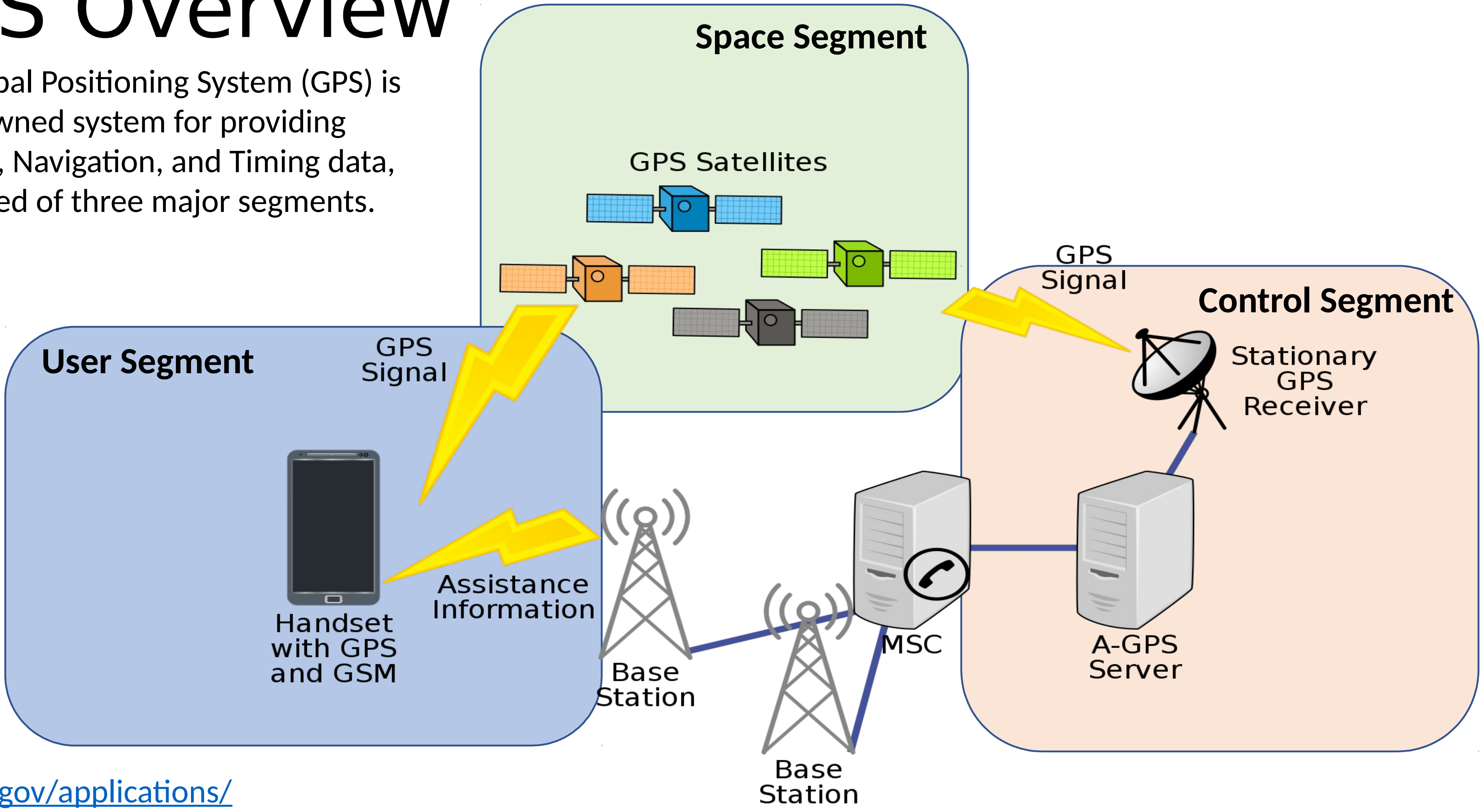
Agenda

- GPS Overview
- STRIDE Threat Model
  - Spoofing
  - Tampering
  - Repudiation
  - Information Disclosure
  - Denial Of Service
  - Escalation of Privilege

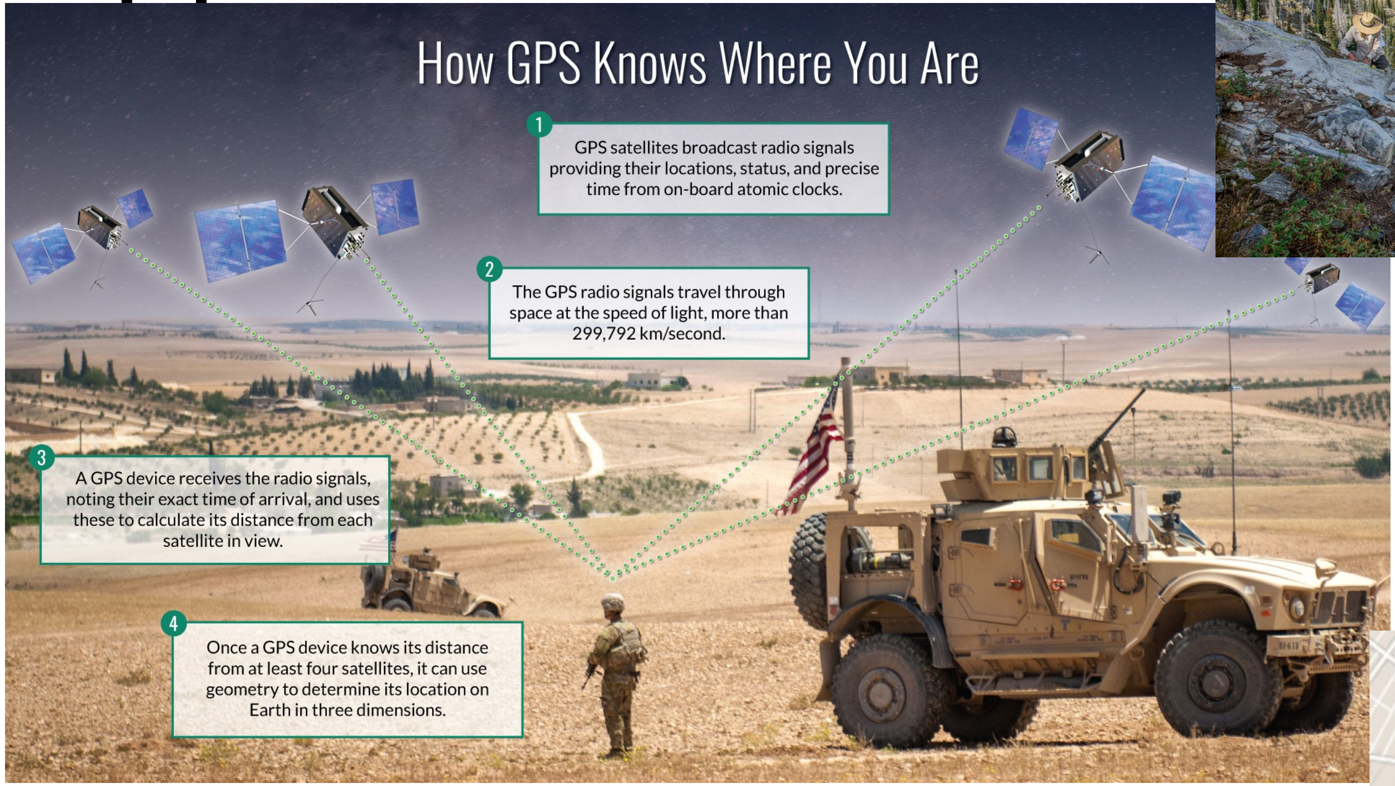
GPS Overview

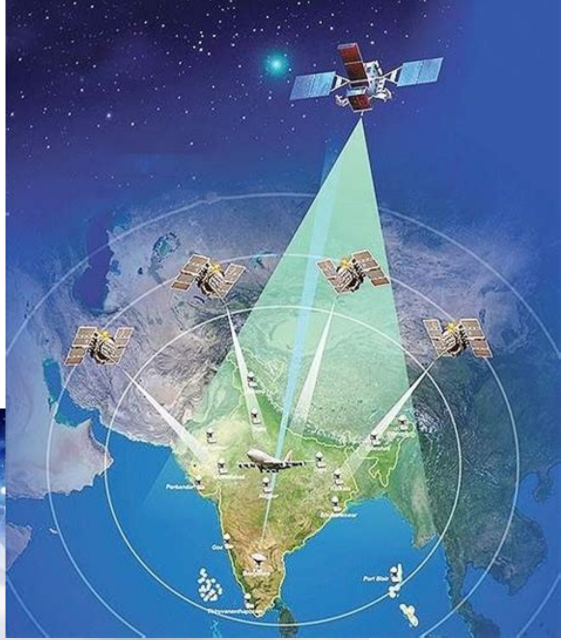
The Global Positioning System (GPS) is a U.S. owned system for providing Position, Navigation, and Timing data, composed of three major segments.

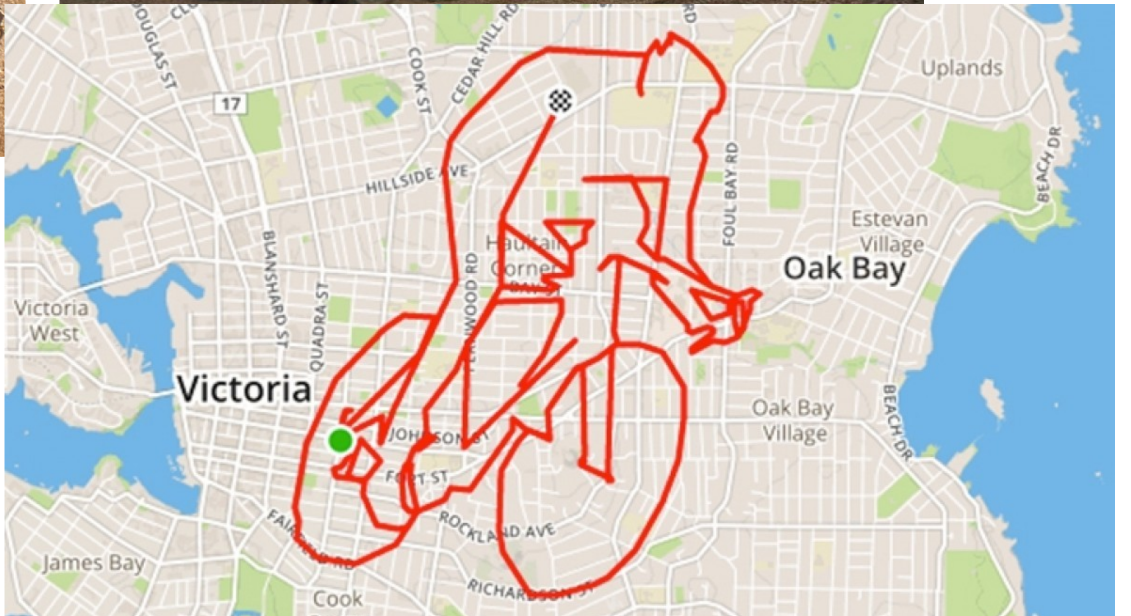


https://www.gps.gov/applications/

Applications







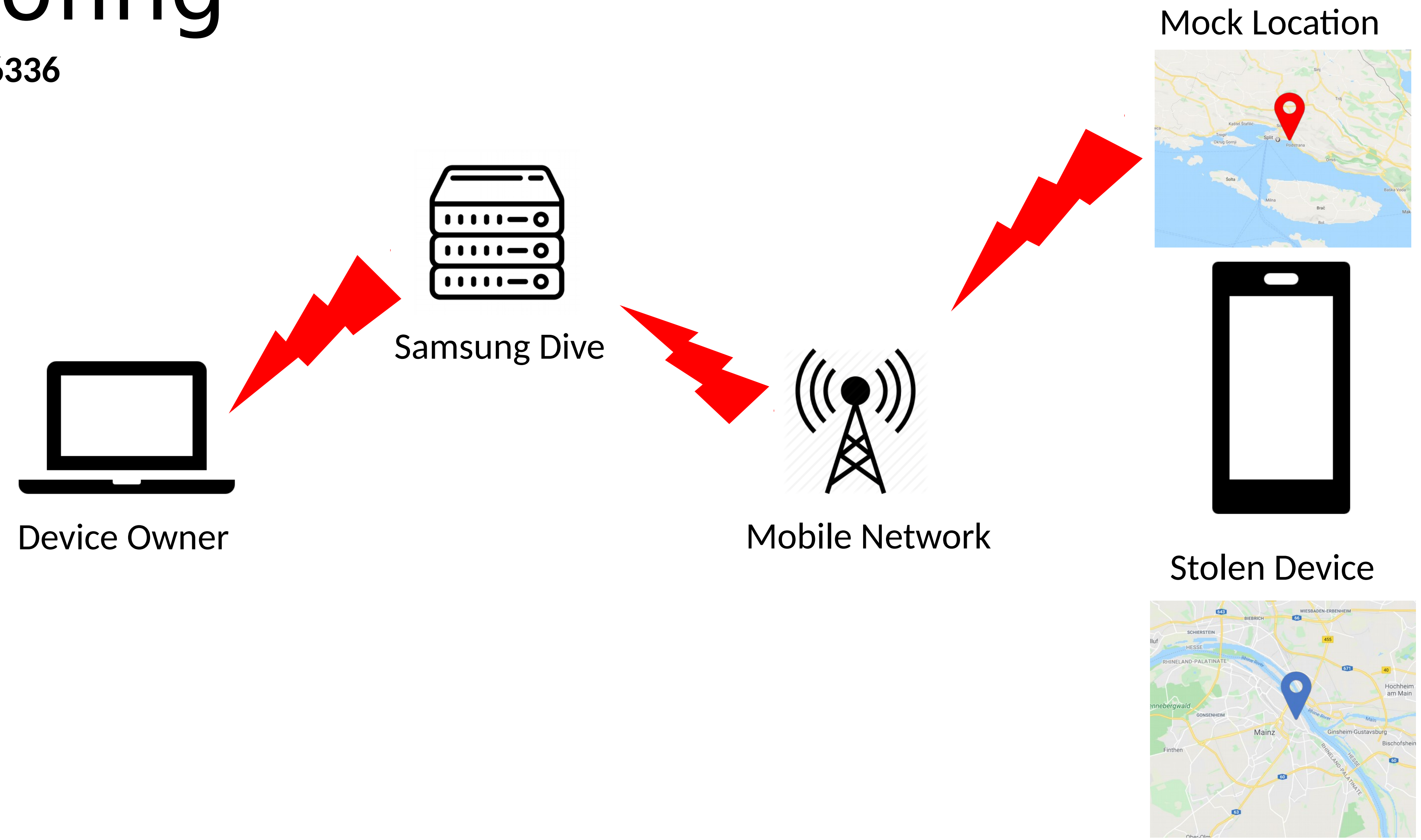
GPS is used in many industries, such as agriculture, aviation, geology and surveying, and military operations.

It is also used in day-to-day activities such as navigating on a road trip or tracking fitness activities like running or cycling.

STRIDE Assessment

Spoofing

CVE-2012-6336



- https://thehackernews.com/2012/12/manufacture-based-gps-tracking-services.html
- https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2012-6336

Spoofing

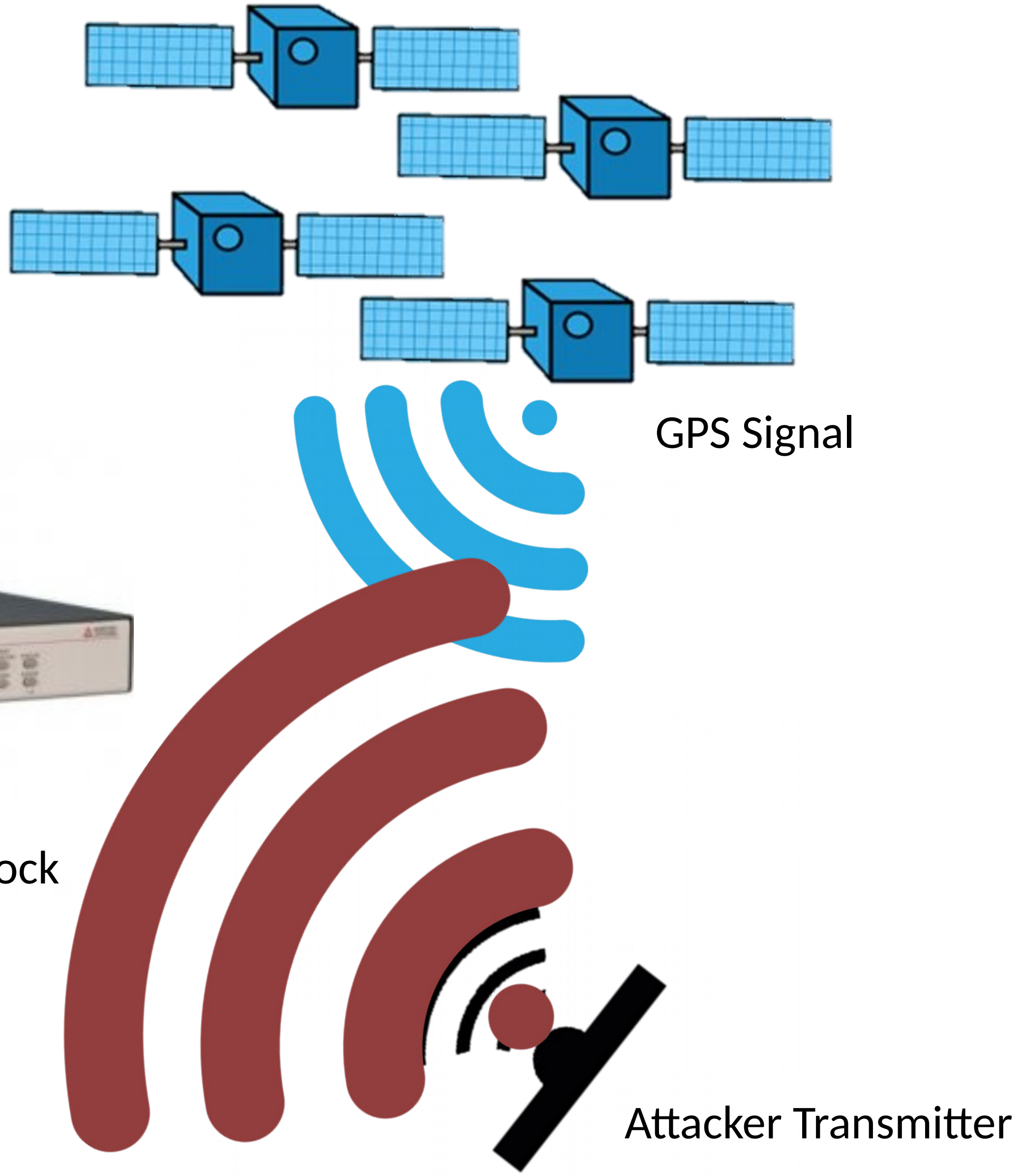
CVE-2014-9194

Arbiter Systems has identified a GPS clock spoofing vulnerability in its 1094B clock.

An attacker with specialized radio equipment and knowledge could transmit signals that can disrupt the clock.

Arbiter Model 1094B GPS Substation Clock

Used in electric utilities substations
Depends on GPS for precision timing



https://us-cert.cisa.gov/ics/advisories/ICSA-14-345-01

Tampering

SCOUT GPS LINK Vulnerability – Vulnerability in Scout GPS App provides access to multimedia screen in Toyota and Lexus Vehicles

hash (h1, salt1, round1) →h2.

For each incoming message enclosed by the connectivity protocol *MQTT*, the broker verifies the access code *AC* firstly.

hash (AC, salt2, round2) → h3;
hash (h3, salt1, round1) → h4.

By comparing *h2* and *h4*, the *Scout GPS Link* decides to accept or deny the message

An attacker can move the comparison process from *h2* and *h4* to *h1* and *h3* to bypass this security mechanism -- *h1* and *hash (AC, salt2, round2)*. All these values are encoded in the binary code.

Since the app accepts any connections, attacker can remotely find the *Scout GPS Link* users by scanning port 7050.

Using MQTT protocol, messages can be published or subscribed

```
mqttc.publish("uma/jsonrpc/mobile","{"jsonrpc":"2.0","method":"DrivingRestriction","params":
{"level":"50"}}",1,True)
```

https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2019-14951

https://sites.google.com/site/iosappnss/more-vulnerable-apps-and-libraries

Repudiation & Information Disclosure

CVE-2017-5237

Given knowledge of the EV-07S's registered phone number, the EV-07S device can be reset to factory level setting by sending "RESET!" as a command in an SMS message to the device

Manual fuzzing test via SMS revealed that the command "REBOOT!" will cause the device to respond with the message "Format error!".
Due to providing this negative response, a malicious actor could use this command to enumerate all devices by trying all likely phone numbers, commonly known as a war dialing operation, using SMS messages containing the "REBOOT!" command.

An unauthenticated attacker can poison the realtime tracking data by injecting device data to the server at www.smart-tracking.com over TCP port 5050. The attacker can do this only if they know a device's IMEI number, but that data is learnable through an additional vulnerability: R7-2016-28.5

An authenticated user can gain access to others users configuration and device GPS data if they know or guess a valid userId, device IMEI or TrackerID

EV-07S GPS Tracker

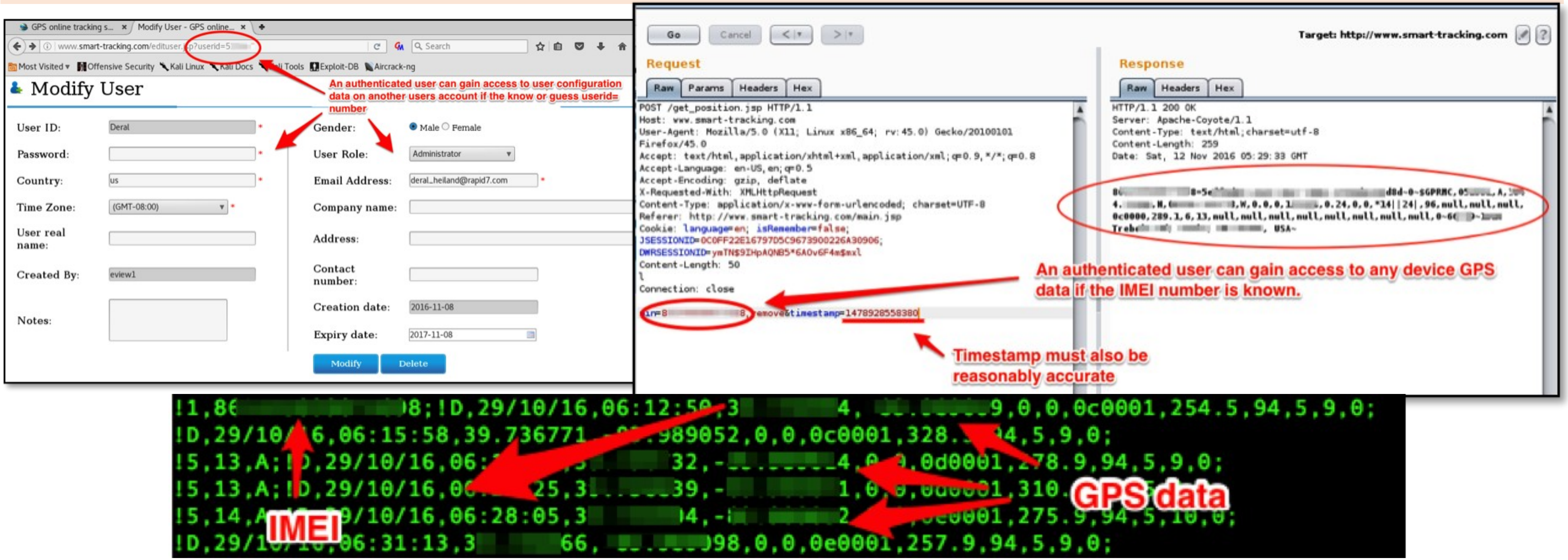
The EV-07S is a personal GPS tracker device used for personal safety and security

https://blog.rapid7.com/2017/03/27/r7-2015-28-multiple-eview-ev-07s-gps-tracker-vulnerabilities/

Information Disclosure

The web application used for realtime tracking web application, hosted at http://www.smart-tracking.com , did not utilize SSL/TLS encryption for HTTP services. Also the EV-07S device passed IMEI and GPS data to this website over the Internet on TCP port 5050 without any encryption

An authenticated user can gain access to others users configuration and device GPS data if they know or guess a valid userId, device IMEI or TrackerID.



https://blog.rapid7.com/2017/03/27/r7-2015-28-multiple-eview-ev-07s-gps-tracker-vulnerabilities/

Denial of Service

CVE-2016-5348

Android devices can be crashed remotely forcing a halt and then a soft reboot by a MITM attacker manipulating assisted GPS/GNSS data provided by Qualcomm. The Android issue was fixed by in the October 2016 Android bulletin.

The device makes periodic calls to the Qualcomm servers to retrieve gpsOneXtra assistance files. These requests were performed almost every time the device connected to a WiFi network at the following URLs:

```
http://xtra1.gpsonextra.net/xtra.bin
http://xtra2.gpsonextra.net/xtra.bin
http://xtra3.gpsonextra.net/xtra.bin

http://xtrapath1.izatcloud.net/xtra2.bin
http://xtrapath2.izatcloud.net/xtra2.bin
http://xtrapath3.izatcloud.net/xtra2.bin
```

Both the Java and the C++ code do not check how large the data file actually is. If a file is served that is larger than the memory available on the device, this results in all memory being exhausted and the phone halting and then soft rebooting. The soft reboot was sufficient to recover from the crash and no data was lost.

To attack, an man-in-the-middle attacker located anywhere on the network between the phone being attacked and Qualcomm's servers can initiate this attack by intercepting the legitimate requests from the phone, and substituting their own, larger files. Because the default Chrome browser on Android reveals the model and build of the phone, it would be possible to derive the maximum memory size from that information and deliver the appropriately sized attack file. Possible attackers can be hostile hotspots, hacked routers, or anywhere along the backbone.

https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2016-5348
https://www.exploit-db.com/exploits/40502

Escalation of Privileges

CVE-2020-5246

Traccar GPS Tracking System before version 4.9 has a LDAP injection vulnerability. It occurs when user input is being used in LDAP search filter. By providing specially crafted input, an attacker can modify the logic of the LDAP query and get admin privileges. The issue only impacts instances with LDAP configuration and where users can craft their own names. This has been patched in version 4.9.

```
            if (this.adminFilter != null) {
                try {
                    InitialDirContext context = initContext();
-                   String searchString = adminFilter.replace(":login", accountName);
+                   String searchString = adminFilter.replace(":login", encodeForLdap(accountName));
                    SearchControls searchControls = new SearchControls();
                    searchControls.setSearchScope(SearchControls.SUBTREE_SCOPE);
                    NamingEnumeration<SearchResult> results = context.search(searchBase, searchString, searchControls);
@@ -107,7 +107,7 @@ public InitialDirContext initContext() throws NamingException {
        private SearchResult lookupUser(String accountName) throws NamingException {
            InitialDirContext context = initContext();

-           String searchString = searchFilter.replace(":login", accountName);
+           String searchString = searchFilter.replace(":login", encodeForLdap(accountName));

            SearchControls searchControls = new SearchControls();
            String[] attributeFilter = {idAttribute, nameAttribute, mailAttribute};
@@ -176,4 +176,34 @@ public boolean login(String username, String password) {
            return false;
        }

+       public String encodeForLdap(String input) {
+           if( input == null ) {
+               return null;
+           }
+           StringBuilder sb = new StringBuilder();
```

In the below example a query is constructed to validate a user's credentials for the purpose of logging in.

String filter = "(&(USER = " + user_name + ") (PASSWORD = " + user_password + "))";

An attacker can enter a crafted input for the variable user_name such as johnDoe)(&) and any value for password the finished query will become (&(USER = johnDoe)(&))(PASSWORD = pass)).

Only the first portion of this query is processed by the LDAP server (&(USER = johnDoe)(&), which always evaluates to true allowing the attacker to gain access to the system without needing to provide valid user credentials.

https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2020-5246
https://github.com/traccar/traccar/commit/e4f6e74e57ab743b65d49ae00f6624a20ca0291e
https://en.wikipedia.org/wiki/LDAP_injection