2021
FCC Speed Test App
Technical Description

Office of Engineering and Technology
Table of Contents

I. Introduction ........................................................................................................................................... 3

II. Mobile Broadband Performance Testing ................................................................................................. 3
   A. FCC Speed Test Methodology Overview .................................................................................................. 3
      1. Measurement Process ............................................................................................................................ 4
      2. Testing and Performance Metrics ......................................................................................................... 5
   B. Test System Architecture ....................................................................................................................... 5
   C. Download Speed and Upload Speed ........................................................................................................ 6
   D. Latency and Packet Loss ......................................................................................................................... 9
   E. Testing Schedule ..................................................................................................................................... 10
   F. Test Server and Data Collection ............................................................................................................ 11
      1. Android Test Flow, Environmental Data, and JSON Format ................................................................. 12
      2. iOS Test Flow, Environmental Data, and JSON Format ......................................................................... 27

III. FCC Speed Test App Privacy Notice and Terms of Use (updated May 13, 2021) .................................. 37
   A. What Information We Collect ................................................................................................................ 37
      1. Location (with your permission) ............................................................................................................. 37
      2. Time of Data Collection ......................................................................................................................... 38
      3. Device Type and Operating System Version ........................................................................................ 38
      4. Mobile Wireless Network Performance and Characteristics ................................................................. 38
      5. Broadband Performance ......................................................................................................................... 38
   B. Data Sharing .......................................................................................................................................... 39
      1. Internally ................................................................................................................................................ 39
      2. Third-Party Partners Who Help Us Run Our Test and Support Our Program .................................... 39
      3. Pursuant to Legitimate Requests by Law Enforcement or Where Otherwise Required by Law or Regulation ................................................................................................................................................. 40
      4. De-Identified Data Available to the Public ............................................................................................ 40
   C. Data Consumption .................................................................................................................................. 40
   D. Stopping Your Participation .................................................................................................................... 41
   E. Other Terms of Use of the Application .................................................................................................... 41

List of Tables and Figures

- Figure 1: Mobile Broadband Test Architecture
- Table 1: Android Data Dictionary
- Table 2: iOS Data Dictionary
I. Introduction

This technical description document ("the App Description") for the FCC Speed Test mobile application ("the FCC Speed Test App" or "the App")\(^1\) provides detailed information on the metrics and methodologies used by the FCC Speed Test App to conduct connection performance tests on mobile broadband networks. The App Description includes reviews of the techniques implemented for mobile network performance measurement, including the test sequence and methodologies, environmental data for Android and iOS operating systems, data reporting features, and an overview of the backend test system architecture.\(^2\)

II. Mobile Broadband Performance Testing

This section describes the technical features of the supporting system architecture and the FCC Speed Test App, as well as other technical aspects of the methods employed to measure mobile broadband performance.\(^3\)

A. FCC Speed Test Methodology Overview

The App’s technologies and methodologies are developed collaboratively with a technical solutions contractor, SamKnows, Inc.\(^4\) The contractor supports the FCC’s efforts to develop and maintain the App.

The FCC Speed Test App has two independent mobile applications due to the differences between iOS and Android operating systems, security features, and hardware. The iOS App is written in the Swift programming language and the Android version is written in the Kotlin programming language.

The iOS operating system supports iPhone and iPad hardware devices. The iOS operating system’s security features do not expose many of the radio parameters that the FCC Speed Test App relies on to verify the network environment. Additionally, the iOS operating system (unlike the Android operating system) allows only for tests to be performed manually in the foreground and does not allow for automated scheduled testing of the FCC Speed Test.


\(^2\) The FCC recently contracted for updates to the App to implement additional capabilities for submitting mobile challenge and crowdsourced data as part of the FCC’s Broadband Data Collection program, [https://sam.gov/opp/cbd05b3514244c62996ef84ce734ea37/view](https://sam.gov/opp/cbd05b3514244c62996ef84ce734ea37/view). OET intends to publish a revised version of this technical description document in the future that includes additional details on the BDC functionalities of the App upon release of an updated version of the App.

\(^3\) This 2021 Technical Description updates the earlier 2016 and 2018 versions to describe the FCC Speed Test app currently in use.

\(^4\) See generally [https://www.samknows.com/](https://www.samknows.com/).
The Android operating system is supported on hardware devices from multiple different vendors such as Google, Samsung, Motorola, Nokia, etc. The Android operating system provides a consistent Application Programming Interface (API), which exposes many radio parameters for use by applications. Many mobile device vendors support the radio parameters that the FCC Speed Test relies on to verify the network environment.

1. Measurement Process

The measurements that provide the underlying data rely both on the measurement client that initiates the test process and measurement servers that function as a test endpoint to the client’s measurements. The measurement client consists of a publicly available app, the FCC Speed Test App, which can be obtained at no cost without in-app ads and has been downloaded on hundreds of thousands of Android and iOS devices in locations across the nation. Measurement tests can be performed on any mobile broadband service provider’s network subscribed to by consumers running the App.

Currently, the measurement servers are hosted by StackPath and are located in ten cities (thirteen locations, some cities with multiple sites) across the United States near a point of interconnection between the broadband provider’s network and the network on which the measurement server resides. Section F provides details of the thirteen measurement servers.

This methodology focuses on the performance of the specific mobile broadband network under test. When a test sequence initiates, the App will check whether the form of connection is cellular or Wi-Fi. If the connection is over a cellular network, the App will further identify its connection technology type; i.e., 3G, 4G, or 5G. If the connection is over a Wi-Fi network, the App will test and report the performance of the Wi-Fi network connection through its connected Internet service.

Once the App selects a measurement server, the metrics are derived from traffic exchanged between a measurement client (the App) and the selected measurement server. The tests use the measurement server for which the latency between the measurement client and server is the lowest value, which means that the selected server may not be geographically the closest from the measurement client location. As a result, the metrics measure performance along a specific path within each mobile broadband provider’s network, through the point of interconnection between the mobile broadband provider’s network, and the network on which the chosen measurement server resides.

---

5 The FCC Speed Test App Privacy Notice and Terms of Use for the FCC Speed Test App describes the steps taken to protect each user’s privacy. See FCC, FCC Speed Test App Privacy Notice and Terms of Use, https://www.fcc.gov/general/mobile-broadband-performance-application-privacy-notice-and-terms-use, (last visited Nov. 1, 2021). The policy and technical measures of the test architecture, client and data processing protect personally identifiable information (PII) of the mobile devices reporting results to the FCC.
2. Testing and Performance Metrics

Each test sequence is preceded by the measurement client selecting a test server as follows:

- **Test server selection**: Measures and compares for the lowest round-trip time by ICMP ping (default for Android) or TCP ping (default for iOS and fallback for Android) from the 10 nearest servers typically over 2 seconds with a 10-second timeout if servers fail to respond.

Once the server is selected, the App performs the following active tests of mobile broadband performance:

- **Download speed**: Measures the average download speed in megabits per second over a maximum 8-second time interval to transmit up to 1,000 megabytes (MB). This includes a fixed 3-second warm-up period (*i.e.*, 3 seconds warm-up and 5 seconds testing).

- **Upload speed**: Measures the average upload speed in megabits per second over a maximum 8-second time interval to transmit up to 1,000 megabytes (MB). This includes a fixed 3-second warm-up period (*i.e.*, 3 seconds warm-up and 5 seconds testing).

- **Latency**: Measures the average round-trip time in milliseconds of up to 200 UDP (User Datagram Protocol) data packets that are acknowledged as received individually within 2 seconds or are recorded as lost. The packets are sent over a fixed 5-second time interval. No warm-up period is applied to latency.

- **Packet Loss**: The ratio of the number of UDP packets either not acknowledged by the measurement server, or acknowledged as received after 2 seconds, to the number of total packets sent from the client.

- **Jitter**: The amount of variation in latency measured in milliseconds. Jitter is calculated across all packets (up to 200 packets, assuming no loss) using the Packet Delay Variation (PDV) approach.

A more detailed discussion of the measurement test methodology is presented in Section C, below.

B. Test System Architecture

The test system architecture focuses on measuring the performance of mobile broadband networks. The measurement servers are located across the U.S., which are hosted by a major content delivery network (CDN) with servers in proximity at multiple Internet exchange points/peering locations for unbiased network performance measurement.

As illustrated below in Figure 1, the App executes tests, as part of the measurement system that comprises a distributed network of “Test Devices” (*i.e.*, measurement clients), used to accurately measure the performance of mobile broadband connections based on real-world usage. Upon completion of each test sequence, the App reports measurement results to the reporting database (“Data Collection”), where the data is collated on the reporting platform (“Reporting”). The App measures download and upload speeds, round-trip latency, packet loss, and jitter by exchanging
information with the selected measurement servers ("Test Targets"), which the App contacts when a Test Device manually initiates a test sequence or according to the test schedule if prescheduled in Android.

The measurement servers, each supporting a 100 Gbps capacity, used for mobile broadband measurement are hosted by StackPath and are distributed nationally to enable a measurement client to select the host server with the least latency.

**Figure 1: Mobile Broadband Test Architecture**

![Mobile Broadband Test Architecture](image)

C. Download Speed and Upload Speed

Two separate tests measure the download and upload speed in megabits per second by establishing multi-TCP connections to perform HTTP GET and POST requests to a target test node. Binary non-zero content, herein referred to as the payload, is hosted on a web server on the target test node. Each test cycle begins with a fixed 3-second warmup, followed by the download or upload speed test. Each speed test concludes either after 1,000 MB of payload is transferred or after a maximum elapsed time period of 5 seconds. The total time for the test cycle is thus no more than 8 seconds, and the client will attempt to download as much of the payload as possible for the duration of the test. The payload and all other testing parameters are configurable and may be subject to change in the future.

Tests are executed using three concurrent TCP connections for byte streams. Hence, each connection used in the test counts the numbers of bytes of the target payload transferred between two points in time, and calculates the speed of each thread as the number of bits transferred over the number of seconds in the active test window. The speeds of the three threads are then summed to determine the total speed.

Factors such as TCP slow start are taken into account by transferring a portion of the target payload before the real testing begins. This “warmup” period will complete after a fixed duration of three seconds. Three individual connections are established, each on its own thread, and are confirmed as all having completed the warmup period before the five-second timing begins.
Content downloaded is copied to /dev/null or equivalent (i.e., it is discarded). Payload content for upload tests is generated and streamed on the fly from /dev/urandom (or equivalent).

The following is an example of the calculation performed for a multi-connection test using three concurrent TCP connections:

- \( S = \text{Speed (Bytes per second)} \)
- \( B = \text{Bytes (Bytes transferred)} \)
- \( T = \text{Time (Seconds) (between start time point and end time point)} \)
- \( S_1 = B_1 / T_1 \) (speed for Thread 1 calculation)
- \( S_2 = B_2 / T_2 \) (speed for Thread 2 calculation)
- \( S_3 = B_3 / T_3 \) (speed for Thread 3 calculation)
- \( \text{Speed} = S_1 + S_2 + S_3 \)
- Example values from a 3 MB payload:
  - \( B_1 = 3077360 \) \( T_1 = 15.583963 \)
  - \( B_2 = 2426200 \) \( T_2 = 15.535768 \)
  - \( B_3 = 2502120 \) \( T_3 = 15.536826 \)
  - \( S_1 = B_1/T_1 = 197469.668017 \)
  - \( S_2 = B_2/T_2 = 156168.655454 \)
  - \( S_3 = B_3/T_3 = 161044.475879 \)
  - \( S_1 + S_2 + S_3 = \text{Total Throughput of the line} = 197469.668017 + 156168.655454 + 161044.475879 = 514682 \text{ (Bps)} * 0.000008 = 4.12 \text{ Mbps} \)

The following pseudo-code describes the algorithm of the test in more detail:

```java
// Call the test
download_test( target, 6500, '/1000MB.bin', 3 sec, 5 sec, 3 )

// Define the test
function download_test( target, port, uri, warmupmaxtime, transfersmaxtime, numberofthreads ) {
    threads = array()
    for i=0; i<numberofthreads; i++ {
        t = new Thread( speedtest_thread( target, port, uri ) )
        t.start()
        threads.put(t)
    }

    warmup_complete = false
    warmup_bytes_transferred = 0
    warmup_time_elapsed = 0
    bytes_transferred = 0
    time_elapsed = 0
```
```c
while( 1 ) {

    // Get latest statistics for each thread
    foreach t in threads {

        // check to see if there's an error on the thread
        if t.status = ERROR
            fail and exit with error status

        // Check to see if the warmup was not complete but now is
        if (!warmup_complete) {
            warmup_bytes_transferred +=
            t.bytes_transferred
            warmup_time_elapsed = now() -
            t.time_start
        }

        // Check to see if the test has completed
        if (warmup_complete) {
            bytes_transferred += t.bytes_transferred
            time_elapsed =
            now() - t.time_start
        }
    }

    // If warmup is now complete, mark it as such and begin testing
    if (!warmup_complete &&
        warmup_time_elapsed >
        warmupmaxtime) {
    }

    // Check to see if we can finish
    if (warmup_complete) {

        // deduct warmup
        bytes_transferred -=
        warmup_bytes_transferred
        time_elapsed -=
        warmup_time_elapsed

        // Check to see if we're done
        if ( time_elapsed > transfermaxtime ) {
            break;
        }
    }

    // calculate final result
    speed = bytes_transferred / time_elapsed
}

// Multiple threads of the below will be called function
speedtest_thread( target, port, uri ) {
```
bytes_transferred = 0
time_start = now()
status = WAITING

try{
    client = new http_client( target, port, uri )
status = CONNECTED
    while( data =
client.recv() ) {
    bytes_transferred +=
data.length
    ) } catch { 
    status = ERROR
}
}

D. Latency and Packet Loss

The UDP Latency and Packet Loss tests measure the round-trip time of UDP packets for a fixed 5-second duration between the device and a target test site. Each 160-byte packet consists of an 8-byte sequence number and an 8-byte timestamp. If an acknowledgement is not received back within two seconds of sending, the packet is treated as lost. The test sends up to 200 packets (datagrams) and records the number of packets sent, the average round-trip time of responses, and the total number of packets lost.

The following pseudo-code describes the algorithm of the test in more detail:

call latencytest( server, 5000, 500ms, 2s, 200, 5s )

function latencytest ( host, port, interPacketTime, delayTimeout, datagrams, maxTime ) {
    sent = 0
    received = 0
    lost = 0
    rttArray = array()
    start = now()

    sock = socket(host,port) for
    i = 0; i < datagrams; i++ {
        t1 = now()
        send(sock, i) sent++

        try {
            recv(i, delayTimeout )
            t2 = now() rtt = t2 -
            t1 insert( rttArray,
            rtt ) received++
        }
    }
}
E. Testing Schedule

The FCC Speed Test App for Android and iOS both enable a user to initiate “manual” tests. In addition, and unlike many commercially available speed tests which are solely user-initiated, the FCC Speed Test App for Android also contains a default setting which causes the App to execute at automated prescheduled test intervals. The automated testing function can be disabled and the App can be configured to start a test only when manually executed. iOS devices do not have automated testing capability and can only execute the speed test manually. The scheduled testing approach minimizes extraneous factors that could degrade a statistically accurate measure of mobile broadband performance. Automatic background tests contribute valuable, high-quality data about mobile broadband performance.

It is possible to schedule tests to run with a fixed test frequency on Android. The test frequency is not exact, and the framework can decide to postpone the running of the tests especially for higher frequencies, or when the device is in the Doze mode (sleeping after a long period of inactivity).

The allowed frequencies are:

- Once every 24 hours
- Once every 12 hours
- Once every 6 hours
- Once an hour
- Every 15 mins

---

6 The Apple iOS does not support features necessary for scheduled test execution on the App for iPhone and thus the App for iPhone only supports manual testing at this time.
It is possible to set constraints to stop running the tests unless some conditions are fulfilled. The possible constraints are:

- Wi-Fi only
- Allow while roaming
- Device idle
- Device charging

It is possible to enable/disable the showing of a notification with the results of the tests.

**F. Test Server and Data Collection**

The initial beta trials of the FCC Speed Test App and the analysis of the data collected by the various beta versions of the App ensured that once the App was officially launched, all collected data met the FCC's technical privacy and engineering requirements. In addition to data collected from active download speed, upload speed, latency, and packet loss tests, data related to the radio characteristics of the device, information about the handset type and operating system version, the GPS coordinates of the handset at the time each test is run (if available), the date and time of the observation, and other metrics are recorded on the handset in JSON (JavaScript Object Notation) nested data elements within flat files. These JSON files can be exported by the user and are transmitted to SamKnows' storage servers at periodic intervals after the completion of active test measurements.

Once the JSON files are received by SamKnows, they are converted into row results that are inserted into a series of SQL tables using software developed by SamKnows, Inc. Each collection of results from a "test" share a primary key, which can be combined with local timestamp or metric values to combine the active and passive metrics that are connected during a single test. The Data Dictionaries for the JSON and CSV formats for Android and iOS describe in detail the nested format of the radio, handset, active test result, and other data recorded on the handset prior to transmission to the storage infrastructure, as well as the converted CSV table structures and data elements can be found on the Measuring Mobile Broadband website.

All measurements are carried-out against any one of our U.S. StackPath test servers. Currently, these are:

---

7 The FCC has contracted with a broadband measurement company, SamKnows, to help manage the program and conduct its technical functions. SamKnows performs functions such as writing and maintaining the software that performs the test measurements, responding to customer service requests, managing the App and the execution of tests, ensuring the integrity of and protecting the data, and providing analysis and support for the preparation of FCC reports. The terms of our contract with SamKnows include obligations to protect the data and your privacy.

8 The tests results are transferred depending on the available connectivity at the conclusion of the test and can be stored and forwarded when connectivity is immediately unavailable.

These servers are found in the **stackpath-us** target set, and are accessible via the Speedtest API at:


Tests should be performed against the "nearest" test server. This server selection is determined at the start of each test sequence, and the same server should be used for all tests in that sequence. The nearest test server should be determined for Android by measuring the latency using the ICMP protocol or, in case this process fails, by TCP ping on port 6500. For iOS, the default latency measurement is done by TCP ping on port 6500. The server with the lowest latency or the lowest TCP connect time should be used for measurements.

### 1. Android Test Flow, Environmental Data, and JSON Format

In a scenario where all test types are switched to active on the app’s Test screen, upon pressing **Start Tests** the following sequence of actions takes place:

1. The app fetches the **stackpath-us** target set from the Speedtest API.
2. The server selection process (described above) is performed against each of the targets in the list.
3. The app downloads a file to “poke” the antenna and eventually triggers the switching to 5G.
4. Pause of 2 seconds.
5. The Download test is performed against the test server identified as the "nearest" test server for the user’s current location, from the previous step. The result and a snapshot of the current environmental data (see later) is captured.
6. Pause of 1750 milliseconds.
7. The Upload test is performed against the chosen test server. The result and a snapshot of the current environmental data (see later) is captured.
8. Pause of 1750 milliseconds.
9. The latency/loss/jitter test is performed against the chosen test server. The result and a snapshot of the current environmental data (see later) is captured.
10. Pause of 1750 milliseconds.
11. The results are submitted to the backend Data Ingestion API and a resulting Test ID is retrieved.
12. The results for the entire test sequence are paired with their respective Test ID and stored locally.

At the end of a test sequence, the following information is displayed to the user:

- Download speed
- Upload speed
- Latency
- Packet loss
- Jitter
- Date and time of measurement
- The Wi-Fi or cellular network that was tested
- The ISP that the user used
- The test server location

If any test fails to obtain a measurement for any reason (such as when there is no network connectivity at all), then “FAILED” is shown for the respective test result. If any other data was unavailable, due to a failure, then a "-" is shown in its place.

**Table 1: Android Data Dictionary**

The table below shows all the data fields that are collected during the tests. The fields with the name that starts with “beginning” or “end” are the fields that are reported twice for each test: one at the beginning of the test and one at the end.

The following is an exhaustive list of all possible fields that are reported by the app. However, not all of these fields will be present in every test. Their presence depends on device support and network support (for example, the base station ID is only returned on CDMA).

Please also note that some field names may differ in data exports from the backend database. The complete backend field names are provided in the “2021 FCC Mobile App Data Dictionary (Raw Data) - Android and iOS.xlsx.”

<table>
<thead>
<tr>
<th>Field name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>testId</td>
<td>Unique identifier for a test.</td>
</tr>
<tr>
<td>carrier_name</td>
<td>The mobile carrier as reported by the operating system.</td>
</tr>
<tr>
<td>iso_country_code</td>
<td>The ISO country code (e.g. &quot;GB&quot;, &quot;HK&quot;).</td>
</tr>
<tr>
<td>manufacturer</td>
<td>Manufacturer of the mobile phone.</td>
</tr>
<tr>
<td>mobile_country_code</td>
<td>Numeric mobile country code (e.g. 234 = UK).</td>
</tr>
<tr>
<td>mobile_network_code</td>
<td>Numeric mobile network code (e.g. 15 = Vodafone).</td>
</tr>
<tr>
<td>---------------------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>model</td>
<td>Model of the mobile phone.</td>
</tr>
<tr>
<td>operating_system_version</td>
<td>Operating system name and version.</td>
</tr>
<tr>
<td>version</td>
<td>App version number.</td>
</tr>
<tr>
<td>submission_type</td>
<td>Indicates whether test was executed manually by the user (&quot;manual_test&quot;) or run automatically (&quot;scheduled_test&quot;).</td>
</tr>
<tr>
<td>utc_date_time</td>
<td>UTC timestamp of when the full test cycle completed.</td>
</tr>
<tr>
<td>dhcp_dns1</td>
<td>Primary DNS server provided by DHCP.</td>
</tr>
<tr>
<td>dhcp_dns2</td>
<td>Secondary DNS server provided by DHCP.</td>
</tr>
<tr>
<td>dhcp_gateway</td>
<td>Default gateway IP provided by DHCP.</td>
</tr>
<tr>
<td>dhcp_ip_address</td>
<td>IP address provided by DHCP.</td>
</tr>
<tr>
<td>dhcp_lease_duration</td>
<td>DHCP lease duration.</td>
</tr>
<tr>
<td>dhcp_netmask</td>
<td>DHCP netmask.</td>
</tr>
<tr>
<td>dhcp_server_address</td>
<td>DHCP server.</td>
</tr>
<tr>
<td>{beginning/end}_cross_traffic_rx_avg_bps</td>
<td>The average volume of downstream cross-traffic at the start/end of the test, excluding test traffic.</td>
</tr>
<tr>
<td>{beginning/end}_cross_traffic_rx_peak_bps</td>
<td>The peak volume of downstream cross-traffic at the start/end of the test, excluding test traffic.</td>
</tr>
<tr>
<td>{beginning/end}_cross_traffic_tx_avg_bps</td>
<td>The average volume of upstream cross-traffic at the start/end of the test, excluding test traffic.</td>
</tr>
<tr>
<td>{beginning/end}_cross_traffic_tx_peak_bps</td>
<td>The peak volume of upstream cross-traffic during the test, excluding test traffic.</td>
</tr>
<tr>
<td>{beginning/end}_location_lat</td>
<td>Latitude at the start of measurement.</td>
</tr>
<tr>
<td>{beginning/end}_location_lon</td>
<td>Longitude at the start of measurement.</td>
</tr>
<tr>
<td>{beginning/end}_location_time</td>
<td>Android GPS location time.</td>
</tr>
<tr>
<td>{beginning/end}_memory_app_used</td>
<td>Memory used in MB by the app at the start/end of the test.</td>
</tr>
<tr>
<td>{beginning/end}_memory_device_free</td>
<td>Memory free on the device in MB at the start/end of the test.</td>
</tr>
<tr>
<td>{beginning/end}_memory_device_used</td>
<td>Memory used on the device in MB at the start/end of the test (not always available).</td>
</tr>
<tr>
<td>{beginning/end}_network_is_available</td>
<td>Is a network connection available at the start/end of the test.</td>
</tr>
<tr>
<td>{beginning/end}_network_is_connected</td>
<td>Is the device connected to a network at the start/end of the test.</td>
</tr>
<tr>
<td>{beginning/end}_network_is_connected_or_connecting</td>
<td>Is the device in a connecting state at the start/end of the test.</td>
</tr>
<tr>
<td>{beginning/end}_network_is_fai lover</td>
<td>Indicates whether the current connection resulted from the device failing over from a disconnect from a different network at the start/end of the test.</td>
</tr>
<tr>
<td>{beginning/end}_network_is_roaming</td>
<td>Indicates whether the cellular network the user is on is different to their home network (i.e. they are roaming).</td>
</tr>
<tr>
<td><strong>{beginning/end}_network_state</strong></td>
<td>Current connection state of the device at the start/end of the test. Enum (Associated, Associating, Authenticating, Completed, Disconnected, Dormant, Four Way Handshake, Group handshake, Inactive, Interface Disabled, Invalid, Scanning, Uninitialized).</td>
</tr>
<tr>
<td><strong>{beginning/end}_network_subtype</strong></td>
<td>The numeric network subtype code at the start/end of the test.</td>
</tr>
<tr>
<td><strong>{beginning/end}_network_subtype_name</strong></td>
<td>The cellular radio technology in use at the start/end of the test: Enum (1xRTT, CDMA, EDGE, GPRS, GSM, IDEN, EHRPD, EVDO-0, EVDO-A, EVDO-B, HSDPA, HSPA, HSPAP, HSUPA, IWLAN, TD-SCDMA, UMTS, LTE, UNKNOWN, NR).</td>
</tr>
<tr>
<td><strong>{beginning/end}_network_type</strong></td>
<td>Indicates whether the user is connected to mobile or mobile and Wi-Fi simultaneously, at the start/end of the test.</td>
</tr>
<tr>
<td><strong>{beginning/end}_other_is_logged_in</strong></td>
<td>Boolean indicating if the user is logged in to the app at the start/end of the test.</td>
</tr>
<tr>
<td><strong>{beginning/end}_telephony_bandwidth</strong></td>
<td>Array of cell bandwidths (kHz) for the current serving cells.</td>
</tr>
<tr>
<td><strong>{beginning/end}_telephony_cellular_country_code</strong></td>
<td>Mobile network-operator country code at the start/end of the test.</td>
</tr>
<tr>
<td><strong>{beginning/end}_telephony_cellular_network_code</strong></td>
<td>Mobile network-operator network code at the start/end of the test.</td>
</tr>
<tr>
<td><strong>{beginning/end}_telephony_cell_signal_absolute_rf_channel_number</strong></td>
<td>For NR New Radio Absolute Radio Frequency Channel Number at the start/end of the test. Integer value in range [0, 3279165] or [@code null] if not available.</td>
</tr>
</tbody>
</table>
| **{beginning/end}_telephony_cell_signal_asu** | For WCDMA there is no standard definition of ASU. However, Android defines it as the the lesser of the following two results (for 1xRTT):

**RSSI Range (dBm): ASU Value**
-75..16: 16
-82..-76: 8
-90..-83: 4
-95..-91: 2
-100..-96: 1
...-101: 99

**Ec/Io Range (dB): ASU Value**
-90.: 16
-100..-91: 8
-115..-101: 4
-130..-116: 2
--150..-131: 1
...-151: 99

For GSM the range is 0..31, 99 |
<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>{beginning/end}_telephony_cell_signal_bands</code></td>
<td>Cellular signal bands.</td>
</tr>
<tr>
<td><code>{beginning/end}_telephony_cell_signal_carrier_aggregation</code></td>
<td>True if the cell is using carrier aggregation.</td>
</tr>
<tr>
<td><code>{beginning/end}_telephony_cell_signal_cell_connection</code></td>
<td>Cell connection status. This has the values:</td>
</tr>
<tr>
<td></td>
<td>0: No connection</td>
</tr>
<tr>
<td></td>
<td>1: Connected to primary serving cell</td>
</tr>
<tr>
<td></td>
<td>2: Connected to secondary serving cell</td>
</tr>
<tr>
<td></td>
<td>2147483647: Unknown</td>
</tr>
<tr>
<td><code>{beginning/end}_telephony_cell_signal_cell_identity</code></td>
<td>For NR the NR(New Radio 5G) Cell Identity. The 36-bit NR Cell Identity in range [0, 68719476735] or 0x7fffffffffffffffL if not available. For LTE 28-bit Cell Identity. For GSM 16-bit GSM Cell Identity described in TS 27.007, 0..65535.</td>
</tr>
<tr>
<td><code>{beginning/end}_telephony_cell_signal_mobile_country_code</code></td>
<td>Mobile Country Code at the start/end of the test as an integer or `{code null} if unknown.</td>
</tr>
<tr>
<td><code>{beginning/end}_telephony_cell_signal_mobile_network_code</code></td>
<td>Mobile Network Code at the start/end of the test as an integer or `{code null} if unknown.</td>
</tr>
<tr>
<td><code>{beginning/end}_telephony_cell_signal_network_type</code></td>
<td>One of the value &quot;NR&quot;, &quot;LTE&quot;, &quot;WCDMA&quot;, &quot;CDMA&quot;, &quot;GSM&quot;.</td>
</tr>
<tr>
<td><code>{beginning/end}_telephony_cell_signal_physical_cell_identity</code></td>
<td>Physical cell ID at the start/end of the test.</td>
</tr>
<tr>
<td></td>
<td>For LTE an integer value in range [0, 503].</td>
</tr>
<tr>
<td></td>
<td>For NR an integer value in range [0, 1007].</td>
</tr>
<tr>
<td><code>{beginning/end}_telephony_cell_signal_rssi</code></td>
<td>Received Signal Strength Indication (RSSI) in dBm.</td>
</tr>
<tr>
<td></td>
<td>For LTE the value range is [-113, -51] inclusively or CellInfo.UNAVAILABLE if unavailable. Reference: TS 27.007 8.5 Signal quality +CSQ. For GSM the value range is (-113, -51) or {code null}.</td>
</tr>
<tr>
<td><code>{beginning/end}_telephony_cell_signal_signal_level</code></td>
<td>Signal strength level at the start/end of the test.</td>
</tr>
<tr>
<td><code>{beginning/end}_telephony_cell_signal_received_signal_power</code></td>
<td>For LTE the RSSNR at the start/end of the test if available or `{code null} if not available.</td>
</tr>
<tr>
<td></td>
<td>For NR CSI reference 3GPP TS 38.215 Range: -140 dBm to -44 dBm or `{code null} if not available.</td>
</tr>
<tr>
<td><code>{beginning/end}_telephony_cell_signal_received_signal_quality</code></td>
<td>For LTE  RSRQ if available or `{code null} if unavailable.</td>
</tr>
<tr>
<td></td>
<td>For NR CSI reference signal received quality at the start/end of the test. Ref 3GPP TS 38.215. Range: -20 dB to -3 dB or `{code null} if not available.</td>
</tr>
<tr>
<td><code>{beginning/end}_telephony_cell_signal_strength</code></td>
<td>Signal strength as dBm.</td>
</tr>
<tr>
<td><code>{beginning/end}_telephony_cell_signal_timing_advance</code></td>
<td>For LTE timing advance value, as a value in range of 0..1282.</td>
</tr>
<tr>
<td></td>
<td>`{code null} is reported when there is no active RRC connection.</td>
</tr>
<tr>
<td>Key</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Refer to 3GPP 36.213 Sec 4.2.3.</td>
<td>For GSM timing advance between 0..219 symbols (normally 0..63). (@code null) is reported when there is no RR connection. Refer to 3GPP 45.010 Sec 5.8.</td>
</tr>
<tr>
<td>(beginning/end)_telephony_cell_signal_tracking_area_code</td>
<td>Tracking area code at the start/end of the test. 16 bit integer or (@code null) if not available.</td>
</tr>
<tr>
<td>(beginning/end)_telephony_cell_signal_base_station_identity_code</td>
<td>The base station identity code (CDMA only).</td>
</tr>
<tr>
<td>(beginning/end)_telephony_cell_signal_location_area_code</td>
<td>The cellular location area code.</td>
</tr>
<tr>
<td>(beginning/end)_telephony_cellular_operator_name</td>
<td>Alphabetic name of current registered operator at the start/end of the test. Availability: Only when user is registered to a network. Result may be unreliable on CDMA networks.</td>
</tr>
<tr>
<td>(beginning/end)_telephony_cell_signal_strength_asu</td>
<td>RSSI in ASU as reported by the cell strength. Asu is calculated based on 3GPP RSSI. Refer to 3GPP 27.007 (Ver 10.3.0) Sec 8.69 0..31, 99 or (@code null).</td>
</tr>
<tr>
<td>(beginning/end)_telephony_cell_signal_strength_network_type</td>
<td>Network type from the signal strength.</td>
</tr>
<tr>
<td>(beginning/end)_telephony_cell_signal_strength_signal_level</td>
<td>Signal strength level at the start/end of the test.</td>
</tr>
<tr>
<td>(beginning/end)_telephony_cell_signal_strength_received_signal_power</td>
<td>For LTE the RSSNR at the start/end of the test if available or (@code null) if not available.</td>
</tr>
<tr>
<td>(beginning/end)_telephony_cell_signal_strength_received_signal_quality</td>
<td>Reference signal received quality.</td>
</tr>
<tr>
<td>(beginning/end)_telephony_cell_signal_strength_signal_strength</td>
<td>The cellular signal strength at the start/end of the test.</td>
</tr>
<tr>
<td>(beginning/end)_telephony_cell_signal_strength_received_signal_noise_ratio</td>
<td>Reference signal signal-to-noise ratio or (@code null) if not available.</td>
</tr>
<tr>
<td>(beginning/end)_wifi_frequency</td>
<td>The Wi-Fi network frequency in MHz at the start/end of the test.</td>
</tr>
<tr>
<td>(beginning/end)_wifi_hidden_ssid</td>
<td>Boolean indicating whether the SSID the user is connected to is hidden at the start/end of the test.</td>
</tr>
<tr>
<td>(beginning/end)_wifi_link_speed</td>
<td>The Wi-Fi link speed in Mbit/s at the start/end of the test.</td>
</tr>
<tr>
<td>(beginning/end)_wifi_network_id</td>
<td>The ID number that the supplicant uses to identify this network configuration entry at the start/end of the test.</td>
</tr>
</tbody>
</table>
### 2021 FCC Speed Test App Technical Description

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{beginning/end}_wifi_rssi</td>
<td>The receive signal strength of the Wi-Fi network at the start/end of the test.</td>
</tr>
<tr>
<td>{beginning/end}_wifi_supplicant_state</td>
<td>Whether or not the Wi-Fi supplicant completed or not at the start/end of the test.</td>
</tr>
<tr>
<td>{beginning/end}_telephony_telemony_display_info_network_type</td>
<td>Network type as reported by the Telephony Display Info.</td>
</tr>
<tr>
<td>{beginning/end}_telephony_telemony_display_info_network_type</td>
<td>Network subtype as reported by the Telephony Display Info.</td>
</tr>
<tr>
<td>{beginning/end}_telephony_cellular_technology</td>
<td>The cellular radio technology generation at the start/end of the test. Enum 2g, 3g, 4g, 5g, simless.</td>
</tr>
<tr>
<td>{beginning/end}_telephony_conection_type</td>
<td>Primary field indicating whether the user is on cellular or Wi-Fi at the start/end of the test. Enum: cell, wi-fi.</td>
</tr>
<tr>
<td>{beginning/end}_telephony_sim_country_code</td>
<td>The SIM country code at the start/end of the test (note that this can differ to the mobile_country_code above if the user is roaming).</td>
</tr>
<tr>
<td>{beginning/end}_telephony_sim_network_code</td>
<td>The SIM network code at the start/end of the test (note that this can differ to the mobile_network_code above if the user is roaming).</td>
</tr>
<tr>
<td>{beginning/end}_telephony_sim_operator_name</td>
<td>The SIM operator name at the start/end of the test (note that this can differ to the carrier_name above if the user is roaming).</td>
</tr>
</tbody>
</table>

### Download and upload speed test fields for Android

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>utc_datetime</td>
<td>Date time of the test in the format &quot;2021-10-26T14:54:12Z&quot;.</td>
</tr>
<tr>
<td>local_datetime</td>
<td>Date time of the test in the format &quot;2021-10-26T15:54:12+01:00&quot; (called “dtime” in the backend database).</td>
</tr>
<tr>
<td>target</td>
<td>The test server in use.</td>
</tr>
<tr>
<td>duration</td>
<td>Duration of the test, in microseconds.</td>
</tr>
<tr>
<td>thread_count</td>
<td>The number of concurrent TCP connections used in the test.</td>
</tr>
<tr>
<td>warmup_duration</td>
<td>Time consumed for all the TCP streams to arrive at optimal window size (units: microseconds).</td>
</tr>
<tr>
<td>warmup_bytes</td>
<td>Bytes transferred for all the TCP streams during the warm-up phase.</td>
</tr>
<tr>
<td>bytes_transferred</td>
<td>Bytes transferred during the measurement phase.</td>
</tr>
<tr>
<td>bytes_sec</td>
<td>The measured speed, in bytes per second. Multiply by 0.000008 to convert to Mbit/s (only present in the backend database).</td>
</tr>
<tr>
<td>throughput</td>
<td>The measured speed, in bytes per second. Multiply by 0.000008 to convert to Mbit/s (same as bytes_sec field).</td>
</tr>
<tr>
<td>successes</td>
<td>Number of successes (always 1 or 0).</td>
</tr>
<tr>
<td>failures</td>
<td>Number of failures (always 1 or 0).</td>
</tr>
</tbody>
</table>
Latency, packet loss, and jitter test fields for Android

<table>
<thead>
<tr>
<th>Field name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>utc_datetime</td>
<td>Date time of the test in the format &quot;2021-10-26T14:54:12Z&quot;.</td>
</tr>
<tr>
<td>local_datetime</td>
<td>Date time of the test in the format &quot;2021-10-26T15:54:12+01:00&quot; (called “dtime” in the backend database).</td>
</tr>
<tr>
<td>target</td>
<td>The test server in use.</td>
</tr>
<tr>
<td>duration</td>
<td>Duration of the test, in microseconds.</td>
</tr>
<tr>
<td>packet_size</td>
<td>Size of packets used in the test (bytes).</td>
</tr>
<tr>
<td>stream_bits_per_sec</td>
<td>The bits per second of the transmitted stream.</td>
</tr>
<tr>
<td>packets_sent</td>
<td>Number of packets sent during test.</td>
</tr>
<tr>
<td>packets_received</td>
<td>Number of packets received during test.</td>
</tr>
<tr>
<td>jitter</td>
<td>Calculated variation in latency between packets flowing between device and server (microseconds).</td>
</tr>
<tr>
<td>round_trip_time</td>
<td>Round trip time in microseconds.</td>
</tr>
<tr>
<td>mean_opinion_score</td>
<td>Mean Opinion Score (MOS) during the latency test.</td>
</tr>
<tr>
<td>status_code</td>
<td>Internal enum indicating the status of the test (0 is success).</td>
</tr>
<tr>
<td>successes</td>
<td>Number of successes (always 1 or 0).</td>
</tr>
<tr>
<td>failures</td>
<td>Number of failures (always 1 or 0).</td>
</tr>
</tbody>
</table>

JSON Local Data Export Example for Android

Results are serialized into the following JSON format for the local data export functionality. Please note that not all fields from the table above will be present in every local data export. Fields may be excluded based upon device or network support.

The JSON format for submissions made to the backend database differ by one JSON object: an outermost object is introduced to indicate that the data belongs to the FCC panel and that the submission is anonymized. Given that the structure is almost identical to the one below, an example has been omitted from this document for the sake of brevity.

```json
{
"version": "2.0.4058",
"carrier_name": "",
"iso_country_code": "GB",
"manufacturer": "Samsung",
"mobile_country_code": "234",
"mobile_network_code": "20",
"model": "SM-N950F",
"operating_system_version": "Android 9",
"tests": {
"download": {
"bytes_transferred": 13617918,
"duration": 4994819,
"environment": {
"beginning": {
"cross_traffic": {
```
```
"rx_avg_bps": 6062510,
"rx_peak_bps": 6062510,
"tx_avg_bps": 97004,
"tx_peak_bps": 97004
},
"dhcp": {
  "dns1": "192.168.1.254",
  "dns2": "0.0.0.0",
  "gateway": "192.168.1.254",
  "ip_address": "192.168.1.99",
  "lease_duration": 86400,
  "netmask": "0.0.0.0",
  "server_address": "192.168.1.254"
},
"location": {
  "lat": 51.5081773,
  "lon": 0.0191087,
  "time": "2021-10-26T14:54:12+0000"
},
"memory": {
  "app_used": 9,
  "device_free": 1786,
  "device_used": 3553
},
"network": {
  "is_roaming": false,
  "subtype_name": "",
  "type": "mobile_wifi"
},
"telephony": {
  "bandwidth": "",
  "cellular_country_code": "234",
  "cellular_network_code": "10",
  "cell_signal": {
    "absolute_rf_channel_number": 115,
    "asu": 31,
    "base_station_identity_code": 2,
    "carrier_aggregation": false,
    "cell_connection": 0,
    "cell_identity_id": 13874,
    "location_area_code": 21212,
    "mobile_country_code": 234,
    "mobile_network_code": 10,
    "network_type": "GSM",
    "signal_level": 4,
    "signal_strength": -51,
    "timing_advance": 0
  },
  "cellular_operator_name": "",
  "cellular_technology": "simless",
  "connection_type": "wi-fi",
  "sim_country_code": "234",
  "sim_network_code": "20",
  "sim_operator_name": ""
},
"wifi": {
  "frequency": 2462,
  "hidden_ssid": false,
  "link_speed": 173,
"sim_network_code": "20"
"sim_operator_name": ""
},
"wifi": {
"frequency": 2462,
"hidden_ssid": false,
"link_speed": 173,
"network_id": 9,
"rssi": -37,
"supplicant_state": "completed"
}
},
"failures": 0,
"local_datetime": "2021-10-26T15:54:25+01:00",
"successes": 1,
"target": "sp2-vm-ashburn-us.samknows.com",
"thread_count": 3,
"throughput": 2726408,
"utc_datetime": "2021-10-26T14:54:25Z",
"warmup_bytes": 1749000,
"warmup_duration": 3005245
},
"latency": {
"duration": 5065802,
"environment": {
"beginning": {
"cross_traffic": {
"rx_avg_bps": 8174878,
"rx_peak_bps": 8174878,
"tx_avg_bps": 171343,
"tx_peak_bps": 171343
},
"dhcp": {
"dns1": "192.168.1.254",
"dns2": "0.0.0.0",
"gateway": "192.168.1.254",
"ip_address": "192.168.1.99",
"lease_duration": 86400,
"netmask": "0.0.0.0",
"server_address": "192.168.1.254"
},
"location": {
"lat": 51.5081716,
"lon": 0.0191263,
"time": "2021-10-26T14:54:04+0000"
},
"memory": {
"app_used": 8,
"device_free": 1821,
"device_used": 3518
},
"network": {
"is_roaming": false,
"subtype_name": "",
"type": "mobile_wifi"
},
"telephony": {
"bandwidth": "",
"duration": 5065802,
"environment": {
"beginning": {
"cross_traffic": {
"rx_avg_bps": 8174878,
"rx_peak_bps": 8174878,
"tx_avg_bps": 171343,
"tx_peak_bps": 171343
},
"dhcp": {
"dns1": "192.168.1.254",
"dns2": "0.0.0.0",
"gateway": "192.168.1.254",
"ip_address": "192.168.1.99",
"lease_duration": 86400,
"netmask": "0.0.0.0",
"server_address": "192.168.1.254"
},
"location": {
"lat": 51.5081716,
"lon": 0.0191263,
"time": "2021-10-26T14:54:04+0000"
},
"memory": {
"app_used": 8,
"device_free": 1821,
"device_used": 3518
},
"network": {
"is_roaming": false,
"subtype_name": "",
"type": "mobile_wifi"
},
"telephony": {
"bandwidth": "",
"duration": 5065802,
"cellular_country_code": "234",
"cellular_network_code": "10",
"cell_signal": {
  "absolute_rf_channel_number": 115,
  "asu": 31,
  "base_station_identity_code": 2,
  "carrier_aggregation": false,
  "cell_connection": 0,
  "cell_identity_id": 13874,
  "location_area_code": 21212,
  "mobile_country_code": 234,
  "mobile_network_code": 10,
  "network_type": "GSM",
  "signal_level": 4,
  "signal_strength": -91,
  "timing_advance": 0
},
"cellular_operator_name": "",
"cellular_technology": "simless",
"connection_type": "wi-fi",
"sim_country_code": "234",
"sim_network_code": "10",
"sim_operator_name": ""
},
"wifi": {
  "frequency": 2462,
  "hidden_ssid": false,
  "link_speed": 173,
  "network_id": 9,
  "rssi": -51,
  "supplicant_state": "completed"
},
"end": {
  "cross_traffic": {
    "rx_avg_bps": 5820067,
    "rx_peak_bps": 8174878,
    "tx_avg_bps": 105017,
    "tx_peak_bps": 171343
  },
  "dhcp": {
    "dns1": "192.168.1.254",
    "dns2": "0.0.0.0",
    "gateway": "192.168.1.254",
    "ip_address": "192.168.1.99",
    "lease_duration": 86400,
    "netmask": "0.0.0.0",
    "server_address": "192.168.1.254"
  },
  "location": {
    "lat": 51.5081773,
    "lon": 0.0191087,
    "time": "2021-10-26T14:54:12+0000"
  },
  "memory": {
    "app_used": 9,
    "device_free": 1790,
    "device_used": 3549
  }
}
"network": {  
  "is_roaming": false,
  "subtype_name": "",
  "type": "mobile_wifi"
},
"telephony": {  
  "bandwidth": "",
  "cellular_country_code": "234",
  "cellular_network_code": "10",
  "cell_signal": {  
    "absolute_rf_channel_number": 115,
    "asu": 31,
    "base_station_identity_code": 2,
    "carrier_aggregation": false,
    "cell_connection": 0,
    "cell_identity_id": 13874,
    "location_area_code": 21212,
    "mobile_country_code": 234,
    "mobile_network_code": 10,
    "network_type": "GSM",
    "signal_level": 4,
    "signal_strength": -51,
    "timing_advance": 0
  },
  "cellular_operator_name": "",
  "cellular_technology": "simless",
  "connection_type": "wi-fi",
  "sim_country_code": "234",
  "sim_network_code": "20",
  "sim_operator_name": ""
},
"wifi": {  
  "frequency": 2462,
  "hidden_ssid": false,
  "link_speed": 173,
  "network_id": 9,
  "rssi": -37,
  "supplicant_state": "completed"
}
"failures": 0,
"jitter": 15098,
"local_datetime": "2021-10-26T15:54:12+01:00",
"mean_opinion_score": 1,
"packet_size": 160,
"packets_received": 200,
"packets_sent": 200,
"round_trip_time": 146558,
"status_code": 0,
"stream_bits_per_sec": 51200,
"successes": 1,
"target": "sp2-vm-ashburn-us.samknows.com",
"utc_datetime": "2021-10-26T14:54:12Z"
"testId": "1635597254",
"upload": {  
  "bytes_transferred": 5776536,
  "duration": 5000122,
"environment": {
  "beginning": {
    "cross_traffic": {
      "rx_avg_bps": 28797,
      "rx_peak_bps": 28797,
      "tx_avg_bps": 879401,
      "tx_peak_bps": 879401
    },
    "dhcp": {
      "dns1": "192.168.1.254",
      "dns2": "0.0.0.0",
      "gateway": "192.168.1.254",
      "ip_address": "192.168.1.99",
      "lease_duration": 86400,
      "netmask": "0.0.0.0",
      "server_address": "192.168.1.254"
    },
    "location": {
      "lat": 51.5081807,
      "lon": 0.0191033,
      "time": "2021-10-26T14:54:33+0000"
    },
    "memory": {
      "app_used": 6,
      "device_free": 1897,
      "device_used": 3442
    },
    "network": {
      "is_roaming": false,
      "subtype_name": "",
      "type": "mobile_wifi"
    },
    "telephony": {
      "bandwidth": "",
      "cellular_country_code": "234",
      "cellular_network_code": "10",
      "cell_signal": {
        "absolute_rf_channel_number": 115,
        "asu": 31,
        "base_station_identity_code": 2,
        "carrier_aggregation": false,
        "cell_connection": 0,
        "cell_identity_id": 13874,
        "location_area_code": 21212,
        "mobile_country_code": 234,
        "mobile_network_code": 10,
        "network_type": "GSM",
        "signal_level": 4,
        "signal_strength": -51,
        "timing_advance": 0
      },
      "cellular_operator_name": "",
      "cellular_technology": "simless",
      "connection_type": "wi-fi",
      "sim_country_code": "234",
      "sim_network_code": "20",
      "sim_operator_name": ""
    },
    "wifi": {
      "ssid": "2021fcspeedtestapp",
      "strength": -74,
      "type": "auto"
    }
  }
}
"frequency": 2462,
"hidden_ssid": false,
"link_speed": 173,
"network_id": 9,
"rssi": -37,
"supplicant_state": "completed" }
},
"end": {
"cross_traffic": {
"rx_avg_bps": 27939,
"rx_peak_bps": 190818,
"tx_avg_bps": 153122,
"tx_peak_bps": 879401
},
"dhcp": {
"dns1": "192.168.1.254",
"dns2": "0.0.0.0",
"gateway": "192.168.1.254",
"ip_address": "192.168.1.99",
"lease_duration": 86400,
"netmask": "0.0.0.0",
"server_address": "192.168.1.254"
},
"location": {
"lat": 51.5081771,
"lon": 0.0191134,
"time": "2021-10-26T14:54:43+0000"
},
"memory": {
"app_used": 7,
"device_free": 1915,
"device_used": 3424
},
"network": {
"is_roaming": false,
"subtype_name": "",
"type": "mobile_wifi"
},
"telephony": {
"bandwidth": "",
"cellular_country_code": "234",
"cellular_network_code": "10",
"cell_signal": {
"absolute_rf_channel_number": 115,
"asu": 31,
"base_station_identity_code": 2,
"carrier_aggregation": false,
"cell_connection": 0,
"cell_identity_id": 13874,
"location_area_code": 21212,
"mobile_country_code": 234,
"mobile_network_code": 10,
"network_type": "GSM",
"signal_level": 4,
"signal_strength": -51,
"timing_advance": 0
},
"cellular_operator_name": ""
2021 FCC Speed Test App Technical Description

"cellular_technology": "simless",
"connection_type": "wi-fi",
"sim_country_code": "234",
"sim_network_code": "20",
"sim_operator_name": ""
},
"wifi": {
"frequency": 2462,
"hidden_ssid": false,
"link_speed": 173,
"network_id": 9,
"rssi": -37,
"supplicant_state": "completed"
}
},
"failures": 0,
"local_datetime": "2021-10-26T15:54:44+01:00",
"successes": 1,
"target": "sp2-vm-ashburn-us.samknows.com",
"thread_count": 3,
"throughput": 1155279,
"utc_datetime": "2021-10-26T14:54:44Z",
"warmup_bytes": 2857512,
"warmup_duration": 3000075
},
"submission_type": "scheduled_test",
"utc_date_time": "2021-10-26T14:54:45Z"
}

2. iOS Test Flow, Environmental Data, and JSON Format

In a scenario where all test types are switched to active on the app’s Test screen, upon pressing Start Tests the following sequence of actions takes place:

1. The app fetches the stackpath-us target set from the Speedtest API.
2. The server selection process (described above) is performed against each of the targets in the list.
3. The Download test is performed against the test server identified to have the lowest TCP connection time for the user’s current location, from the previous step. The result and a snapshot of the current environmental data (see later) is captured.
4. Pause of 2 seconds.
5. The Upload test is performed against the chosen test server. The result and a snapshot of the current environmental data (see later) is captured.
6. Pause of 2 seconds.
7. The latency/loss/jitter test is performed against the chosen test server. The result and a snapshot of the current environmental data (see later) is captured.
8. Pause of 2 seconds.
9. The results are submitted to the backend Data Ingestion API and a resulting Test ID is retrieved.
10. The results for the entire test sequence are paired with their respective Test ID and stored locally.

At the end of a test sequence, the following information is displayed to the user:

- Download speed
- Upload speed
- Latency
- Packet loss
- Jitter
- Date and time of measurement
- The Wi-Fi or cellular network that was tested
- The ISP that the user used
- The test server location

If any test fails to obtain a measurement for any reason (such as when there is no network connectivity at all), then “FAILED” is shown for the respective test result. If any other data was unavailable, due to a failure, then a "-" is shown in its place.

**Table 2: iOS Data Dictionary**

The table below shows all the data fields that are collected during the tests.

The iOS device’s JSON data export feature within the app generates different field names and data structures from the JSON submission that is sent to the backend. The field names that are sent to the backend are listed in the first column, whilst the field names used in the device JSON export are listed in the second column. Future versions of the app will unify these.

Please also note that some JSON field names may differ from the backend database. The complete backend field names are provided in the “2021 FCC Mobile App Data Dictionary (Raw Data) - Android and iOS.xlsx.”

<table>
<thead>
<tr>
<th>Common and environmental fields for iOS</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Backend submission JSON field name</strong></td>
<td><strong>Device JSON export field name</strong></td>
</tr>
<tr>
<td>panelId</td>
<td>-</td>
</tr>
<tr>
<td>anonymised</td>
<td>-</td>
</tr>
<tr>
<td>version</td>
<td>-</td>
</tr>
<tr>
<td>Key</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>data.tests.&lt;TEST&gt;.environment.location.lat</td>
<td>Latitude at the start of measurement.</td>
</tr>
<tr>
<td>data.tests.&lt;TEST&gt;.environment.location.lon</td>
<td>Longitude at the start of measurement.</td>
</tr>
<tr>
<td>data.tests.&lt;TEST&gt;.environment.location.accuracy.vertical</td>
<td>Vertical accuracy at the start of measurement.</td>
</tr>
<tr>
<td>data.tests.&lt;TEST&gt;.environment.location.accuracy.horizontal</td>
<td>Horizontal accuracy at the start of measurement.</td>
</tr>
<tr>
<td>data.tests.&lt;TEST&gt;.environment.location.address</td>
<td>Approximate postal address at the start of measurement.</td>
</tr>
<tr>
<td>data.tests.&lt;TEST&gt;.environment.memory.app_used</td>
<td>Memory used in MB by the app at the start of the test.</td>
</tr>
<tr>
<td>data.tests.&lt;TEST&gt;.environment.memory.device_free</td>
<td>Memory free on the device in MB at the start of the test.</td>
</tr>
<tr>
<td>data.tests.&lt;TEST&gt;.environment.memory.device_used</td>
<td>Memory used on the device in MB at the start of the test (not always available).</td>
</tr>
<tr>
<td>data.tests.&lt;TEST&gt;.environment.wifi.ssid</td>
<td>The SSID of the router that was used to perform the test if over WiFi.</td>
</tr>
<tr>
<td>data.tests.&lt;TEST&gt;.environment.network.subtype_name</td>
<td>The cellular radio technology in use at the start of the test: Enum (1xRTT, CDMA, EDGE, GPRS, GSM, IDEN, EHRPD, EVDO-0, EVDO-A, EVDO-B, HSDPA, HSPA, HSPAP, HSUPA, IWLAN, TD-SCDMA, UMTS, LTE, UNKNOWN, NR).</td>
</tr>
<tr>
<td>data.tests.&lt;TEST&gt;.environment.telephony.cellular_technology</td>
<td>The generation of cellular radio technology (Enum 2g, 3g, 4g, 5g, simless).</td>
</tr>
<tr>
<td>data.tests.&lt;TEST&gt;.environment.telephony.connection_type</td>
<td>Primary field indicating whether the user is on cellular or WiFi (Enum: cell, wi-fi).</td>
</tr>
<tr>
<td>data.tests.&lt;TEST&gt;.environment.other.is_logged_in</td>
<td>Boolean indicating if the user is logged in to the app at the start of the test.</td>
</tr>
<tr>
<td>data.device_environment.id</td>
<td>Platform-specific unique identifier for the device. Generated randomly upon every test for the FCC (so it’s not a persistent unique identifier for the device).</td>
</tr>
<tr>
<td>data.device_environment.mobile_network_code</td>
<td>Mobile network code for the SIM card in the device, in its numeric representation. If multiple SIMs are present, this will be the active one.</td>
</tr>
<tr>
<td>Backend submission JSON field name</td>
<td>Device JSON export field name</td>
</tr>
<tr>
<td>-----------------------------------</td>
<td>------------------------------</td>
</tr>
<tr>
<td>data.tests.&lt;TEST&gt;.utc_datetime</td>
<td>tests.&lt;TEST&gt;.utc_datetime</td>
</tr>
<tr>
<td>data.tests.&lt;TEST&gt;.local_datetime</td>
<td>tests.&lt;TEST&gt;.local_datetime</td>
</tr>
<tr>
<td>data.tests.&lt;TEST&gt;.successes</td>
<td>tests.&lt;TEST&gt;.success</td>
</tr>
<tr>
<td>data.tests.&lt;TEST&gt;.failures</td>
<td>-</td>
</tr>
<tr>
<td>data.tests.&lt;TEST&gt;.duration</td>
<td>-</td>
</tr>
<tr>
<td>data.tests.&lt;TEST&gt;.target</td>
<td>-</td>
</tr>
<tr>
<td>data.tests.&lt;TEST&gt;.thread_count</td>
<td>-</td>
</tr>
<tr>
<td>data.tests.&lt;TEST&gt;.warmup_duration</td>
<td>-</td>
</tr>
<tr>
<td>data.tests.&lt;TEST&gt;.warmup_bytes</td>
<td>-</td>
</tr>
<tr>
<td>data.tests.&lt;TEST&gt;.bytes_transferred</td>
<td>-</td>
</tr>
<tr>
<td>data.tests.&lt;TEST&gt;.throughput</td>
<td>tests.&lt;TEST&gt;.testResult</td>
</tr>
</tbody>
</table>
### Backend submission JSON field name

<table>
<thead>
<tr>
<th>Device JSON export field name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>data.tests.&lt;TEST&gt;.utc_datetime</td>
<td>Date time of the locale time-zone in the format &quot;2021-10-26T14:54:12+01:00&quot;.</td>
</tr>
<tr>
<td>data.tests.&lt;TEST&gt;.local_datetime</td>
<td>Date time of the locale time-zone in the format &quot;2021-10-26T15:54:12+01:00&quot;.</td>
</tr>
<tr>
<td>data.tests.&lt;TEST&gt;.successes</td>
<td>Number of successes (always 1 or 0).</td>
</tr>
<tr>
<td>data.tests.&lt;TEST&gt;.failures</td>
<td>Number of failures (always 1 or 0).</td>
</tr>
<tr>
<td>data.tests.&lt;TEST&gt;.duration</td>
<td>Duration of the test, in microseconds.</td>
</tr>
<tr>
<td>data.tests.&lt;TEST&gt;.target</td>
<td>The test server in use.</td>
</tr>
<tr>
<td>data.tests.&lt;TEST&gt;.packet_size</td>
<td>Size of packets used in the latency test (bytes).</td>
</tr>
<tr>
<td>data.tests.&lt;TEST&gt;.stream_bits_per_second</td>
<td>The measured speed, in bits per second.</td>
</tr>
<tr>
<td>data.tests.&lt;TEST&gt;.packets_sent</td>
<td>Number of packets sent during latency test.</td>
</tr>
<tr>
<td>data.tests.&lt;TEST&gt;.packets_received</td>
<td>Number of packets received during latency test.</td>
</tr>
<tr>
<td>data.tests.&lt;TEST&gt;.round_trip_time</td>
<td>Round trip time in the latency test.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------------------</td>
<td>----------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>data.tests.&lt;TEST&gt;.jitter</td>
<td>Calculated variation in latency between packets flowing between device and server in the latency test.</td>
</tr>
<tr>
<td>tests.Latency.Jitter.testResult</td>
<td>In the backend submission JSON this is reported in microseconds, whilst in the device export JSON it is reported in milliseconds.</td>
</tr>
<tr>
<td>data.tests.&lt;TEST&gt;.status_code</td>
<td>Internal code to denote the resulting status of the latency test. Detafs to 0 meaning NO_ERROR.</td>
</tr>
<tr>
<td>data.tests.&lt;TEST&gt;.mean_opinion_score</td>
<td>An estimate of perceived connection quality, factoring-in aspects such as latency, packet loss and jitter.</td>
</tr>
<tr>
<td>-</td>
<td>tests.Latency.PacketLoss.testResult</td>
</tr>
<tr>
<td>-</td>
<td>Packet loss percentage.</td>
</tr>
<tr>
<td>-</td>
<td>tests.&lt;TEST&gt;.testProvider</td>
</tr>
<tr>
<td>-</td>
<td>A lookup of the public IP address against the Maxmind database. Never submitted with the JSON submission. Same as the maxmind_isp value in the backend database. Obtained on the device using the <a href="https://speedtest-api.samknows.com/isp">https://speedtest-api.samknows.com/isp</a> api call.</td>
</tr>
<tr>
<td>-</td>
<td>tests.&lt;TEST&gt;.isWiFi</td>
</tr>
<tr>
<td>-</td>
<td>Boolean value which indicates if the test was performed over Wifi.</td>
</tr>
<tr>
<td>-</td>
<td>tests.&lt;TEST&gt;.testNetwork</td>
</tr>
<tr>
<td>-</td>
<td>The mobile network for the device’s active SIM card for data (e.g. “vodafone UK (4G)“). If on WiFi, this value will contain the the WiFi SSID.</td>
</tr>
<tr>
<td>-</td>
<td>tests.&lt;TEST&gt;.referenceNumber</td>
</tr>
<tr>
<td>-</td>
<td>The testID for this test sequence.</td>
</tr>
<tr>
<td>-</td>
<td>tests.&lt;TEST&gt;.timestamp</td>
</tr>
<tr>
<td>-</td>
<td>The numerical timestamp of the test time (e.g. 1634661555.4815311).</td>
</tr>
<tr>
<td>-</td>
<td>tests.&lt;TEST&gt;.testServer</td>
</tr>
<tr>
<td>-</td>
<td>The location name of the test server.</td>
</tr>
<tr>
<td>-</td>
<td>tests.&lt;TEST&gt;.testIndex</td>
</tr>
<tr>
<td>-</td>
<td>The index of the test.</td>
</tr>
</tbody>
</table>
JSON Reporting Format Example for iOS

Results are serialized into the following JSON format when transmitting them to the backend database.

```json
{
  "anonymised": true,
  "panelId": 253,
  "version": 1,
  "data": {
    "device_environment": {
      "mobile_network_code": "15",
      "mobile_country_code": "234",
      "operating_system_version": "iOS 14.4",
      "model": "iPhone13,3",
      "iso_country_code": "gb",
      "manufacturer": "Apple",
      "id": "94F90F11-B7C0-4EC6-BC30-C5601B970E64",
      "carrier_name": "vodafone UK"
    },
    "metadata": {
      "app_version": "v4.1.4 b3"
    },
    "tests": {
      "latency": {
        "packets_received": 200,
        "utc_datetime": "2021-03-05T13:22:46Z",
        "target": "sp1-vm-chicago-us.samknows.com",
        "local_datetime": "2021-03-05T13:22:46Z",
        "duration": 5000355,
        "environment": {
          "other": {
            "is_logged_in": false
          },
          "network": {
            "subtype_name": "LTE"
          },
          "telephony": {
            "connection_type": "wi-fi",
            "cellular_technology": "4g"
          },
          "memory": {
            "device_used": 0,
            "app_used": 144,
            "device_free": 399277
          },
          "wifi": {
            "ssid": ""
          },
          "location": {
            "lat": 51.495334211107462,
            "address": "Building XX, YY Road, ZZ11 1XY",
            "accuracy": {
              "horizontal": 65,
              "vertical": 10
            },
            "lon": 0.073787086332168406
          }
        }
      }
    }
  }
}
```
"status_code": 0,
"stream_bits_per_sec": 51200,
"failures": 0,
"packet_size": 160,
"packets_sent": 200,
"jitter": 932,
"mean_opinion_score": 1,
"round_trip_time": 93911,
"successes": 1
},
"download": {
"warmup_bytes": 9460368,
"warmup_duration": 3000181,
"bytes_transferred": 19393056,
"throughput": 3878241,
"successes": 1,
"failures": 0,
"utc_datetime": "2021-03-05T13:22:32Z",
"environment": {
"telephony": {
"cellular_technology": "4g",
"connection_type": "wi-fi"
},
"location": {
"accuracy": {
"vertical": 10,
"horizontal": 65
},
"lat": 51.495334211107462,
"lon": 0.073787086332168406,
"address": "Building XX, YY Road, ZZ11 1XY"
},
"memory": {
"device_used": 0,
"app_used": 144,
"device_free": 399278
},
"other": {
"is_logged_in": false
},
"network": {
"subtype_name": "LTE"
},
"wifi": {
"ssid": ""
}
},
"thread_count": 3,
"duration": 5000476,
"local_datetime": "2021-03-05T13:22:32Z",
"target": "sp1-vm-chicago-us.samknows.com"
},
"upload": {
"target": "sp1-vm-chicago-us.samknows.com",
"thread_count": 3,
"bytes_transferred": 7045920,
"warmup_duration": 3000042,
"environment": {
"network": {
"subtype_name": "LTE"
}
},
"thread_count": 3,
"duration": 5000476,
"local_datetime": "2021-03-05T13:22:32Z",
"target": "sp1-vm-chicago-us.samknows.com"
Results are serialized into the following JSON format when exporting them using the device’s local data export functionality:

```
{
  "tests": {
    "Upload": {
      "local_datetime": "2021-09-21T12:06:47+01:00",
      "referenceNumber": "3046270262",
      "testIndex": 0,
      "latitude": 51.495279410522393,
      "testServer": "Ashburn, US",
      "timestamp": 1632222413.0320759,
      "testProvider": "Vodafone UK",
      "testNetwork": "HomeHub",
      "longitude": 0.073730860056579903,
      "utc_datetime": "2021-09-21T11:06:47Z",
      "testResult": 15.746728721313875,
    },
    "failures": 0,
    "duration": 5000047,
    "throughput": 1409170,
    "warmup_bytes": 6346920
  }
}```

**JSON Local Data Export Example for iOS**

Results are serialized into the following JSON format when exporting them using the device’s local data export functionality:

```
{
  "tests": {
    "Upload": {
      "local_datetime": "2021-03-05T13:22:40Z",
      "referenceNumber": "3046270262",
      "testIndex": 0,
      "latitude": 51.495279410522393,
      "testServer": "Ashburn, US",
      "timestamp": 1632222413.0320759,
      "testProvider": "Vodafone UK",
      "testNetwork": "HomeHub",
      "longitude": 0.073730860056579903,
      "utc_datetime": "2021-03-05T13:22:40Z",
      "testResult": 15.746728721313875,
    },
    "failures": 0,
    "duration": 5000047,
    "throughput": 1409170,
    "warmup_bytes": 6346920
  }
}```
"isWiFi" : true,
"success" : true
},
"Download" : {
  "testProvider" : "Vodafone UK",
  "testIndex" : 0,
  "utc_datetime" : "2021-09-21T11:06:39Z",
  "local_datetime" : "2021-09-21T12:06:39+01:00",
  "testServer" : "Ashburn, US",
  "latitude" : 51.495279410522393,
  "success" : true,
  "longitude" : 0.073730860056579903,
  "timestamp" : 1632222413.032166,
  "testResult" : 70.013874046316829,
  "testNetwork" : "HomeHub",
  "isWiFi" : true,
  "referenceNumber" : "3046270262"
},
"Latency" : {
  "timestamp" : 1632222413.0321879,
  "referenceNumber" : "3046270262",
  "testProvider" : "Vodafone UK",
  "testNetwork" : "HomeHub",
  "testResult" : 83.165214285714285,
  "latitude" : 51.495279410522393,
  "success" : true,
  "PacketLoss" : {
    "latitude" : 51.495279410522393,
    "testServer" : "Ashburn, US",
    "testProvider" : "Vodafone UK",
    "testResult" : 0,
    "isWiFi" : true,
    "timestamp" : 1632222413.0322151,
    "longitude" : 0.073730860056579903,
    "referenceNumber" : "3046270262",
    "local_datetime" : "2021-09-21T12:06:52+01:00",
    "utc_datetime" : "2021-09-21T11:06:52Z",
    "testNetwork" : "HomeHub",
    "success" : true,
    "testIndex" : 0
  },
  "testServer" : "Ashburn, US",
  "longitude" : 0.073730860056579903,
  "local_datetime" : "2021-09-21T12:06:52+01:00",
  "utc_datetime" : "2021-09-21T11:06:52Z",
  "isWiFi" : true,
  "testIndex" : 0,
  "Jitter" : {
    "local_datetime" : "2021-09-21T12:06:52+01:00",
    "testIndex" : 0,
    "latitude" : 51.495279410522393,
    "isWiFi" : true,
    "longitude" : 0.073730860056579903,
    "success" : true,
    "utc_datetime" : "2021-09-21T11:06:52Z",
    "testProvider" : "Vodafone UK",
    "referenceNumber" : "3046270262",
    "testResult" : 1.4590000000000001,
    "testServer" : "Ashburn, US",
    "timestamp" : 1632222413.032203,
    "testNetwork" : "HomeHub"
  }
},
"success" : true,
"isWiFi" : true,
"testIndex" : 0,
"referenceNumber" : "3046270262"
III. FCC Speed Test App Privacy Notice and Terms of Use (updated May 13, 2021)

A. What Information We Collect

The information the FCC Speed Test App ("App") collects is limited to information used to measure your mobile (including WiFi-enabled) broadband service. We do not collect personally identifiable information, such as name, phone number, location, or identifiers associated with your device, without your permission. We collect five kinds of information to measure the performance of your mobile broadband service.

1. Location (with your permission)

If enabled, the App will report your network name and your physical location when using the App.

- When using the App, you will be prompted to select whether the App may access location information only when using the App, always, or never.

- You do not need to grant access to location information to use the App, but you will be unable to view a map of your historical tests or have your results included in a map created by the FCC to study mobile broadband access if you do not.

- Without the location data, the measurement results will be of limited usefulness to the FCC in its studies of mobile broadband service. When possible, we will aggregate your location data with other users' location information to provide aggregated network performance and coverage statistics.

---

10 For the most recent update, see https://www.fcc.gov/general/mobile-broadband-performance-application-privacy-notice-and-terms-use.
2. Time of Data Collection

The App will report the date and time when it performs measurements of your broadband performance. Date and time are important metrics to capture because network loading, which affects performance, can vary across times of the day and/or days of the week. Time zone is an important metric to capture as it allows the comparison of measurements against known outages.

3. Device Type and Operating System Version

The App reports the manufacturer and model of the device you are using, as well as the version of the operating system installed on your device.


The App will report information about your mobile device connection performance, such as your service provider, the strength and quality of the radio signal, and the type of connection.

5. Broadband Performance

When not connected to WiFi, the App tests and reports your mobile broadband upload and download rates, round trip latency, and packet loss measured between your device and the servers managed by the FCC’s contractor and other project partners.

If you are connected to WiFi, the App will test and report the performance of your connection through your WiFi network, which performance will depend upon, but is not limited to, your WiFi configuration, your mobile device operating system and performance, the exact location of the mobile device (including the proximity of the device to the WiFi access point), the number of devices sharing your WiFi bandwidth, and the performance of your fixed internet service. This data is stored on your device for you to review, and periodically is uploaded to our servers automatically. You can find more detail in the FCC’s technical summary of this program at https://www.fcc.gov/measuring-broadband-america/mobile/technical-summary.html.
B. Data Sharing

The data the App reports is helpful for consumers, the FCC, and others interested in understanding broadband in the United States. We share broadband performance information collected through the App in four ways.

1. Internally

We may share your information within the FCC to write various reports on mobile broadband performance and to support our development of broadband policy.

2. Third-Party Partners Who Help Us Run Our Test and Support Our Program

We share your mobile broadband performance information with our contractor who manages the measurement program, with partners who support the program and architecture, and with partners supporting our analysis or contributing peer reviews. We share data subject to agreements that include strict requirements to safeguard the data, not disclose the data, and use data only in furtherance of the goals of the program.

The FCC has contracted with a broadband measurement company, SamKnows, to help manage the program and conduct its technical functions. SamKnows performs functions such as writing and maintaining the software that performs the test measurements, responding to customer service requests, managing the App and the execution of tests, ensuring the integrity of and protecting the data, and providing analysis and support for the preparation of FCC reports. The terms of our contract with SamKnows include obligations to protect the data and your privacy.

Research platforms and your carrier may support various aspects of the testing and architecture for the program. We may share your detailed mobile performance data with these partners supporting our data processing and analysis for the specific purpose of identifying patterns in the data and recommending ways of addressing any concerns for your privacy and anonymity. The FCC directs researchers, its contractor, and your carrier working under non-disclosure agreements (NDA) to analyze the collected data for privacy risks or concerns and to plan for future measurement efforts.
3. Pursuant to Legitimate Requests by Law Enforcement or Where Otherwise Required by Law or Regulation

We may share broadband performance data collected by this App subject to legitimate, written requests by law enforcement or where otherwise required by law or regulation.

4. De-Identified Data Available to the Public

We will publish de-identified data in summary statistics, coarsened data, the creation of maps, and other minimally necessary datasets that support our review of the statistics only if we determine that the data poses very low risks to your privacy. These include:

a. **Summary Data**
   We will publish averages and other summary statistics that describe important features of broadband performance. These summary statistics pose very low risks to your privacy.

b. **Coarsened Data**
   We will publish only de-identified data in a coarsened form that supports the averages and other statistics that we publish in our reports. Prior to release of this data, the FCC will consider the likelihood that the data could be combined with other datasets to identify individual users of the App and make adjustments to minimize the risk of identification.

c. **Creation of Maps**
   The FCC will use de-identified data to create maps that reflect the general location, time, and results of the tests to inform the public on the availability of broadband and the FCC in its efforts to improve broadband mapping and access to broadband. If possible, your test results will be aggregated with the results of other tests conducted nearby to minimize the risk of identification.

C. Data Consumption

By default, the App is configured to use no more than 100 megabytes (MB) of data each month for automated testing. Any additional tests you initiate will also consume between 100 kilobytes (kB) for the smallest packet loss test and 40 MB for a complete suite of tests. At any time, you can change the amount of data the App uses by increasing or reducing the “monthly data cap” in the Settings menu. You should ensure that your monthly broadband usage, including the data used by the App, does not exceed the data allowance included in your mobile broadband package. Your carrier may charge you for any data used in excess of your package’s data allowance.
D. Stopping Your Participation

If at any time you want to stop participating in this program, simply delete the App, and the App will not collect any additional data. Any data collected prior to the deletion of the App will remain in the data sets subject to the above terms and protections.

E. Other Terms of Use of the Application

- GNU Licensing Agreement (http://www.gnu.org/licenses/gpl-2.0.html)

Mobile broadband providers endorse some applications certified for use on their network. This application may not have been formally certified by your provider or carry a formal endorsement from your mobile broadband network.