedures were analyzed for potential insecure defaults or weak instrumentation that could be exploited by both remote and local attackers. The repository left a good impression in terms of structure and configuration. Cure53 observed several complete certificates, noting that all the detected public / private keys belonged exclusively to the staging environments. No production certificates were observed in a similar context.
- Additionally, the configuration for encrypting sensitive parameters relies on GPG and Vaults, which were found to be securely handled and correctly implemented.
- While assessing the deployments on the hosts within the scope of this engagement, Cure53 observed that all playbooks and corresponding roles were locally stored. This was deemed an unnecessary exposure, as not all components in the Ansible repository are used for the context of the deployment. Consequently, a ticket regarding Ansible hardening was filed (see MUL-04-002).
- Next, two staging hosts were made available for Cure53 in order to validate the deployed configuration and security concepts used on each host. The hosts consequently ran OpenVPN and WireGuard relay configuration. These hosts were not part of the production infrastructure and could not be reached directly without instrumenting the Mullvad applications with additional parameters.
- In order for Cure53 to assess communication from the originating client all the way to the VPN infrastructure, additional allow-listing was temporarily enabled by Mullvad. This had to be manually added by the Mullvad team, since this practice is normally not acceptable.
- The OpenVPN and WireGuard configuration running on the assessed hosts were analyzed, looking for configurations that could facilitate potential leaks and insecure configurations that could be leveraged from either a remote or local attacker. No issues were spotted during this phase of the assessment.
- The network configuration on each of the hosts within the scope was analyzed, revealing a notable attention to detail. All iptables rules are set using Source, Destination, and NAT, demonstrating a strong security awareness and careful planning.

- Network isolation concepts were assessed by looking for exposed local networks, both IPv4 and IPv6. In addition to this, internal services such as DNS and broadcast traffic were assessed, looking for potential leaks and information leaks that could be leveraged by a local attacker.
- VPN leak tests were conducted to identify any potential data leaks. This investigation aimed to detect various types of leaks, including DNS leaks, IP address exposure, and any other data that might inadvertently be transmitted outside of the secure VPN tunnel. The tests were designed to ensure the integrity and confidentiality of user data, verifying that the VPN properly shields all sensitive information and maintains robust security standards. No issues were identified during this phase.
- The hardening efforts on the hosts within the scope were analyzed, and overall, the measures taken using sysctl and AppArmor profiles left a positive impression. However, as a minor critique, Cure53 filed the ticket MUL-04-003, recommending the removal of common binaries that could assist a local attacker in post-exploitation activities, or in establishing persistence on the local hosts.
- Local privilege escalation vectors were analyzed, looking for potential vectors that could be leveraged by a non-privileged local attacker. This led to the discovery of redundant sudoers configuration filled as MUL-04-001. Whilst assessing the sudoers and permission delegation topology, a local privilege escalation vector was found within a service running in systemd; this was filled as MUL-04-004.
- During this phase of the engagement, it was noted that a local attacker could potentially bypass the authorized login checks. This issue was documented as MUL-04-005.
- As a final step in assessing the integrity of the hosts within the scope, efforts were made to evaluate out-of-band management systems, and to detect the potential presence of hypervisors. This involved a detailed examination of the management interfaces and infrastructure to ensure that no additional layers, such as hidden hypervisors, could compromise the security or integrity of the environment. No issues were identified during this phase.

Identified Vulnerabilities

The following section lists all vulnerabilities and implementation issues identified during the testing period. Notably, findings are cited in chronological order, rather than by degree of impact, with the severity rank offered in brackets following the title heading for each vulnerability. Furthermore, all tickets are given a unique identifier (e.g., MUL-04-001) to facilitate any future follow-up correspondence.

MUL-04-004 WP1/2: LPE for user *mullvad-local-checks* to root (*Low*)

Note: *This finding was in an additional hardening and security feature. The bug itself should not be seen as an indication of bad system setup or bad engineering, but the extra surface provided by some hardening features can lead to more vulnerabilities, so the ticket was added to the report.*

Fix note: *The problem was fixed during testing by the Mullvad team. The ownership of the file has been corrected to be owned by the root user, preventing the mullvad-local-checks user from modifying it.*

A local privilege escalation (LPE) vulnerability was discovered involving the *mullvad-local-checks* and the root user. The crontab for root executes a file that is not owned by root, but by said lower-privilege user. This allows *mullvad-local-checks*, the lower-privilege user, to modify the script, enabling them to execute arbitrary code with root privileges. This vulnerability can be exploited to gain unauthorized root access.

Affected files:
/etc/systemd/system/mullvad-rescue-local-resolver.service

Affected code:

```
User=root
ExecStart=/opt/local_checks/rescue-local-resolver
```

PoC:

```
martin@se-got-wg-999:~$ ls -alh /opt/local_checks/rescue-local-resolver
-rwxr-x--- 1 mullvad-local-checks mullvad-local-checks 299 May 27 18:09
/opt/local_checks/rescue-local-resolver
```

To mitigate this issue, it is advisable to align file ownership and process ownership, thereby preventing any owner boundaries from being breached.

MUL-04-005 WP1/2: User can hide from check-unauthorized-logins *(Medium)*

Note: *Please see the note for MUL-04-004 regarding vulnerabilities in hardening features as the same applies to this discovery.*

Fix note: *The problem was fixed during the test by the Mullvad team. The script has been updated to correct the username regex to prevent unauthorized users from bypassing the login check.*

The script designed to check for unauthorized logins contains a flaw in its username validation process, allowing it to be bypassed. It uses a regex to compare logged-in usernames against a list of authorized usernames. However, the script incorrectly marks unknown usernames as authorized if they are a substring of any valid username. This flaw allows unauthorized users to bypass the login check and maintain access undetected, providing a false sense of security.

Affected file:
/opt/local_checks/check-unauthorized-logins

Affected Code:

```
AUTHORIZED_USERS_PATTERN="some|name|other|another|yetanother|"
LOGINS=$(last --since ${LAST_RUN_EPOCH:-today} --fullnames \
              | grep -vP "$AUTHORIZED_USERS_PATTERN|reboot|super" \
              | head -n -2)
```

To fix the bypassable logged-in user check, it is recommended to adjust the username regex to avoid matching substrings.

Miscellaneous Issues

This section covers any and all noteworthy findings that did not incur an exploit, but which may assist an attacker in successfully achieving malicious objectives in the future. Most of these results are vulnerable code snippets that did not provide an easy method by which to be called. Conclusively, while a vulnerability is present, an exploit may not always be possible.

MUL-04-001 WP1/2 Superfluous sudo configuration for nonexistent group *(Info)*

***Fix note:** The Mullvad team fixed the issue during the test by removing the unnecessary sudo rule and permissions.*

During the audit of the sudo configuration on the VPN relays, it was noticed that one of the sudo rules was unnecessary and for a non-existent group. This issue is not a security vulnerability, but rather a minor future hardening step. Efforts to slim down security-critical configs lead to a more concise system state and, while continuous, are worthwhile endeavors that enhance overall security.

Configuration file:
/etc/sudoers.d/super

Unnecessary sudo rule:

```
super ALL=(ALL) NOPASSWD:ALL
```

PoC:

```
root@se-got-ovpn-999:/ # cat /etc/passwd /etc/group | grep super | wc -l
0
```

It is advised that removing unnecessary sudo rules will fully mitigate this issue. Keeping the number of sudo rules to a minimum helps maintain optimal oversight of systems, particularly security-critical subsystems like sudo configuration.

MUL-04-002 WP1/2 Ansible hardening suggestions *(Info)*

Note: *After further discussion with the Mullvad team, there was sufficient justification for maintaining local copies of Ansible playbooks and roles. The local Ansible system is critical to Mullvad's deployment and configuration management processes, improving deployment time and ensuring configuration consistency. The inventory created is specific to each host.*

Therefore, while removing local copies of playbooks and roles could improve security in general, in this particular environment the benefits of ansible-local outweigh the potential risks. The presence of these files is an accepted and necessary part of their deployment strategy.

While analyzing the hosts as part of this engagement, it was discovered that a local copy of playbooks and roles exists on the WireGuard host. While this should not be considered a serious security issue, it is an unnecessary disclosure of information. A local attacker could gain valuable insight into the deployment and instrumentation process of the entire environment. Since the deployment is done remotely, the local copies should be considered unnecessary.

Affected files:
/home/mad/ansible-local#

It is recommended to remove the Ansible playbooks and roles from the local system, and to ensure they are not cached during deployment. This practice will minimize the risk of unnecessary information disclosure. By eliminating local copies of these files, the team can reduce the potential for a local attacker to gain insights into deployment and instrumentation processes. This approach will enhance the overall security posture and protect sensitive configuration data.

MUL-04-003 WP1/2: Linux hosts not deployed with minimalist userspace *(Info)*

Note: *After further discussion with the Mullvad team, it was determined that this issue did not pose a significant threat in the specific context of the target environment. The rationale behind this decision is that the attack chain required to gain an initial foothold and exploit these tools is complex and unlikely in the current security posture of the network.*

Therefore, while the presence of these tools is generally a concern, it was considered a lower-priority issue in this case. This conclusion reflects the specific risk assessment for this environment and does not negate the potential benefits of implementing a minimalist userspace in other or less secure contexts.

During the test, it was noticed that the Linux servers have been deployed without a minimalist userspace, resulting in the presence of unnecessary tools and utilities. These additional components can be leveraged by an attacker who gains an initial foothold, to escalate privileges on the host, and facilitate lateral movement within the network. The expanded userspace increases the risk of these security breaches.

PoC:

```
root@se-got-ovpn-999:/run/openvpn-server# which curl nc perl wget gcc
/usr/bin/curl
/usr/bin/nc
/usr/bin/perl
/usr/bin/wget
/usr/bin/gcc
```

It is recommended to implement a minimalist userspace on all Linux hosts by removing non-essential tools and utilities. This will enhance security by ensuring potential attackers do not find tools that can be abused for further exploitation once a primary vulnerability has been found. This approach provides an additional layer of security, and more time for detecting lateral movement or privilege escalation.

Conclusions

As noted in the *Introduction*, this Q2 penetration test and source code audit carried out by Cure53 assessed the security posture of Mullvad WireGuard and OpenVPN relay infrastructure. The assessment marked the fourth time that Cure53 has evaluated the security posture of Mullvad's infrastructure complex.

The test methodology used here adhered to a white-box approach, granting the Cure53 team full access to all items within the scope of the engagement. The suite of available materials included a repository containing Ansible playbooks and roles that define the configuration and deployment processes used by Mullvad's relay infrastructure. Additionally, Cure53 was provided with SSH-access to a dedicated staging area, where two VPN relay hosts running OpenVPN and WireGuard specific configurations were made available for testing.

The team’s overall verdict on the current security posture of the assessed items within the scope is very positive. The attention to detail and deliberate application of security concepts clearly indicate that the infrastructure team is highly knowledgeable about, and committed to sound security practices and awareness.

With that being said, Cure53 did manage to identify five security-related issues during this assessment. Two of these were classified as security vulnerabilities, while the remaining three were categorized as miscellaneous security issues. Importantly, none of the discovered security vulnerabilities could be exploited by a remote attacker. Rather, these issues are only applicable to a local attacker who has already gained an initial foothold.

When analyzing the core product, Cure53 attempted to identify any potential methods by which a user's VPN traffic anonymity or integrity could be compromised. No such issues were found, and no vulnerabilities affecting the core product were detected. Therefore, it can be concluded that the examined VPN product maintains its promised security guarantees for its users. Extensive efforts were made to leak or inject traffic into confidential parts of the network, but no progress was made in this area.

Next, the VPN's servers were thoroughly examined for vulnerabilities that an attacker could exploit in order to elevate local privileges. This area yielded the highest number of findings, specifically MUL-04-004 and MUL-04-005. Although these vulnerabilities cannot be exploited remotely, they would become significant if an attacker gained an initial foothold on the system.

It is important to note that while a few issues were discovered during this engagement, most were found during the examination of hardening features. Although these tickets describe vulnerabilities, they should not be seen as a negative reflection on the Mullvad team or its work. Mullvad's system includes a multitude of hardening features, and this is extremely positive. It also contributes to a robust security posture that mitigates many attack vectors. A small number of these hardening features are incomplete, and it is recommended that they are adjusted (e.g., MUL-04-005), or have minor bugs (e.g., MUL-04-004). It should be noted that the discovery of a few broken hardening features is actually a positive sign overall. More features can create more opportunities for bugs, but this is generally preferable to a system without any hardening features at all. Once the recommended fixes for these issues are implemented, Mullvad will have an even stronger security posture.

The Cure53 team's overall impression of the Mullvad system is that it is well-designed, well set up, and generally in exemplary condition. This reflects positively on the work, culture, practices, and capabilities of the team behind the services.

Cure53 would like to thank Joshua Björkäng from the Mullvad VPN AB team for their excellent project coordination, support and assistance, both before and during this assignment.