NetMap - Network Performance Measurements

Victor Costan, Tiffany Yu-Han Chen, Ravi Netravali
Computer Science and Artificial Intelligence Laboratory
Massachusetts Institute of Technology, Cambridge, Mass., USA
{costan,yuhan,ravinet}@mit.edu

ABSTRACT

1. ABSTRACT

We introduce a platform for measuring wireless Internet access performance across the world's cellular and WiFi networks. Our platform consists of a library for collecting network performance measurements on Android devices, a pipeline for storing and processing the performance measurements, and tools and sample code for accelerating the development of location-based mobile games that contribute peformance measurements to our project. We plan to develop one such game, and release it for free on the Android market. Players pay for the by game by implicitly allowing us to use their phone to measure network performance. We also opened up and documented our platform, hoping to influence other (more talented) game developers to build games on top of our platform.

2. INTRODUCTION

Recent years have seen an immense growth in the number of wireless devices in use. Functionality of these devices has similarly evolved as one can find a mobile application for just about any task. In turn, the need for efficient and widespread wireless network coverage is at a high. However, today, customers are often susceptible to poor network coverage over both WiFi and cellular networks. Application developers find it difficult to understand how to optimize their application performance for different devices over wireless networks. Researchers struggle to find data to test potential improvements in wireless network and device performance. This arises from a lack of information about the behavior and performance of wireless networks for different devices, locations, and times.

This work describes the design, implementation, and experimental evaluation of NetMap, a system that provides a battery-aware network performance measurement collection service with a simple API for mobile devices. Our main contributions are a measurement collection API designed for games as well as a packaging mechanism to collect performance data in a mobile device game.

Our API provides a means to collect various types of data including user-specific data such as device model and location (GPS), network performance data including latency, bandwidth, and average round-trip-time, and neighboring network infrastructure information such as neighboring network type and signal strength to neighboring towers/access points. It is specifically designed for mobile device games. Game development is straightforward, as designers are not required to know how our measurements are collected, processed, stored, etc. Game designers simply use functions in our API to collect the requested data.

We developed a scheme which aligns incentives so game players collect measurements while playing. That is, we tie measurement collection at real-world locations with movement and action in the mobile device game. Ideally, the provided measurement library would be called upon when a game player moves (reflecting movement in real-life and consequently movement in-game). Measurement collection can thus be widespread as users provide measurements from various locations while playing their mobile device game.

Users of these mobile games are aware that they will be collecting information on network connectivity that will be used for research purposes. However, beyond providing permission, gameplay is the users sole focus, as no additional effort must be expended to gather, store, or analyze the network performance measurements. We are also battery-aware so battery-intensive data collection and transfers are done only when battery permits reducing user overhead.

Targeted results of using NetMap and a mobile device game are a collection of various network performance statistics such as bandwidth, latency, average round-trip-time, etc., user statistics such as phone models and battery, and a comprehensive map of how different wireless networks behave across various regions. From these results, researchers and game developers can infer cellular network and WiFi performance for different areas, devices, and times. Additionally, this will provide a reliable and up-to-date source of data for researchers to use in developing improvements for wireless and mobile device connectivity.

The remainder of this paper is organized as followsK

3. RELATED WORK

Network Diagnostic Tool (NDT) [1] collects performance data over wireless links using a client/server architecture. The server consists of a webserver and an analysis engine. The client communicates with this enhanced server to per-

form diagnostic tests including web page request and the server collects the resulting measurements and attempts to identify the cause of performance issues. The primary goal of ndt is to identify network performance issues, which occur close to users (eg. incorrectly set TCP buffers). The server locations are all known (clients connect to one of the closest servers) and servers collect data making it easier to measure certain statistics such as one-way latency.

Dasu [4] is a measurement platform for the Internet's edge. Dasu can support broadband characterization as well as internet measurement experiements. They design Dasu for the edge of the Internet so measurements reflect end users' views of the services they are using. Dasu also does not use dedicated infrastructures for experimentation. Instead, they use an incentive model to make sure it is widely adopted at the edge of the Internet. Dasu has a distributed set of clients and a set of management services. Clients perform measurements and the management services configure clients, perform administration of experiments, and handle data collection. Dasu provides a programming interface that is flexible to run many kinds of tests (when-then model where condition dictates type of test).

MIST (mobile internet services test), a distributed platform for measuring cellular network performance of users with hopes of aiding mobile application developers. MIST is a mobile app connected to server back-end. Communication between the mobile application on the user's device and the servers are performed to measure characeteristics of the cellular networks, including latency, jitter, throughput, etc. The database at the server saves the measurement data along with mobile device info/configuration from the test. Perk is that MIST can be deployed on top of mobile devices (don't have to change cell network infrastructure). App first collects info about mobile device, service provider, and test location. Mobile app connects to closest server to get most accurate measurements. Then app sends packets of set byte-size to analyze uplink and downlink latency, throughput, and timeouts. Difference from ours is that MIST is an app designed to get such measurements for mobile app developers (we wrap measurement collection in a game so it is not specifically used for this purpose).

Balachandran et al. [2] capture a workload at a large conference and analyze it to understand user behavior and network performance. They collect a continuous trace of SNMP data from all APs in the conference main room as well as a tcpdump trace of network-level headers of packets going through switch which all APs connected to. This provided aggregate packet level statistics of all traffic passing through these APs at the link, network and transport layers. Also, they obtained information about the users associated with the APs such as their MAC addresses, SNR, and effective throughput. They inferred the number of distinct wireless users by counting the number of distinct MAC addresses in packets passing through the APs present. The primary goal was to analyze user behavior in terms of mobility, application popularity, data rates, etc. In terms of network performance, they measure the aggregate offered load for each of the APs and observe the bursty behavior. They also measure packet errors by using the SNMP trace where APs count the total number of packets transmitted and received, and the number of packets in error (account for inbound packets that could not be delivered to higher layer and outbound packets that cant be transmitted due to channel).

VISUM [3] is a framework for wireless network monitoring that uses set of agents within network (scales better than centralized) to monitor network devices and store info at repositories. VISUM also visualizes the data into real-time statistical graphs and interactive network topology maps. They target single-hop wireless networks. Thus, VISUM relies on a distributed architecture (agents at different locations) to monitor large scale wireless networks. Agents collect measurement info from network devices using SNMP and store the data in a centralized repository (data stored per device using device OID).

4. DESIGN OVERVIEW

4.1 The Network Performance Measurement Library

We measure more than 200 types of network data, including device-specific data such as device model, location data, network performance data including latency, bandwidth, and average round-trip-time, neighboring network infrastructure information such as neighboring network type and signal strength to neighboring towers/access points, and the DHCP information. All the data is stored in JSON format, so that researchers can easily parse it.

Device-Specific Information. NetMap collects device-specific information including the device ID (IMEI or ESN), the device type (GSM or CDMA), the software version. With the device ID, we can identify a device, track the device, and discard bad data that comes due to cheating. We can even provide personalized network usage diaries which allow users to better understand their network using habits. The OS version allows us to infer the effect of software on the network performance. NetMap also logs SIM card information, such as the phone number, the SIM card operator (AT&T, T-Mobile, etc.), the radio type (EDGE, GPRS, HSDPA, etc.). This permits us to create the map of network quality for each phone carrier or type, and how the network quality evolves over time.

Location. The system gathers the user's location information from the GPS or network location provider. For the GPS, in addition to the latitude and longitude, our system also collects information about the observed satellites, including the almanac, ephemeris, azimuth, elevation, and SNR. We plan to use the stallites information to detect users that cheat the game by faking their GPS locations. Because GPS is unavailable indoors, we also use the network location provider's locations.

Network Infrastructure Information. NetMap collects information about neighboring WiFi APs and cell towers.

The cell tower information includes the mobile country code (MCC), mobile network code (MNC), location area code (LAC), and the signal strength; the WiFi AP information includes the MAC address, SSID, IP address, frequency band, RSSI, link speed, etc. With the locations where users observed the cell tower/AP, we can approximate the location of each cell tower/AP. Many research projects [6, 5] have focused on using energy efficient sensors to provide accurate position estimation, but most of them suffer from scalability problem due to lack of cell tower/AP observations. We belive we can greatly assist those projects with the crowdsourced data that we have. However, neighboring GSM cell tower information isn't universally available across all phones. Some phone models limit the cell information to only the connected tower or do not make it available at all.

NetMap also collects the DHCP information, including the assigned IP, network mask, IP of the DHCP server, etc. DHCP information is useful for inferring the network topology, and for cheating detection.

Network Performance. We collect sophisticated network performance measurements between the device and servers, such as the network latency, bandwidth, average round-trip-time, and some TCP variables. These measurements are crucial for both network researchers and normal users. For researchers, they can use the information for research on improving network reliability and performance, as it will provide data on data transport latency and throughput for cellular and WiFi networks. For normal users, these measurements help them choose the network carrier which provides the best reception in their neighborhood.

We use the Network Diagnostic Tool (NDT) [1] to collect network quality information. NDT measures various network performance metrics between the mobile device and their distributed servers. There are some straightforward and incredibly tedious problems in measuring the network performance. First, one needs to consider the time synchronization problem between the device and the server; second, to eliminate the noise in the measurement, one needs to maintain servers in multiple places. NDT solves the time synchronization problem, and it maintains 81 servers in 27 coutries. This allows NetMap to provide accuracy network performance measurements across the world.

4.2 Battery/Network Awareness

There are two important aspects that designers need to take into account when implementing a mobile programming API. First, the API should be energy-efficient. Games using the API should not drain the battery of the phone quickly; second, the API should not drain the user's data plan or make the user pay extra money.

NetMap addresses these two problems by incorporating the concept of battery and network aware. To be battery aware, NetMap monitors the battery level and status. Games using the API would get notified when the phone is charging or when the battery level is low, and take certain actions to save energy. For example, if the phone is charging, the game can start uploading the measurements it collected to the server or taking measurements more frequently.

NetMap is also network aware. NetMap monitors the network type that the device is currently using, and notifies the games that using the API if the user switches to either WiFi or cellular network. Games can react to the notification by uploading all the collected measurements to the server when the phone is using WiFi.

5. REFERENCES

[1] NDT. http://www.internet2.edu/performance/ndt/.

[2] A. Balachandran, G. M. Voelker, P. Bahl, and P. V. Rangan. Characterizing user behavior and network performance in a public wireless lan. In *Proceedings of the 2002 ACM SIGMETRICS international conference on Measurement and modeling of computer systems*, SIGMETRICS '02, pages 195–205, New York, NY, USA, 2002. ACM.

[3] C. C. Ho, K. N. Ramachandran, K. C. Almeroth, and E. M. Belding-Royer. A scalable framework for wireless network monitoring. In *Proceedings of the 2nd ACM international workshop on Wireless mobile applications and services on WLAN hotspots*, WMASH '04, pages 93–101, New York, NY, USA, 2004. ACM.

[4] M. A. Sánchez, J. S. Otto, Z. S. Bischof, D. R. Choffnes, F. E. Bustamante, B. Krishnamurthy, and W. Willinger. Dasu: Pushing experiments to the Internet's edge. In *Proc. of USENIX NSDI*, April 2013.

[5] A. Thiagarajan, L. Ravindranath, K. LaCurts, S. Madden, H. Balakrishnan, S. Toledo, and J. Eriksson. VTrack: Accurate, Energy-Aware Road Traffic Delay Estimation Using Mobile Phones. SenSys '09.

[6] A. Thiagarajan, L. S. Ravindranath, H. Balakrishnan, S. Madden, and L. Girod. Accurate, Low-Energy Trajectory Mapping for Mobile Devices. NSDI '11.

6. APPENDIX

6.1 An Example of the Measurement


```
{
    "serial": 87,
    "app_id": 1071359345,
    "app_uid": "4842233025919687312",
    "created_at": "2013-05-14T00:41:44.993Z",
    "ip": "18.111.84.182",
    "data": {
        "uid": "1071359345.4842233025919687312.puumX9DcBa6UbfTfBdI6g4fSKew_h_FUh7gEgzrUGBM",
        "timestamp": 1368492034049,
        "location": {
            "latitude": 42.3587538,
            "longitude": -71.0949983,
            "provider": "network",
            "timestamp": 1368491627298,
            "accuracy": 31.571
        },
        "battery": {
            "status": "discharging",
            "plugged": "unknown",
            "charge": {
                "level": 91,
                "scale": 100
            },
            "health": "unknown"
        },
        "cellular": {
            "phoneId": "353918055485722",
            "line1Number": "16172309694",
            "networkOperator": "310260",
            "networkOperatorName": "T-Mobile",
            "simOperator": "310260",
            "simOperatorName": "",
            "softwareVersion": "03",
            "networkCountryISO": "us",
            "simCountryIso": "us",
            "simSerialNumber": "8901260562590079971",
            "isRoaming": false,
            "callState": 0,
            "subscriberId": "310260569007997",
            "phoneType": "lgsm",
            "radioType": "edge",
            "dataActivity": "none",
            "dataState": "disconnected",
            "simState": "ready"
        },
        "gps": {
            "enabled": true,
            "started": true,
            "timeToFix": 0,
```

```
"satellites": [
    {
        "prn": 1,
        "used": false,
        "almanac": true,
        "ephemeris": true,
        "azimuth": 104,
        "elevation": 67,
        "snr": 0
    },
    {
        "prn": 7,
        "used": false,
        "almanac": true,
        "ephemeris": true,
        "azimuth": 187,
        "elevation": 28,
        "snr": 0
    },
    {
        "prn": 8,
        "used": false,
        "almanac": true,
        "ephemeris": true,
        "azimuth": 216,
        "elevation": 53,
        "snr": 0
    },
    {
        "prn": 9,
        "used": false,
        "almanac": true,
        "ephemeris": true,
        "azimuth": 286,
        "elevation": 43,
        "snr": 0
    },
    {
        "prn": 11,
        "used": false,
        "almanac": true,
        "ephemeris": true,
        "azimuth": 63,
        "elevation": 54,
        "snr": 0
    },
    {
        "prn": 17,
        "used": false,
        "almanac": true,
        "ephemeris": true,
        "azimuth": 261,
        "elevation": 29,
        "snr": 0
    },
    {
        "prn": 19,
        "used": false,
        "almanac": true,
        "ephemeris": true,
        "azimuth": 63,
        "elevation": 16,
        "snr": 0
    },
    {
        "prn": 20,
        "used": false,
        "almanac": true,
        "ephemeris": true,
        "azimuth": 140,
        "elevation": 10,
        "snr": 0
    },
    {
        "prn": 26,
        "used": false,
        "almanac": true,
        "ephemeris": true,
        "azimuth": 277,
        "elevation": 6,
        "snr": 0
    },
    {
        "prn": 28,
        "used": false,
        "almanac": true,
        "ephemeris": true,
        "azimuth": 316,
        "elevation": 60,
        "snr": 0
    },
    {
        "prn": 32,
        "used": false,
        "almanac": true,
        "ephemeris": true,
        "azimuth": 112,
        "elevation": 11,
        "snr": 0
    },
    {
        "prn": 70,
        "used": false,
        "almanac": false,
        "ephemeris": false,
        "azimuth": 21,
        "elevation": 7,
        "snr": 0
    },
    {
        "prn": 71,
        "used": false,
        "almanac": true,
        "ephemeris": true,
        "azimuth": 68,
        "elevation": 12,
        "snr": 0
    },
    {
        "prn": 72,
        "used": false,
        "almanac": true,
        "ephemeris": true,
        "azimuth": 115,
        "elevation": 1,
        "snr": 0
    },
    {
        "prn": 76,
        "used": false,
        "almanac": false,
        "ephemeris": false,
        "azimuth": 168,
        "elevation": 27,
        "snr": 0
    },
    {
        "prn": 77,
        "used": false,
        "almanac": false,
        "ephemeris": false,
        "azimuth": 239,
        "elevation": 71,
        "snr": 0
    },
    {
        "prn": 78,
        "used": false,
        "almanac": false,
        "ephemeris": false,
        "azimuth": 323,
        "elevation": 35,
        "snr": 0
    },
    {
        "prn": 86,
        "used": false,
        "almanac": false,
        "ephemeris": false,
        "azimuth": 45,
        "elevation": 26,
        "snr": 0
    },
    {
        "prn": 87,
        "used": false,
        "almanac": false,
        "ephemeris": false,
        "azimuth": 337,
        "elevation": 68,
        "snr": 0
    },
    {
        "prn": 88,
        "used": false,
        "almanac": false,
        "ephemeris": false,
        "azimuth": 254,
        "elevation": 34,
        "snr": 0
    }
]
},
"wifi": {
    "enabled": true,
    "connection": {
        "ssid": "\"MIT\"",
        "hidden": false,
        "bssid": "00:21:d8:49:a3:8d",
        "mac": "10:68:3f:45:82:2b",
        "rssi": "-68",
        "linkMbps": 39,
        "state": "COMPLETED",
        "ip": "18.111.84.182"
    },
    "dhcp": {
        "ip": "18.111.84.182",
        "netmask": "255.255.224.0",
        "gateway": "18.111.64.1",
        "dhcpServer": "18.7.50.68",
        "dns1": "18.71.0.151",
        "dns2": "18.70.0.160",
        "lease": 86400
    },
    "aps": [
        {
            "ssid": "MIT",
            "bssid": "00:21:d8:49:a3:8d",
            "channelMhz": 5805,
            "signalDb": -57,
            "timestamp": 1368492027217,
            "capabilities": "[ESS]"
        },
        {
            "ssid": "MIT SECURE N",
            "bssid": "00:21:d8:49:a3:8b",
            "channelMhz": 5805,
            "signalDb": -56,
            "timestamp": 1368492027217,
            "capabilities": "[WPA2-EAP-CCMP][ESS]"
```

```
},
{
    "ssid": "MIT SECURE",
    "bssid": "00:21:d8:49:a3:8c",
    "channelMhz": 5805,
    "signalDb": -58,
    "timestamp": 1368492027217,
    "capabilities": "[WPA2-EAP-CCMP][ESS]"
},
{
    "ssid": "MIT SECURE",
    "bssid": "00:21:d8:4a:04:2c",
    "channelMhz": 5180,
    "signalDb": -67,
    "timestamp": 1368492027217,
    "capabilities": "[WPA2-EAP-CCMP][ESS]"
},
{
    "ssid": "MIT SECURE N",
    "bssid": "00:21:d8:4a:04:2b",
    "channelMhz": 5180,
    "signalDb": -68,
    "timestamp": 1368492027217,
    "capabilities": "[WPA2-EAP-CCMP][ESS]"
},
{
    "ssid": "2QS9H",
    "bssid": "00:7f:28:8f:32:0b",
    "channelMhz": 2437,
    "signalDb": -79,
    "timestamp": 1368492027217,
    "capabilities": "[WPA2-PSK-TKIP+CCMP][ESS]"
},
{
    "ssid": "MIT SECURE",
    "bssid": "00:21:d8:49:1d:e3",
    "channelMhz": 2462,
    "signalDb": -80,
    "timestamp": 1368492027217,
    "capabilities": "[WPA2-EAP-CCMP][ESS]"
},
{
    "ssid": "MIT SECURE",
    "bssid": "00:21:d8:49:c6:f3",
    "channelMhz": 2437,
    "signalDb": -81,
    "timestamp": 1368492027217,
    "capabilities": "[WPA2-EAP-CCMP][ESS]"
},
{
    "ssid": "",
    "bssid": "00:21:d8:49:c6:f5",
    "channelMhz": 2437,
    "signalDb": -83,
    "timestamp": 1368492027217,
    "capabilities": "[WPA2-PSK-CCMP][ESS]"
},
{
    "ssid": "MIT SECURE",
    "bssid": "00:21:d8:49:c6:3c",
    "channelMhz": 5240,
    "signalDb": -91,
    "timestamp": 1368492027217,
    "capabilities": "[WPA2-EAP-CCMP][ESS]"
},
{
    "ssid": "MIT SECURE N",
    "bssid": "00:21:d8:49:c6:3b",
    "channelMhz": 5240,
    "signalDb": -93,
    "timestamp": 1368492027217,
    "capabilities": "[WPA2-EAP-CCMP][ESS]"
},
{
    "ssid": "MIT SECURE",
    "bssid": "00:21:d8:49:90:63",
    "channelMhz": 2462,
    "signalDb": -83,
    "timestamp": 1368492027217,
    "capabilities": "[WPA2-EAP-CCMP][ESS]"
},
{
    "ssid": "MIT N",
    "bssid": "00:21:d8:49:a3:8f",
    "channelMhz": 5805,
    "signalDb": -56,
    "timestamp": 1368492027217,
    "capabilities": "[ESS]"
},
{
    "ssid": "MIT GUEST",
    "bssid": "00:21:d8:49:a3:8e",
    "channelMhz": 5805,
    "signalDb": -57,
    "timestamp": 1368492027217,
    "capabilities": "[ESS]"
},
{
    "ssid": "MIT GUEST",
    "bssid": "00:21:d8:49:a3:81",
    "channelMhz": 2437,
    "signalDb": -57,
    "timestamp": 1368492027217,
    "capabilities": "[ESS]"
},
{
    "ssid": "MIT",
    "bssid": "00:21:d8:4a:04:2d",
    "channelMhz": 5180,
    "signalDb": -68,
    "timestamp": 1368492027217,
    "capabilities": "[ESS]"
},
{
    "ssid": "MIT GUEST",
    "bssid": "00:21:d8:4a:04:2e",
    "channelMhz": 5180,
    "signalDb": -67,
    "timestamp": 1368492027217,
    "capabilities": "[ESS]"
},
{
    "ssid": "MIT N",
    "bssid": "00:21:d8:4a:04:2f",
    "channelMhz": 5180,
    "signalDb": -69,
    "timestamp": 1368492027217,
    "capabilities": "[ESS]"
},
{
    "ssid": "MIT",
    "bssid": "00:21:d8:49:c3:bd",
    "channelMhz": 5805,
    "signalDb": -80,
    "timestamp": 1368492027217,
    "capabilities": "[ESS]"
},
{
    "ssid": "MIT GUEST",
    "bssid": "00:21:d8:49:c3:be",
    "channelMhz": 5805,
    "signalDb": -81,
    "timestamp": 1368492027217,
    "capabilities": "[ESS]"
},
{
    "ssid": "MIT GUEST",
    "bssid": "00:21:d8:49:c6:3e",
    "channelMhz": 5240,
    "signalDb": -91,
    "timestamp": 1368492027217,
    "capabilities": "[ESS]"
},
{
    "ssid": "MIT GUEST",
    "bssid": "00:21:d8:49:bc:8e",
    "channelMhz": 5745,
    "signalDb": -91,
    "timestamp": 1368492027217,
    "capabilities": "[ESS]"
},
{
    "ssid": "MIT N",
    "bssid": "00:21:d8:49:c6:3f",
    "channelMhz": 5240,
    "signalDb": -94,
    "timestamp": 1368492027217,
    "capabilities": "[ESS]"
},
{
    "ssid": "MIT",
    "bssid": "00:21:d8:49:c6:3d",
    "channelMhz": 5240,
    "signalDb": -92,
    "timestamp": 1368492027217,
    "capabilities": "[ESS]"
},
{
    "ssid": "MIT",
    "bssid": "00:21:d8:49:fc:92",
    "channelMhz": 2462,
    "signalDb": -90,
    "timestamp": 1368492027217,
    "capabilities": "[ESS]"
},
{
    "ssid": "MIT GUEST",
    "bssid": "00:21:d8:49:90:61",
    "channelMhz": 2462,
    "signalDb": -84,
    "timestamp": 1368492027217,
    "capabilities": "[ESS]"
},
{
    "ssid": "MIT",
    "bssid": "00:21:d8:49:8b:b2",
    "channelMhz": 2437,
    "signalDb": -92,
    "timestamp": 1368492027217,
    "capabilities": "[ESS]"
},
{
    "ssid": "",
    "bssid": "00:21:d8:4a:04:25",
    "channelMhz": 2462,
    "signalDb": -52,
    "timestamp": 1368492027217,
    "capabilities": "[WPA2-PSK-CCMP][ESS]"
},
{
    "ssid": "MIT SECURE",
    "bssid": "00:21:d8:49:c3:b3",
    "channelMhz": 2412,
    "signalDb": -72,
    "timestamp": 1368492027217,
    "capabilities": "[WPA2-EAP-CCMP][ESS]"
},
{
    "ssid": "MIT SECURE",
    "bssid": "00:21:d8:49:c6:33",
```

```
            "channelMhz": 2412,
            "signalDb": -85,
            "timestamp": 1368492027217,
            "capabilities": "[WPA2-EAP-CCMP][ESS]"
        },
        {
            "ssid": "MIT SECURE N",
            "bssid": "00:21:d8:49:1d:eb",
            "channelMhz": 5805,
            "signalDb": -92,
            "timestamp": 1368492027217,
            "capabilities": "[WPA2-EAP-CCMP][ESS]"
        },
        {
            "ssid": "MIT",
            "bssid": "00:21:d8:49:1d:e2",
            "channelMhz": 2462,
            "signalDb": -78,
            "timestamp": 1368492027217,
            "capabilities": "[ESS]"
        },
        {
            "ssid": "MIT N",
            "bssid": "00:21:d8:49:c3:bf",
            "channelMhz": 5805,
            "signalDb": -85,
            "timestamp": 1368492027217,
            "capabilities": "[ESS]"
        }
    ]
},
"gsm": {
    "enabled": true,
    "type": 2,
    "cid": 60493,
    "lac": 5103,
    "rssi": 8,
    "ber": 0,
    "numOfCells": 1,
    "cells": [
        {
            "type": 2,
            "cid": 60491,
            "lac": 5103,
            "rssi": 2,
            "psc": -1
        }
    ]
},
"ndt": {
    "provider": "wifi",
    "cwndtime": "0.9020",
    "PktsIn": "9399",
    "SmoothedRTT": "18",
    "AckPktsOut": "0",
    "AckPktsIn": "9399",
    "bw": "17.77",
    "congestion": "0",
    "c2sData": "6",
    "SubsequentTimeouts": "0",
    "CurCwnd": "52128",
    "DataBytesIn": "0",
    "cwin": "0.5192",
    "WinScaleSent": "7",
    "SndLimTimeRwin": "870708",
    "StartTimeUsec": "789481",
    "s2cData": "8",
    "MinRTT": "9",
    "CurRTO": "218",
    "MaxMSS": "1448",
    "spd": "18.09",
    "bad_cable": "0",
    "rwin": "0.5083",
    "aspd": "0.00000",
    "SndLimBytesRwin": "2379840",
    "rttsec": "0.018214",
    "MinRTO": "210",
    "Timeouts": "1",
    "timesec": "10.00",
    "MaxRwinSent": "5888",
    "SlowStart": "247",
    "PktsRetrans": "72",
    "WinScaleRcvd": "5",
    "SndWinScale": "5",
    "s2cAck": "7",
    "RcvWinScale": "7",
    "NagleEnabled": "1",
    "DSACKDups": "18",
    "minCWNDpeak": "2896",
    "DupAcksIn": "589",
    "MinRwinSent": "5792",
    "MaxSsthresh": "94894680",
    "MinRwinRcvd": "14624",
    "DataPktsIn": "0",
    "half_duplex": "0",
    "CWNDpeaks": "20",
    "DataPktsOut": "15458",
    "SampleRTT": "20",
    "maxCWNDpeak": "68056",
    "waitsec": "0.22",
    "loss": "0.001164446",
    "DupAcksOut": "0",
    "SndLimTransCwnd": "50",
    "FastRetran": "17",
    "link": "0",
    "X_Sndbuf": "176344",
    "SndLimBytesCwnd": "20366280",
    "ECNEnabled": "0",
    "BytesRetrans": "104256",
    "PktsOut": "15458",
    "OtherReductions": "132",
    "CWND-Limited": "907.95",
    "CongestionOverCount": "141",
    "MaxRTO": "264",
    "rwintime": "0.0861",
    "order": "0.0627",
    "SndLimTimeSender": "120444",
    "MaxRTT": "116",
    "swin": "1.3454",
    "CongestionSignals": "18",
    "Duration": "10113834",
    "SumRTT": "160244",
    "sendtime": "0.0119",
    "SACKsRcvd": "906",
    "Client": "localhost/127.0.0.1",
    "CountRTT": "8798",
    "CongAvoid": "7322",
    "X_Rcvbuf": "87380",
    "MaxCwnd": "68056",
    "TimestampsEnabled": "1",
    "SndLimBytesSender": "119608",
    "CurSsthresh": "21720",
    "MinMSS": "1448",
    "CurRwinSent": "5888",
    "SndLimTransRwin": "46",
    "SendStall": "0",
    "avgrtt": "18.21",
    "SACKEnabled": "3",
    "CurRwinRcvd": "66624",
    "CurMSS": "1448",
    "SndLimTransSender": "25",
    "mismatch": "0",
    "MaxRwinRcvd": "66624",
    "c2sAck": "6",
    "SndLimTimeCwnd": "9121960",
    "DataBytesOut": "22865728",
    "Sndbuf": "176344"
  }
 }
}
```