Security Research Labs

Blue Merle:
Reducing your Cellular Footprint

Documentation

v1.0, Oct 13, 2022

Infrared Team (in alphabetical order)

Nicholas Farnham	nicholas@
Matthias Marx	matthias@
Linus Neumann	linus@
Dominik Oepen	dominik@
Laura Pros Segura	lauraps@
Jannes Quer	jannes@
Folkert Saathoff	folkert@
	srlabs.de

- 15 pages -

Content

1 Project motivation

The *blue merle* software package enhances anonymity and reduces forensic traceability of the GL-E750 / Mudi 4G mobile Wi-Fi router ("Mudi router"). The portable device is explicitly marketed to privacy-interested users. It connects to the Internet via a user-provided Subscriber Identity Module (SIM) and can route traffic through a user-defined VPN or via Tor. Due to the device's privacy-enhanced settings the mobile network operator cannot see the content of a Mudi router user's traffic nor its end goal and is limited to seeing that the user is using a VPN- or Tor-based connection.

Upon examination of the device, we found that the Mudi router retains considerable user information that can be used to identify users through the mobile network or if the device falls into an adversary's hands. The Mudi router in default configuration is prone to tracking[1] in four ways:

a. Mobile-network tracking via International **Mobile Subscriber Identity (IMSI)**, which uniquely identifies a subscriber by its SIM card. This tracking threat can be mitigated by regularly changing SIM cards

b. Mobile-network tracking through the International **Mobile Equipment Identity (IMEI)**, which uniquely identifies the Mudi router

c. Local Wi-Fi based tracking through the **Basic Service Set Identifier (BSSID)**

d. Through forensic analysis of the **Media Access Control (MAC)** address of Wi-Fi devices that have connected to the Mudi router

The *blue merle* software package addresses the traceability of the Mudi router by adding the following features reducing the tracking risks b, c and d:

1. IMEI changer
2. BSSID randomization
3. MAC address log wiper

This whitepaper illustrates these features and elaborates on how they can mitigate user deanonymization risks.

[1] For a deeper dive into mobile network attack categories, see SRLabs' GSM map project: https://gsmmap.org

2 Privacy threat assessment

The Mudi router comes with built-in Virtual Private Network (VPN) and onion routing (Tor) capabilities, promising anonymity online. However, the anonymity promises do not extend to the Wi-Fi and cellular protocol levels. In addition, the device stores MAC addresses of connected devices, which may facilitate forensic analysis.

2.1 International Mobile Equipment Identifier (IMEI)

The IMEI is a unique 15 decimal digit code (14 digits and a check digit) assigned to each cellular device. It includes information on the origin, model, and serial number of the device. Its initial 8 digits, known as a Type Allocation Code (TAC) and assigned by the GSM Association (GSMA), denote the make and model of the phone. The remainder of the IMEI is manufacturer-defined and unique for each device. Figure 1 shows the structure of an IMEI.

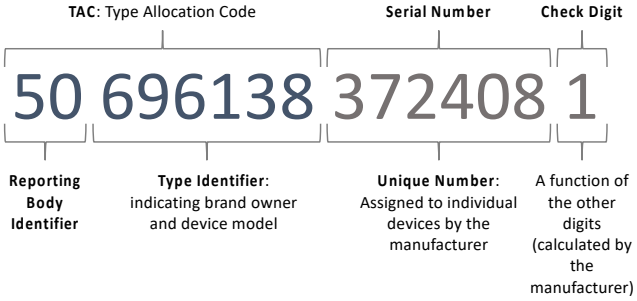


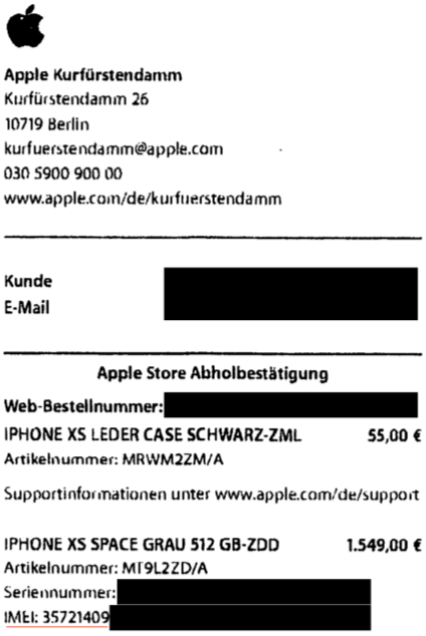
Figure 1. Structure of an IMEI

2.1.1 IMEI tracking

The IMEI is usually used in mobile networks to identify mobile devices and in some cases to prevent stolen devices from being used by not allowing them to connect to the network. As a unique identifier, the IMEI can be used to track users' locations and activity. This is actively applied as part of data retention laws, for example.[2]

It is a common misconception that changing the SIM – ideally to an anonymously sourced prepaid card – results in a completely new identity, dropping all traceability. In fact, a user changing SIM cards would only change their subscriber identity in the eyes of their mobile network. If the device remains the same, both identities are associable with the IMEI.

A device might even be traceable to a specific purchase as shown in Figure 2, allowing identification of the purchaser.



Apple Kurfürstendamm
Kurfürstendamm 26
10719 Berlin
kurfuerstendamm@apple.com
030 5900 900 00
www.apple.com/de/kurfuerstendamm

Kunde [redacted]
E-Mail

Apple Store Abholbestätigung

Web-Bestellnummer: [redacted]

IPHONE XS LEDER CASE SCHWARZ-ZML 55,00 €
Artikelnummer: MRWM2ZM/A

Supportinformationen unter www.apple.com/de/support

IPHONE XS SPACE GRAU 512 GB-ZDD 1.549,00 €
Artikelnummer: MT9L2ZD/A
Seriennummer: [redacted]
IMEI: 35721409 [redacted]

Figure 2. Invoice detailing an IMEI

2.1.2 IMEI randomization benefits

By changing the SIM, and therefore the IMSI, a user can obtain a new subscriber identity. By changing the IMEI, a user's device obtains a new identity.

[2] See https://en.wikipedia.org/wiki/Data_retention for data retention scopes and the current status in different jurisdictions

Figure 3 illustrates how IMSI and IMEI identifiers can be linked if not changed simultaneously. Only by changing the IMEI and IMSI at the same time can the user shake off unique traces associated with their old subscriber- and device-based identity.

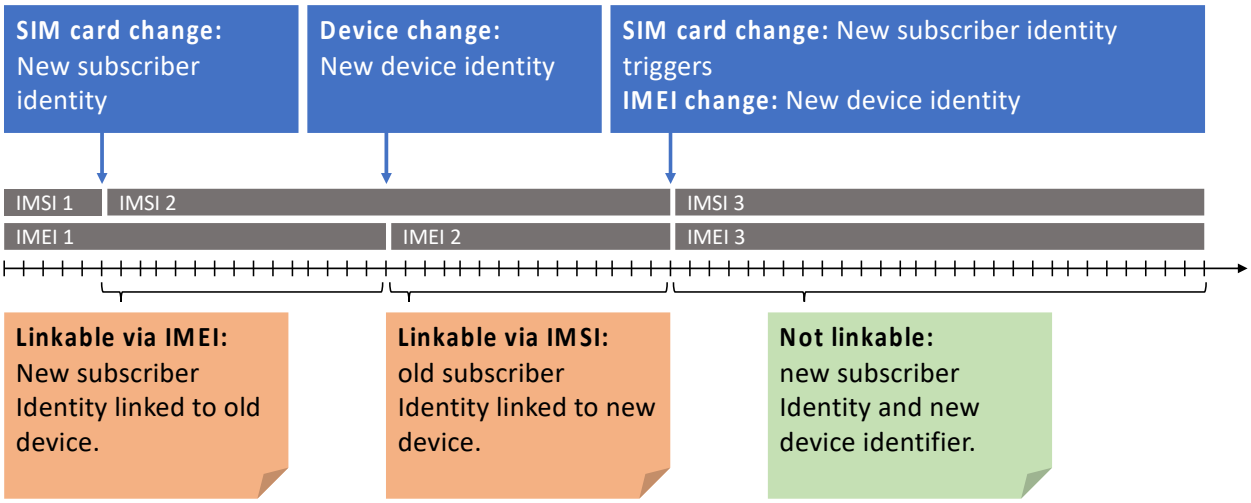


Figure 3. IMEI and IMSI change scenarios and linkability

The *blue merle* software package can be used to conduct an IMEI change upon every SIM card change to provide a new device identity. See Section 3.1 for implementation details.

2.2 Basic Service Set Identifier (BSSID)

The BSSID is an identifier code associated with a specific Wi-Fi access point. The identifier is included in all wireless packets and correlates the access point to associated clients. By convention, an access point's MAC address is used as the ID of a BSS.

2.2.1 BSSID tracking scenarios

BSSIDs are constantly transmitted by the Mudi router when it is offering a Wi-Fi connection. By passively collecting BSSIDs, device identifiers can be mapped to locations, again opening a passive location tracking attack surface to the Mudi router.

2.2.2 BSSID randomization benefits

Databases like WiGLE[3] provide geolocations of Wi-Fi hotspots. By regularly randomizing the Mudi router's BSSID, we minimize the risk of the BSSID being used to geolocate the Mudi by means of such services. The device loses a uniquely identifying artifact. Furthermore, Wi-Fi clients such as mobile phones frequently leak SSIDs – and in some cases BSSIDs – of Wi-Fi connections they have previously connected to. Changing the Mudi router's BSSID eliminates the risk posed by this source of persistent data leakage. See Section 3.2 for implementation details.

2.3 Media Access Control (MAC) address

A MAC address is a unique identifier assigned to a network interface controller (NIC) to work as a network address in communications within a network segment. This address is linked and given to the hardware of a network adapter during the manufacturing process.

[3] Wireless Geographic Logging Engine, https://wigle.net/

The MAC address consists of 48 bits, typically represented by six pairs of hexadecimal digits. On broadcast networks, such as Ethernet and Wi-Fi, the MAC address is expected to uniquely identify each node and thus allows frames to be addressed specifically to their hardware address.

Just like any device participating in a wireless network, the Mudi has a MAC address, and observes MAC addresses of other devices on the same network, each of which uniquely identifies a device.

2.3.1 MAC address log access scenarios

The Mudi router stock firmware collects and stores all MAC addresses of every connected device in the file system and makes them available via the web interface. This provides easy access to reliable proof for every device that has connected to the router. In case of loss, theft or confiscation, this data collection may prove detrimental to the users' privacy interests.

The Mudi router also offers the functionality to use other Wi-Fis as uplink. In this case, the Mudi itself leaves its MAC address on foreign systems, again leaving an uncontrollable trace. Additionally, the MAC address can be collected via passive intercept, as it is not encrypted. Therefore, the device's own unique and static MAC address is also a risk for activity and location tracking.

2.3.2 Host MAC address randomization and address log wiper benefits

By using a different MAC address on each boot, the Mudi router cannot be linked to past activities, whereabouts, and Wi-Fi connections. See Section 3.2 for implementation details.

By wiping the Mudi router's cache of stored MAC addresses at each boot, third parties with remote or physical access can no longer enumerate the devices that have connected to the Mudi router. See Section 3.3 for implementation details.

3 Implementation details

3.1 IMEI randomization

The Mudi router's baseband unit is a Quectel EP06-E/A Series LTE Cat 6 Mini PCIe module.[4]

The Mudi router's IMEI can be changed by issuing Quectel LTE series-standard AT commands. The AT command to write a new IMEI to a Quectel EP06-E/A-based device is `AT+EGMR`. Our IMEI randomization functionality is built around this command.

We implemented two approaches to IMEI generation. The first (and default) mode generates a fully random IMEI, while the second deterministic mode seeds the new value with the user's IMSI. There are some benefits and some drawbacks associated with each of these options.

Generating a random IMEI has the advantage that there is no link between the SIM card used and the generated IMEI.

Alternatively, a deterministic IMEI ensures a consistently spoofed IMEI, as the identifier is generated using the IMSI as a seed. Once the SIM is removed and the IMEI is changed, there is no proof that the SIM was ever used in a specific Mudi. This mode can be of advantage if one repeatedly uses a set of SIM cards and wants to prevent that many IMEIs become associated with each IMSI.

We recommend limiting IMEI changes to actual SIM card changes, as IMEI changes under the same IMSI do not provide any additional protection and may instead be flagged as suspicious.

Ideally, an IMEI change

1. occurs every time and only if the SIM is changed
2. results in a fully random IMEI.

3.1.1 Verification

The *blue merle* IMEI change functionality was verified using a laboratory base station set-up, confirming that the IMEI change takes place on both on the device- and network-level. See Figure 4 for the command line interface and the Wireshark trace confirming the IMEI change.

We identified a short delay between the AT-command and the actual IMEI change. This opens a risk of identity contamination as shown in Figure 3. To prevent this, the Mudi router's cellular connection must be deactivated before conducting the SIM (and IMEI) change. This issue is addressed by *blue merle* (see section 3.1).

[4] https://www.quectel.com/wp-content/uploads/pdfupload/Quectel_EP06_Series_LTE-A_Specification_V1.7.pdf

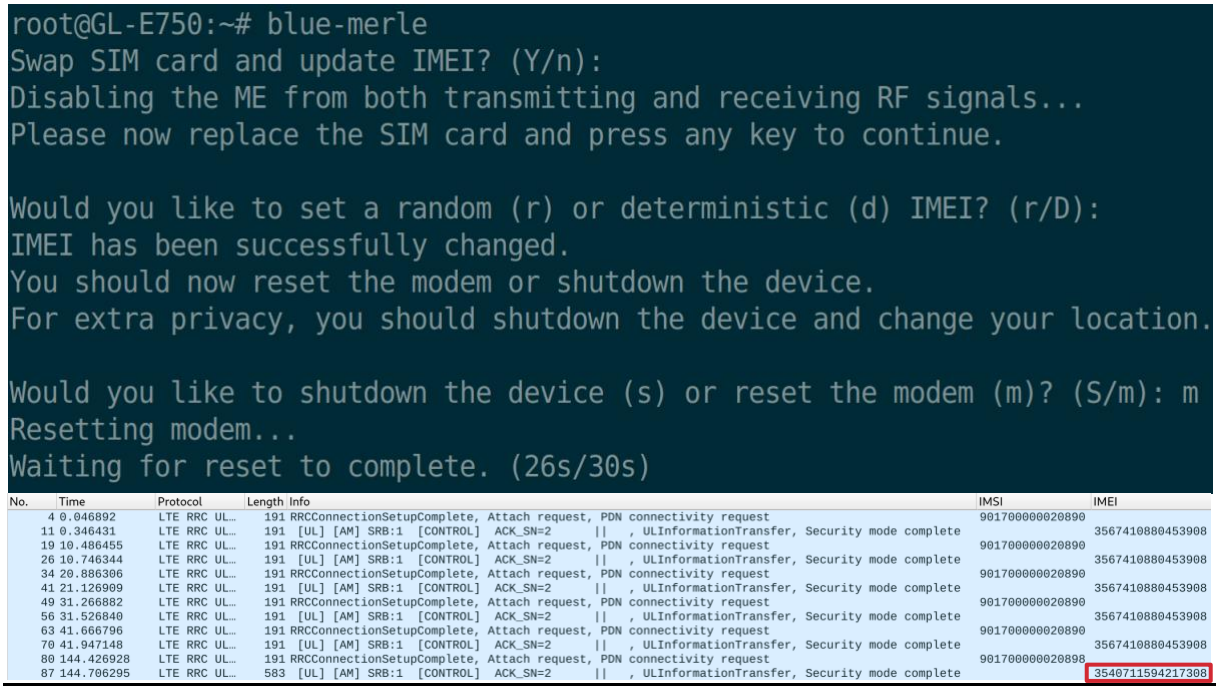


Figure 4. The IMEI change was verified in a laboratory environment.

3.1.2 Limitations

The *blue merle* package aims to eradicate remaining tracking and de-anonymization risks for users with truly anonymous SIMs. However, a mobile network could still identify that a *blue merle* Mudi is in use by linking IMEI and IMSI, or comparing frequency bands to device IMEI's technical specifications.

This alone, however, does not have any impact on the users' privacy, which is much more reliant on SIM and VPN anonymity.

3.1.2.1 Link between IMEI and IMSI in deterministic mode

In deterministic mode, the IMEI is statically derived from the IMSI. By checking the relationship between IMSI and IMEI, an observer can identify that a *blue merle* IMEI changer is in use.

3.1.2.2 Frequency bands

The IMEI is generated using TAC prefixes associated to a choice of modern and popular mobile phone models. This makes *blue merle* IMEIs less salient upon superficial analysis. As Table 1 outlines, each spoofed model supports a number of LTE frequency bands. Not all of these bands are necessarily supported by the Mudi, and vice versa.

Table 1. LTE frequency bands supported by mobile device models spoofed by *blue merle*. Frequency bands also supported by the Mudi 4G router are emphasized in bold.

Make	***Model***	**TAC**	***Frequency***
GL i.Net	Mudi GL-E750	N/A	**B1, B3, B5, B7, B8, B20, B28, B32, B38, B40, B41**
Apple	iPhone X	35674108	B2, **B3**, B4, **B5**, B12, B13, B17, B25, B26, **B41**
	iPhone 11	35290611 35397710	B2, **B3**, B4, **B5**, B12, B13, B14, B17, B25, B26, B29, B30, **B41**, B66, B71
	iPhone 11 Pro	35323210 35384110	B2, **B3**, B4, **B5**, B12, B13, B14, B17, B25, B26, B29, B30, **B41**, B66, B71
	iPhone 12	35982748	B2, **B3**, B4, **B5**, B12, B13, B17, B25, B26, B30, **B41**, B66
	iPhone 12 Pro Max	35672011	B2, **B3**, B4, **B5**, B12, B13, B14, B17, B25, B26, B29, B30, **B41**, B66
	iPhone 13 Mini	35759049	B2, **B3**, B4, **B5**, B12, B13, B14, B17, B25, B26, B29, B30, **B41**, B66, B71
	iPhone 13 Pro	35266891 35407115	B2, **B3**, B4, **B5**, B12, B13, B14, B17, B25, B26, B29, B30, **B41**, B66, B71
	iPhone 13 Pro Max	35538025	B2, **B3**, B4, **B5**, B12, B13, B14, B17, B25, B26, B29, B30, **B41**, B66, B71
Samsung	Galaxy S10 Plus	35480910	B2, **B3**, B4, **B5**, B12, B13, B17, B25, B26, **B41**, B66
	Galaxy A32 5G	35324590	B2, **B3**, B4, **B5**, B12, B13, B14, B25, B26, B29, B30, **B41**, B66, B71
	Galaxy A12	35901183	B2, **B3**, B4, **B5**, B12, B14, B29, B30, **B41**, B66
	Galaxy S20 FE LTE	35139729, 35479164	B2, **B3**, **B5**, B12, B13, B17, B26, **B41**, B66

A fingerprinting risk emerges when *blue merle* generates an IMEI with a TAC of a phone model *not* supporting LTE frequency bands the Mudi router supports, namely B1, B3, B5, B7, B8, B20, B28, B32, B38, B40 and B41. When a *blue merle* Mudi uses a frequency band that does not match the TAC's specification, an observer can deduce that the IMEI is spoofed.

This risk can be mitigated by disabling frequencies that do not match the spoofed model. However, this might impact service quality. As the fingerprinting risk alone has no immediate impact on user privacy, this feature is not included in the initial *blue merle* release and left as an exercise for the curious reader and contributor.[5]

[5] See https://github.com/srlabs/blue-merle/issues for inspiration on how to contribute

3.2 BSSID and MAC randomization

The Mudi router BSSID is set by the process `hostapd` using the function `mac80211_prepare_vif()` in `/rom/lib/netifd/wireless/mac80211.sh`.
The resulting BSSID is stored in `/etc/config/wireless`.

The *blue merle* BSSID randomization function generates a valid unicast address value and overrides the current MAC values set for the `wlan0` and `wlan1` interfaces. This is done by issuing the OpenWrt[6] command `uci set` targeting the `macaddr` fields of `wireless.@wifi-iface[0]` and `wireless.@wifi-iface[1]`. The Mudi router's Wi-Fi is then reset to implement the changes.

BSSID randomization is performed on boot, ensuring that a new BSSID is generated each time the device is started. In the same process, the Mudi router's MAC address is also randomized, eliminating the tracking risks outlined in section 2.2.

3.3 MAC address log wiping

MAC addresses of devices that connected to the Mudis Wi-Fi connection are stored in `/tmp/tertf[_bak]` and `/etc/tertf[_bak]`. The *blue merle* MAC address log wiper first symbolically links the `gl_tertf` file responsible for the `gltertf` process, which reads and logs MAC addresses. It then kills the `gltertf` process if active, checks if either file contains any data and uses `shred` to delete any data if found.

The MAC address log wiper is run on boot, ensuring that the Mudi device's initial MAC log read/write functionality is disrupted each time the device is started.

[6] https://openwrt.org/start

4 Usage Instructions

4.1 Installation

A few simple steps allow users install *blue merle* on their Mudi router, as shown in Figure 5.

Note: The *blue merle* project was developed and tested on firmware version 3.215 and might not be compatible with future firmware updates.

1. Update the Mudi router to firmware version 3.215 if running an earlier firmware version
2. Copy the OPKG file to the Mudi router via `scp`
3. Install the package: `opkg install blue-merle*.ipk`

The scripts for BSSID randomization and MAC address log wiping are run automatically after installation and during each boot. MAC address logs are also wiped on shutdown.

```
[redacted]~/tmp/openwrt$ scp bin/packages/mips_24kc/base/blue-merle_1.0.0-0_mips_24kc.ipk
mudi1:./
root@192.168.203.194's password:
blue-merle_1.0.0-0_mips_24kc.ipk                          100% 6231   501.7KB/s   00:00
[redacted]:~/tmp/openwrt$ ssh mudi1
root@192.168.203.194's password:


BusyBox v1.30.1 () built-in shell (ash)

  _______                     ________        __
 |       |.-----.-----.-----.|  |  |  |.----.|  |_
 |   -   ||  _  |  -__|     ||  |  |  ||   _||   _|
 |_______||   __|_____|__|__||________||__|  |____|
          |__| W I R E L E S S   F R E E D O M
 -----------------------------------------------------
 OpenWrt 19.07.8, r11364-ef56c85848
 -----------------------------------------------------
root@GL-E750:~# opkg install blue-merle_1.0.0-0_mips_24kc.ipk
Installing blue-merle (1.0.0-0) to root...
Device is supported. Installing blue-merle...
Configuring blue-merle.
The /tmp/ directory does not exist. This should be fine...
The /etc/ directory exists.
killall: gltertf: no process killed
No file found within /tmp/tertf. No shredding to be done there.
No file found within /etc/tertf. No shredding to be done there.
Looks like /tmp/ is clean!
Looks like /etc/ is clean!
root@GL-E750:~#
```

Figure 5. Installation of *blue merle* via ssh

4.2 IMEI change

The IMEI randomization can be initiated (a) using the toggle button, or (b) manually by running the `blue-merle` command via ssh. Both approaches implement the following steps.

1. Disable the baseband to prevent transmission of RF signals
2. Prompt the user to replace the SIM card
3. Once the SIM card has been changed, initialize the new SIM card
4. Generate a new IMEI randomly or deterministically
5. Initiate a shutdown or hard reset of the baseband

For debugging purposes, we also detail the completely manual approach below.

4.2.1 Toggle button

To initiate this sequence, move the toggle button on the left side of your Mudi to the top position and follow the display instructions (see Figure 6). The device will shutdown after completing the IMEI change.

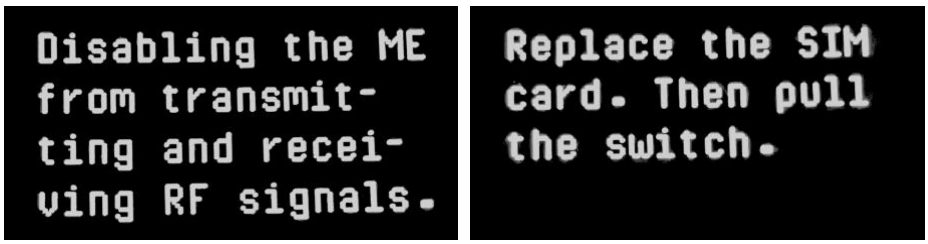


Figure 6. Display instructions during SIM/IMEI change

4.2.2 blue-merle command via ssh

Instead of issuing the IMEI change via the toggle button,

1. Connect to the device via ssh
2. Run the `blue-merle` command
3. Follow on-screen instructions as shown in Figure 4.

Using the blue-merle command line version, you can choose to set an IMEI either randomly or deterministically. As an alternative to shutting down the Mudi, you can choose to reset the modem only.

4.2.3 Debugging

Note: The manual mode detailed below is recommended for development and debugging purposes only.

1. Establish an SSH session
2. Turn off the device radio through the following command:
   `gl_modem AT AT+CFUN=4`
3. Replace SIM card

4. Run AT commands `gl_modem AT AT+CFUN=0 && gl_modem AT AT+CFUN=4` to initialize the new SIM and disable signal transmission to prevent IMEI leakage

5. Randomly change IMEI:
   `python3 /lib/blue-merle/imei_generate.py -r`
   Deterministically change IMEI based on the IMSI:
   `python3 /lib/blue-merle/imei_generate.py -d`

   The new IMEI value can be displayed through the following command:
   `gl_modem AT AT+GSN`

6. Power down the device: `echo {\"poweroff\": \"1\"} >/tmp/mcu_message && sleep 0.5 && killall -17 e750-mcu`

7. [Recommended] Change physical location

8. Check that the IMEI is still the changed one by issuing the command:
   `gl_modem AT AT+GSN`

5 Limitations & Disclaimer

While *blue merle* improves the anonymity and reduces the traceability of your Mudi, the actual level of anonymity depends on other factors. *Blue merle* can't provide any protection if you use it wrong.

Firstly, *blue merle* permanently removes all traces of the router's old identity. However, any resulting privacy benefits depend on the degree of anonymity of the used SIM card and the used IP anonymization technology (VPN / Tor)[7]. In some countries, anonymous SIM cards may be difficult to acquire.

Secondly, *blue merle* can lead to issues when changing the router's identifier to an IMEI that is also active within the same network or to the IMEI of a stolen device. Depending on your jurisdiction, changing your IMEI might violate local regulation or law.

Finally, *blue merle* is a research project and further or unanticipated modifications to the Mudi router can significantly impact its effectiveness.

Use *blue merle* at your own risk and only if you understand what is does and does not do for you.

6 Acknowledgements

The Mudi router shares its name with the well-known Hungarian dog breed typically used to guard and herd flocks of livestock. Mudis are agile, fast-learners, and extremely friendly.

"Blue merle" is one of the five coat colors recognized for the Mudi dog breed by the *Federation Cynologique Internationale* and is characterized by its mottled or patched appearance. The black splashes on the blueish-gray coat of the blue merle Mudi inspired the name of this project because of its obscuring appearance and camouflaging symbolism. Figure 7 shows a particularly beautiful specimen of the blue merle Mudi.

Figure 7. Blue merle Mudi by Taru T Torpström, CC BY-SA 3.0

[7] Please note that an evaluation of the advantages and disadvantages of VPN providers and Onion Routing are beyond the scope of this case study. In general, Tor is preferable over commercial VPNs because no single entity needs to be trusted. We trust that blue merle users make an informed choice based on their individual threat landscape.

7 Contribution

The contents of the open-source project *blue merle* can be found on Github and are licensed under BSD-3-Clause license.[8]

We look forward to contributions from the community. In particular, we welcome pull requests to support for other devices using the Quectel EP06-E/A baseband, or other basebands that allow changing the IMEI.

[8] https://github.com/srlabs/blue-merle