**Cyber Security Analysis of clients’ vehicle fleet logs, including in-depth analysis of chosen vehicle CAN Trace to uncover possible cyber-attacks, vulnerabilities, reoccurring suspicious patters and threats that require immediate attention.**

**Introduction**

It is crucial to constantly monitor one’s network for cyber security threats and anomalies, frequently performing investigations thus enriching the knowledge base enveloping the inner and most valuable, but at the same time vulnerable resources of the company, to enable a clear and up to date vintage point overlooking the current state of events as it unfolds at a constantly increasing pace throughout time. Attackers constantly evolve and do not hesitate to use Immediate chance opportunities and zero-day vulnerabilities, and the recent events in the cyber world, including the breach to NVIDIA and SAMSUNG servers must serve as an immediate red flag to urgently act in a thrive to achieve the most possibly secure cyber ecosystem possible through precise monitoring and constant research, uncovering potentially hazardous behavior and patterns long before it grows and snowballs into a humongous threat.

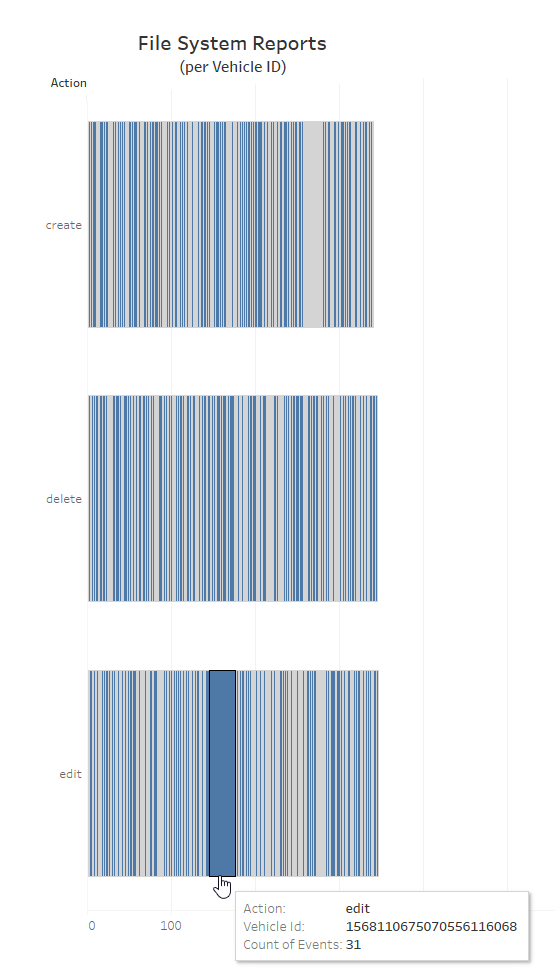
Consequently, the purpose of this analysis project is to translate our clients logs accumulating information about thousands of vehicles, into clear and plain results, highlighting the potentially hazardous and anomalous parts inside those logs, thus allowing the company to take swift action both in the sphere of fixing the damage done, and preventing further attacks. In other words, it is aimed at unveiling immediate dangers as well as potential dangers in the future, thus allowing the company to notice and understand potential threats which may develop beneath the surface, hidden by the veil of one dimensional thinking caused by seemingly beneficial situations and false perception of statistical events, (‘this will not happen to me..’) as dictated by our primordial instincts forcing us to focus on the good parts and procrastinating the mere thought about the bad occasions - which is a first standing reason for missing such dangers.

**Methods**

Conclusively to the goals of the paper, to provide clear and unbiased analysis the research will commence without previous knowledge about the company or its clients and products, thus forcing an open-minded approach to the data presented via the logs and later CAN Trace. The dangers and attacks are buried deep within a plain mix of text and numbers. And this is where the analyst must focus his attention.

While analyzing the data I used such tools as Kibana for getting a profound glimpse of the data and understand it’s structure, Python to clean the data and prepare it for the next stages, Excel to extrapolate conclusions and understand the data deeply, and finally Tableau to produce visualizations and graphs.

**Results**

First, I decided to create a ‘File System Reports’ per ‘vehicle ID’ visualisation, as it will show if there is some vehicle with anomalous amout of file system events, which might indicate a past or an ongoing attack.

We see that indeed one vehicle has an anomalous amount of events, a whole magnitude above other vehicles. The events are EDIT file system reports, which might indicate this vehicle’s file system was tampered with.

A further investigation is required on this vehicle.

Table

Description automatically generatedNext, I decided to make ‘Firewall Event Reports’ per ‘Source IP’ when the direction is ‘in’, and all that broken by vehicle ID to see the amount of foreign connections by each public IP. Having some foreign public IP trying to connect or sending traffic to a lot of cars might be a sign of a potential attack.

Full Size:

<https://prnt.sc/YD6t-6M_qiV8>

We see that indeed in contrast to the vast majority of foreign IPs that connect to only one unique vehicle only once, there are 4 IPs that divert form this pattern.

104.168.155.129 connects (causes Firewall Event Report) to a lot of vehicles a lot of times.

52.16.223.194 connects to a lot of vehicles, few times to each.

63.32.155.212 also connects to a lot of vehicles, few times to each.

51.178.161.32 connects to one vehicle 10 times.

Further investigation is required on each of those IPs, but for now we can deduct that 52.16.223.194 and 63.32.155.212 share the same anomalous pattern, while 104.168.155.129 and 51.178.161.32 show unique anomalois behaviour.

This might indicate that the IPs that share the same anomalous pattern are related to eachother, while those which show unique behaviour are standalone entities.

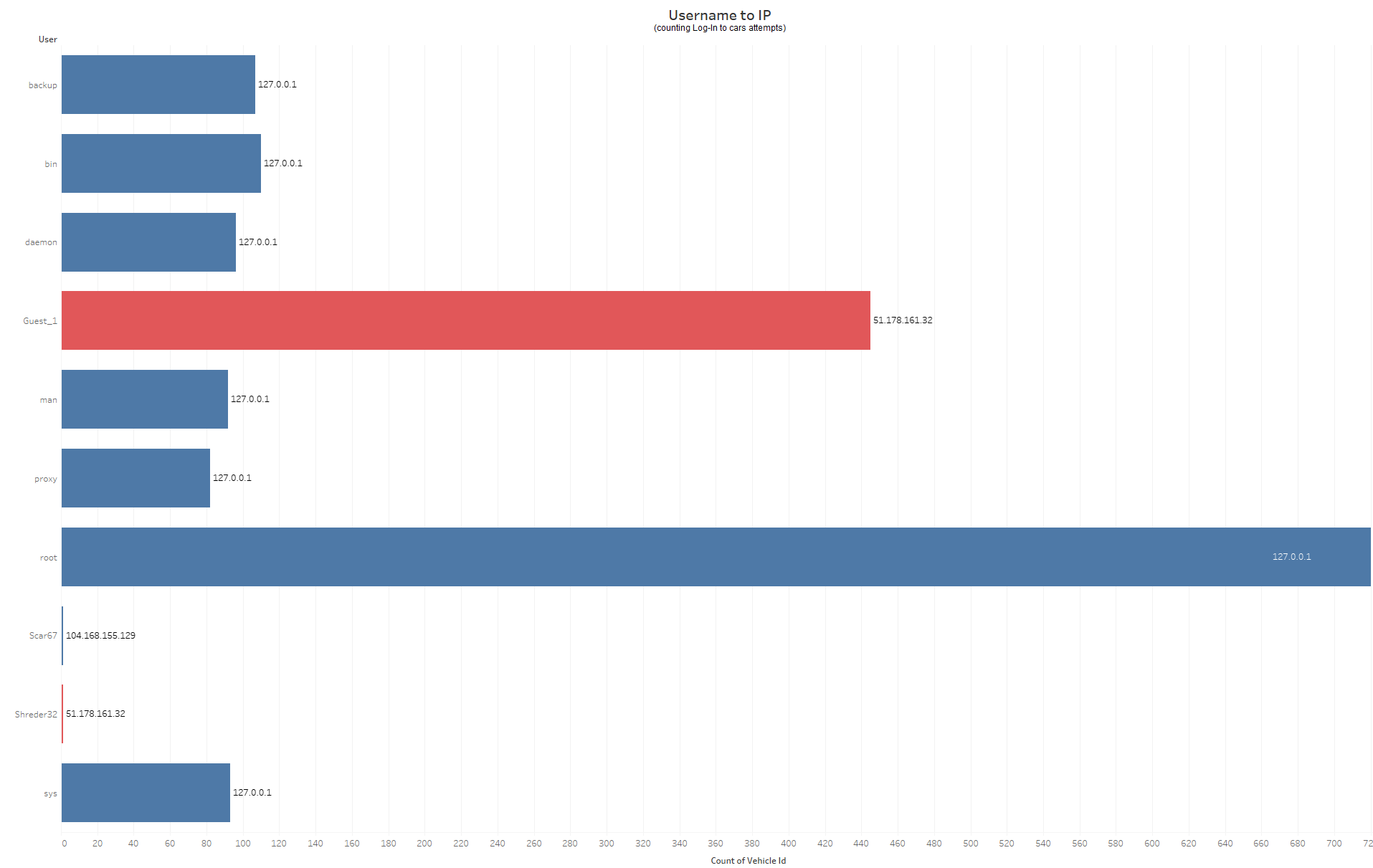
Chart

Description automatically generatedContinuing the research, I decided to make a per IP log-in graph, showing both successful and failed attempts. Seeing a certain IP attempting a lot of log-ins to different cars might indicate a brute force attack, where the attacker tries manufacturer default and common passwords to log-in into the cars.

Indeed, we see that while the vast majority have one log-in per one car, 51.178.161.32 (previously suspected) performed 441 failed and 5 successful log-ins to cars. Such humongous number clearly shows his intents are malicious, and demands a further analysis of the cars he did manage to log into. (127.0.0.1 is not anomalous as it’s a direct physical connection).

A message should be distributed to all company clients, advising them to change their default or weak and known password to a strong one.

Finally, I decided to create a users graph, to connect the IPs to Users (counting log-in to cars attempts to get depth of activity) and possibly see another interesting insights. Spotting unregular users, or different users having the same IP might show that some users were hacked from the IP they’re now sharing.



Indeed, we can see that the IP we initially suspected and saw making a lot of log-in attempts (51.178.161.32) is behind a Guest user, strengthening the idea it tries to log-in using default manufacturer passwords. Interestingly, a log-in under Shreder32 user is also performed from this IP, strengthening the idea the attacker also tries to find the legitimate user profile password, trying easy and popular combinations.

(It is noteworthy that it is technically possible that Shreder32 is the account of the attacker he inserted into a car, but as it’s a widespread brute force attack such targeted approach is unlikely).

Additionally, from the IP we also initially suspected (104.168.155.129) a log in under Scar67 user is registred. In contrast to 51.178.161.32, 104.168.155.129 attacker do not appear in other log in attempts, which shows either he changes his IP contantly, or he performed a targeted attack especially on one vehicle.

(it is noteworthy that Scar67 might be the legitimate owner user, but since it is a targeted attack it may be the attackers specially inserted user as well. Further investigation will tell us more.)

It is possible to continue the visualization and overall investigation indefinitely, but as I already have a lot of starting points to continue the investigation more thoroughly, I decided to move on to the exact cases investigation including CAN trace, focusing on the targeted and therefore likely elaborate attack.

**The main vectors of investigation will be:**

1. The elaborate (targeted) attack on the 1568110675070556116068 Vehicle by 104.168155.129 (case Scar67).

* LOGS Analysis.
* CAN Analysis.

1. The scattered (brute force) attack on numerous vehicles by 51.178.161.32. Focusing on cases where the attacker managed to log in into the vehicles. CAN traces are unavailable for those vehicles, so I’ll try to develop the best picture possible from logs.

* LOGS of the scattered attack overall.
* LOGS of 1943288260238536165725 as Shreder32 (1 log-in).
* LOGS of 1862204050972187937795 as Guest\_1 (4 log-ins)

1. The actions of 63.32.155.212 and 52.16.223.194, which arise suspicion.

* Are they connected?
* Are they malicious?

The goal of the in-depth investigation is to provide an answer for the question of what happened during the attack, what vulnerability made it possible, and how it can be prevented.

There are a lot of different possibilities and plausible explanations in the cyber security world. Thus, when appropriate, an alternate version to the events will be presented, to allow a view on the events from another side, encouraging further discussion and research to determine which of the versions is more likely to be correct.

**Targeted Attack: case 1568110675070556116068**

**Attacker:** 104.168.155.129

**Target:** Vehicle1568110675070556116068

**Result:** The attacker managed to log-in, alter and add new files on the target vehicle.

**Current Situation:** The attacker maintains a backdoor inside the vehicle.

The research of the attack on this car will commence from two vectors:

1. Information gathered from the logs.
2. Information gathered from the CAN trace.

Those are all the logs stored for the car in question, ordered by date & time:A picture containing chart

Description automatically generated

Full size: logs of 156…068

<https://prnt.sc/6uvR-TBTC-3c>

In green, are parts that to my understanding are legitimate and performed by the owner of the car (like the log in in line 6) or the system background processes (like the SSL certificate update in line 9, or automatic backups in lines 3 and 43).

Local log-ins (127.0.0.1) are also less suspicious, as it’s most likely the attacker operates remotely and don’t have a physical access to the vehicle. However, since this is a targeted attack, such access is also a potential possibility.

In yellow are lines that are possibly suspicious, and related to other attacks. For example in lines 4,5 and 38 the car is possibly being attacked by the two IPs suspicious from the foreign IPs graph. (Further research showed those are most likely legitimate server services, more information in the third part of the report, focusing on those IPs).

Also, as the logs for this car are beginning from a deletion of a file, it is possible they do not cover the whole picture of the events, and maybe an attacker infiltrated the car before they started recording\deleted those logs, so the attacker was the one to alter the certificates file in line 9 to enable further safe connection with the car at a later stage. (We see the attacker used ports 22 (SSH) 80(HTTP) and 443(HTTPS) to send traffic to the car, so certificate would be needed for SSH and HTTPS as those are encrypted protocols).

In red are lines tied directly to the attack from IP 104.168.155.129. According to the most plausible explanation of the attack, at 20:25 January 5th 2022 the attacker started trying to establish communication with the car, using various protocols and methods. (he may have established it at an earlier stage, or the whole attack might be automatic and launched from an infected website – those possibilities will be explored at a later stage).

After establishing communication, he logged in with the user Scar67, (which may be either the legitimate user profile the attacker found/guessed the password to, or a the attacker’s user he inserted into the car computer.) and began altering various, mostly security related files, culminating in the creation on his own file ‘cryp\_01012022.exe’. The attacker continued altering files, making himself comfortable inside the car computer.

**Alternate versions:**

Another version, is that the attack was automatic and used some vulnerability to run code on the system after clicking some malicious link (for further information refer to the research of the suspicious IPs connection with which are seen in lines 1416 and 895), or was pre-prepared physically by the attacker to connect to the car later

In that case at 20:25 January 5th 2022 the user clicked some malicious link, as seen by the HTTP and HTTPS protocols involved, but simultaneously to being redirected to the site he wanted, an SSH communication was opened, through which the attacker logged in. Such approach would require either the exploit of a severe vulnerability, or previous access to the machine to prepare the communication (and the SSH certificates file indeed was altered beforehand in line 9).

Thus, since a local log-in (line 6) was registered in the same day as the certificate file edit, it is possible the attacker knows the victim personally, which is common in directed attacks, so he accessed his car physically, set up the certificate file to allow connection, afterwards he (through a malicious link or direct connection) connected the car computer, and began altering more files and creating his own.

**Can Trace:**

A picture containing text, measuring stick, device

Description automatically generatedFirst and foremost, I’ve looked for anomalous readings in the CAN protocol. I’ve noticed that there are signs or possible buffer overflow in the steering angle variable, having multiple FF (255) values, as well as the Direction value that should be either 0 or 1, got anomalous values like 2,3, and up to 8. Curiously, BUS 2 are responsible for both those values, so an attack on it might have caused such anomalous readings.

I’ve started the research by translation of all the relevant parts of the CAN trace, and its’ visualization to understand where to start looking for anomalies.

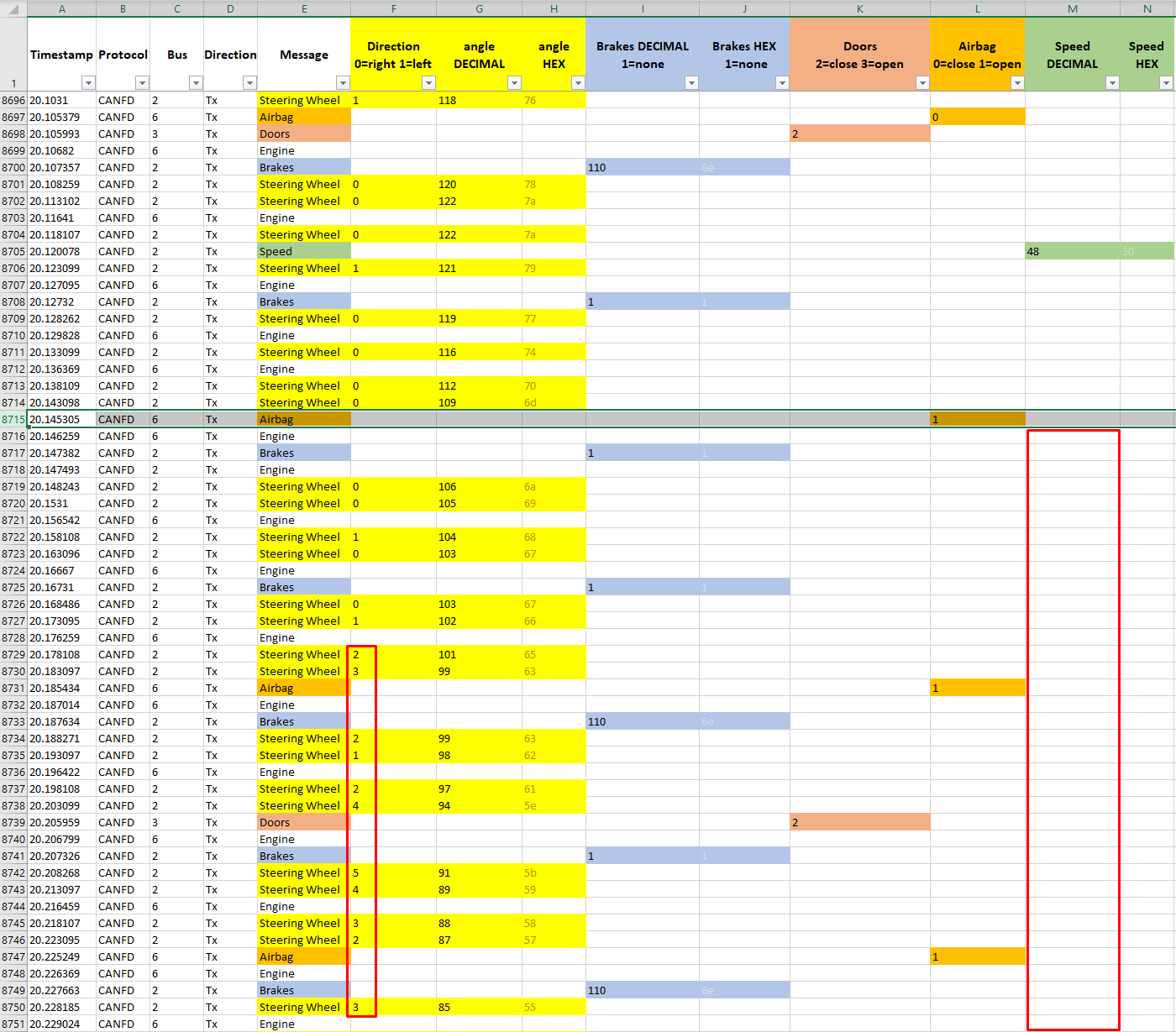
Choosing what fields to visualize and the way to do so, I decided comparing all the reports from all the subsystem categories vs time, to see when something strange had started to happen.

Such graph clearly showed that there was a sudden stop in speed telemetry right when the airbag (shown as the red part) was reported as activated.

This response might indicate something happened to the nodes of the car at this timestamp, especially in the car wasn’t involved in some crash.

(Theoretically, the crash could be caused by the attack. We should keep this thought to the CAN trace analysis).

Assuming the car didn’t went through an accident, a bit flipping its value without physical reason might serve as another prove to buffer overflow attack, if addition to the FF one we already found. More data should be seen in the CAN Trace at the moment the airbag was reported opened.

****without further adieu, I moved on to the CAN Trace, to the point the airbag was reported as activated.

When the Airbag is reported as opened (line 8715), a few interesting things happen simultaneously:

* The Direction value begins to show incorrect information. It must be either 0 or 1, but gets other numerical values.
* The speed data just disappear from appearing for a while, returning only from line 9814, but remaining 0. (the anomaly we saw in the visualization).
* The breaks seem to repeatedly jump from 110 (high pressure) to 1 (no pressure at all).

All those are clues for possible buffer overflow, that happened 20.14 seconds into the CAN Trace.

Continuing the CAN Trace investigation, I found more interesting insights:

The brakes data seem to not have been recorded/transmitted at all from line 55 and up to line 7270 (remained 1 – no break input) despite the fact the car accelerated and decelerated during this time, including of up to speeds of 254 units. Therefore, either the brakes or the speed data is faulty.

Graphical user interface

Description automatically generated with medium confidenceExample of correct functionality: (line 8562) – decelerated speed is registered in the brakes.

Graphical user interface, application, table, Excel

Description automatically generatedExample of corrupted functionality: (lines 55-7270) decelerated speed is not registered.

**Case 1568110675070556116068 Final Conclusions:**

* According to all the discovered information, I believe the vehicle was subjected to buffer overflow attack that took place 20.14 seconds in Can trace, and presumably around 20:35 January 5th, 2022, according to the logs. (Assuming this overflow allowed the attacker to connect. It is possible he used this overflow at a later stage, for example to achieve higher system privileges).
* After the initial connection was established, the attacker altered key system files, (mostly security like groups or certificates) and created his own files, presumably to maintain control over the vehicle.
* At the time of logs and CAN trace end, the attacker still maintains a footstep inside the vehicle system.
* It is possible the attacker used some vulnerability in the breaks subsystem (BUS 2, just like angle and direction) that allowed him to perform the attack, since it was building up for over 7,000 lines.
* Overall, Additional investigation of this case is advised. Focusing on helping the client to get rid of the attacker and assessing possible damage that was done, and developing protection mechanisms to avoid such attacks in the future. (Possible breaks system vulnerability).

**Alternative Version:**

Assuming the car did suffer an incident (if the airbag was actually deployed, and not just registered as deployed), there are proves the brake system of the car wasn’t functioning properly for over 7,000 lines (around 20 seconds), which, combined with the fact it was a targeted attack, might serve as a reason to launch a criminal investigation.

**Widespread Attack: Case 1943288260238536165725 & 1862204050972187937795**

**Attacker:** 51.178.161.32

**Target:** 446 Vehicles. For 2 vehicles the attack succeeded.

**Result:** The attacker managed to log-in to 2 vehicles and alter files on the targeted vehicles.

* 1943288260238536165725 – logged in as shreder32 (1 time).
* 1862204050972187937795 – logged in as Guest\_1 (4 times).

**Current Situation:** The attacker maintains a footstep inside the two vehicles.

The research of the attack on this car will focus on the logs of the two vehicles that were logged-in to. CAN traces are unavailable for those vehicles.

Graphical user interface, application, table, Excel

Description automatically generatedThe Attack: Example of successful log-ins amidst hundreds of failed ones:

Graphical user interface, application, table, Excel

Description automatically generated

It is clear the attacker used a widespread attack approach: trying for a lot of vehicles (possibly with default or easy and known passwords) and hoping some of them would work. Now let’s dive into the two cases where he succeeded to connect.

Full size: logs of 194…725

<https://prnt.sc/tdejQ_q8xirQ>

**Graphical user interface, Excel

Description automatically generatedOrdered logs of 1943288260238536165725 – attacker logged in as shreder32 once.**

The green lines 1721, 861 and 1106 are assumed to be legitimate, as the IPs involved in them do not repeat in other cases. Lines 2224 and 2254 represent direct physical connection, so unless an attacker had a physical access to this vehicle (highly unlikely due to the widespread nature of the attack) they’re legitimate as well.

It is noteworthy however, that those lines might be considered suspicions if assuming the attacker changed his IP and connected with root privileges. This however would require the attacker to infiltrate the machine beforehand, and it doesn’t make sense to perform all the SSH connection attempts afterwards had he had such abilities.

Lines 2646 and 2647 is suspicious, as they’re a CREATE file system event, with the relevant information (file path) missing. It may indicate the log was altered to cover tracks, or was created incorrectly, maliciously or not.

Line 2891 is suspicious because while a security file is deleted, it happens before the attacker logged in. therefore, it might be a routine legitimate user or operating system operation.

The red chunk between line 3001 and 2190 (visually. the line ID isn’t sorted by date & time). Is where the attack took place. There are multiple connection attempts via SSH from 51.178.161.32 to the vehicle. In contrast to other cases where he failed to connect (except one case with car 1862204050972187937795 that will be discussed later) here he managed to log in as user Shreder32. Similarly to the Scar67 case, it is unknown if this is the naming of the legitimate user profile the attacker got the password of, or a user the attacker managed to insert into the system. As it’s part of a widespread brute force of phishing attack, I assume it is likely the legitimate user account this time, and the attacker obtained the password.

After the connection, the attacker seem to do nothing (at least his actions weren’t spotted in the log) up to January 11th, where he deletes a security file and the log ends. It is possible however this deletion was a part of normal operating system functionality like on line 2891, and we must be aware of this option.

Graphical user interface, application

Description automatically generated with medium confidence**Ordered logs of 1862204050972187937795 – attacker logged in as Guest\_1 four times.**

Full size: logs of 186...795

<https://prnt.sc/bU0CzzGdHzOU>

Those logs show a similar picture to the previous case, here there’s also some unknown (appearing only once in the whole log) IP connecting to the car in line 2066, followed by physical connections. In my opinion those are legitimate connections, but theoretically they may as well be malicious connections by the attacker, especially if he had physical access to the car, or established a way to log in to the car beforehand. 1202, 481 and 488 are also local based connections, therefore they are most likely legitimate.

In lines 838, 1461 and 1460 we see the same IPs from Scar67 case, hence they are possibly suspicious. Further research about those IPs will be conducted at a later stage. (Further research showed those are most likely legitimate server services, more information in the third part of the report, focusing on those IPs).

In line 2560 we see a CREATE file system event, without any specification of file path. Similarly to the previous case, it is very suspicios.

In line 1264, January 9th the attack started, including over a dozen tries to connect to the car, presumably using a password brute force guessing, based on manufacturer default settings and commonly used passwords. After 14 attempts the attacker manages to connect to the car, and interestingly does so 4 times in a row within less than a second. It may indicate the log in attempts were handled by a specially written code, that just bombarded the car with possible passwords, after the connection, the attacker created his own file, where he possibly stored means for further attack on the vehicle.

The root connection on line 2365 might be legitimate as it’s an IP not seen in any other part of the log, but most likely it’s a connection mean the attacker established for himself during the attack, so later he’d be able to connect to the vehicle from another IP and with root privileges.

Following on this thought, it is possible that those strange root log ins from IPs not seen anywhere else in the log might be the result of such attacks at an earlier stage, not covered by the currently available logs.

**The actions of 63.32.155.212 and 52.16.223.194**

Finally, I wanted to check the actions of those IPs that proved to be suspicious, as appearing in two of the three attacks, and being overall widely spread across the logs.

Table

Description automatically generated**52.16.223.194 in ingoing traffic to the cars:**

Table

Description automatically generated**52.16.223.194 in outgoing traffic from the cars:**

A picture containing text, window

Description automatically generated**63.32.155.212 in ingoing traffic to the cars:**

**A screenshot of a computer

Description automatically generated with low confidence63.32.155.212 in outgoing traffic from the cars:**

**Actions of 63.32.155.212 and 52.16.223.194 Conclusion**

From those logs we can see that 52.16.223.194is always on port 443, which means HTTPS. It is receiving communications from many cars and transmitting back to many cars, without repetitions.

63.32.155.212 also receives and sends a lot of traffic to a lot of different cars, without repetitions. It’s working on port 80, which means HTTP.

This data make me think that those IPs belong to some service that a lot of people are accessing, for example google maps, as it receives and sends back communications.

What is interesting is the involvement of this IP in 2 out of the 3 attacks, which forces us to explore the possibility that either the whole site those people are accessing is a malicious website (less likely), or, one site is legitimate (for example the HTTPS one), but it was subjected to a separate attack that forces it to redirect (after clicking something for example) the users to the malicious site (HTTP), that initiates the attack. In that case, all the attacks involving those IPs might be automatic, with the attacker returning to his tools to get the results after some time – that would explain the delay sometimes present before accessing the car systems, and actual actions being made.

It is advisable for a further investigation to be launched, aimed at determining the true nature of those IPs, and the reasons they were involved in 2 of the 3 attacks on the vehicles.

**Analysis Wide Conclusions:**

* The company seems to manage most of its’ vehicle fleet safe from cyber threats, however no system is safe, and this research proves just that.
* The company must continue investing in cyber security, both hardening its car electronics based on the CAN Trace research, and other independent research.
* The Company must aid the affected clients, in order to make sure the attackers are purged from their vehicles and do not pose any additional threat.
* The Company must conduct more cyber security researchers like this one, to constantly be updated in regard to the latest happenings and threats inside its systems.

Daniel Levin.