



Webroot Mobile Security vs. Nine Competitor Mobile Security Products

(August 2015)

Android KitKat 4.4.2

Mobile Security Performance Benchmark

Document: Webroot Mobile Security vs. Nine Competitor Mobile Security Products
Authors: M. Baquiran, D. Wren
Company: PassMark Software
Date: 30 July 2015
File: Webroot_Mobile_Security_vs_9_Competitors_2015.docx
Edition: 1

Table of Contents

TABLE OF CONTENTS.....	2
REVISION HISTORY.....	3
REFERENCES.....	3
EXECUTIVE SUMMARY.....	4
OVERALL SCORE.....	5
PRODUCTS AND VERSIONS.....	6
PERFORMANCE METRICS SUMMARY.....	7
TEST RESULTS.....	9
BENCHMARK 1 – PACKAGE SIZE (MB).....	9
BENCHMARK 2 – POST INSTALLATION INFLATION (MB).....	9
BENCHMARK 3 – SIZE ON DISK (MB).....	10
BENCHMARK 4 – INSTALLATION TIME (SEC).....	10
BENCHMARK 5 – NETWORK USAGE DURING INSTALLATION (MB).....	11
BENCHMARK 6 – BATTERY USAGE DURING INSTALLATION (%).....	11
BENCHMARK 7 – MEMORY USAGE DURING IDLE (MB).....	12
BENCHMARK 8 – CPU USAGE DURING IDLE (%).....	12
BENCHMARK 9 – BATTERY USAGE DURING IDLE (%).....	13
BENCHMARK 10 – MEMORY USAGE DURING APP ACTIVITY (MB).....	13
BENCHMARK 11 – CPU USAGE DURING APP ACTIVITY (%).....	14
BENCHMARK 12 – BATTERY USAGE DURING APP ACTIVITY (%).....	14
BENCHMARK 13 – MEMORY USAGE DURING DEVICE ACTIVITY (MB).....	15
BENCHMARK 14 – CPU USAGE DURING DEVICE ACTIVITY (%).....	15
BENCHMARK 15 – BATTERY USAGE DURING DEVICE ACTIVITY (%).....	16
BENCHMARK 16 – SCAN TIME (SEC).....	16
BENCHMARK 17 – 7-DAY NETWORK USAGE (MB).....	17
CONTACT DETAILS.....	18
APPENDIX 1 – TEST ENVIRONMENT.....	19
APPENDIX 2 – METHODOLOGY DESCRIPTION.....	20

Revision History

Rev	Revision History	Date
Edition 1	Initial version of this report, includes new results for ten (10) mobile security products.	31 July 2015

References

Ref #	Document	Author	Date
1	Process Stats: Understanding How Your App Uses RAM (URL)	Dianne Hackborn	2014-2015

Executive Summary

PassMark Software® conducted objective performance testing on ten (10) mobile security apps, on Android KitKat 4.4.2 during June and July 2015. This report presents our results and findings as a result of performance benchmark testing conducted for these consumer antivirus products.

The aim of this benchmark was to compare the performance impact of Webroot's Mobile Security product with nine (9) competitor mobile security products. Testing was performed on all products using seventeen (17) performance metrics. These performance metrics are as follows:

- Download Package Size;
- Post Installation Inflation;
- Size on Disk;
- Installation Time;
- Network Usage during Installation;
- Battery Usage during Installation;
- Memory Usage during Idle;
- CPU Usage during Idle;
- Battery Usage during Idle;
- Memory Usage during App Activity;
- CPU Usage during App Activity;
- Battery Usage during App Activity;
- Memory Usage during Device Activity;
- CPU Usage during Device Activity;
- Battery Usage during Device Activity;
- Scan Time; and
- 7-day Network Usage.

Overall Score

PassMark Software assigned every product a score depending on its ranking in each metric compared to other products in the same category. In the following table the highest possible score attainable has been normalized to 100. This would be the score given if a product attained first place in all seventeen (17) metrics. Products have been ranked by their overall scores:

Product Name	Overall Score
Webroot Mobile Security	87
ESET Mobile Security	74
Kingsoft Security Plus	72
Trend Micro Mobile Security	71
Bitdefender Mobile Security & Antivirus	69
Norton Security & Antivirus	47
Lookout Security	42
Kaspersky Internet Security	42
McAfee Security & Antivirus	41
IKARUS mobile.security	41

Products and Versions

For all products we have tested the full, retail release of the most current, publicly available version of each antivirus product. The names and versions of products are given below:

Manufacturer	Product Name	Release Year	Product Version	Date Tested
ESET, spol. s r.o.	ESET Mobile Security	2015	3.01318.0	June 2015
Webroot Inc.	Webroot Mobile Security	2015	3.6.0.6677	June 2015
Symantec Corp	Norton Security & Antivirus	2015	3.11.0.2511	June 2015
Trend Micro Inc.	Trend Micro Mobile Security	2015	6.00	July 2015
McAfee, Inc.	McAfee Security & Antivirus	2015	4.4.0.467	July 2015
Kingsoft Corp	Kingsoft Mobile Security Plus	2015	4.0.2.3	July 2015
Bitdefender	Bitdefender Antivirus	2015	2.49.891	June 2015
IKARUS Security Software GmbH	IKARUS mobile.security	2015	1.7.21	July 2015
Lookout	Lookout Security	2015	9.23-da2fc8f	July 2015
Kaspersky Lab	Kaspersky Internet Security	2015	11.8.4.625	June 2015

Performance Metrics Summary

The following metrics have been selected with the view to provide a comprehensive and realistic indication of the areas in which the mobile security app may impact an Android device's performance for end-users. Our methodology is designed to emulate a typical user experience and thus include scenarios in which common tasks are performed on the device both manually by the end-user and in the background by the security app.

All of PassMark Software's test methods can be replicated by third parties using the same environment to obtain similar benchmark results. Detailed descriptions of the methodologies used in our tests are available as "[Appendix 2 – Methodology Description](#)" of this report.

Benchmark 1 – Package Size (MB)

This metric measures the installation package size of the security app that is downloaded via the Google Play store.

Benchmark 2 – Post Installation Inflation (MB)

This metric measures the difference in available space before and after the security app has been installed on the device.

Benchmark 3 – Size on Disk (MB)

This metric measures the amount of disk space used by the security app once it is installed on the device.

Benchmark 4 – Installation Time (sec)

This metric measures the total time taken to install the app on the device. The installation process includes download, setup, and the first device scan.

Benchmark 5 – Network Usage During Installation (MB)

This metric measures the data usage by the device over the course of installation. The installation process includes download, setup, and the first device scan.

Benchmark 6 – Battery Usage During Installation (%)

This metric measures the battery usage over the course of installation. The installation process includes download, setup, and the first device scan.

Benchmark 7 – Memory Usage During Idle (MB)

This metric measures the memory usage by the security app during a 2 hour idle period.

Benchmark 8 – CPU Usage During Idle (%)

This metric measures the percentage of CPU Usage by the security app during a 2 hour idle period.

Benchmark 9 – Battery Usage During Idle (%)

This metric measures the overall battery impact on the device over a 2-day idle period.

Benchmark 10 – Memory Usage During App Activity (MB)

This metric measures the average memory usage by the security app during a period in which the security app is used to carry out tasks.

Benchmark 11 – CPU Usage During App Activity (%)

This metric measures the average CPU usage by the security app during a period in which the security app is used to carry out tasks.

Benchmark 12 – Battery Usage During App Activity (%)

This metric measures the overall battery impact on the device during a period in which the security app is used to carry out tasks.

Benchmark 13 – Memory Usage During Device Activity (MB)

This metric measures the average memory usage by the security app during a period of typical device activity, including web browsing, app installation, and scheduled tasks.

Benchmark 14 – CPU Usage During Device Activity (%)

This metric measures the average CPU usage by the security app during a period of typical device activity, including web browsing, app installation, and scheduled tasks.

Benchmark 15 – Battery Usage During Device Activity (%)

This metric measures the average battery usage by the security app during a period in which typical during a period of typical device activity, including web browsing, app installation, and scheduled tasks.

Benchmark 16 – Scan Time (sec)

This metric measures the average time taken to carry out an on-demand scan on the device.

Benchmark 17 – 7-day Network Usage (MB)

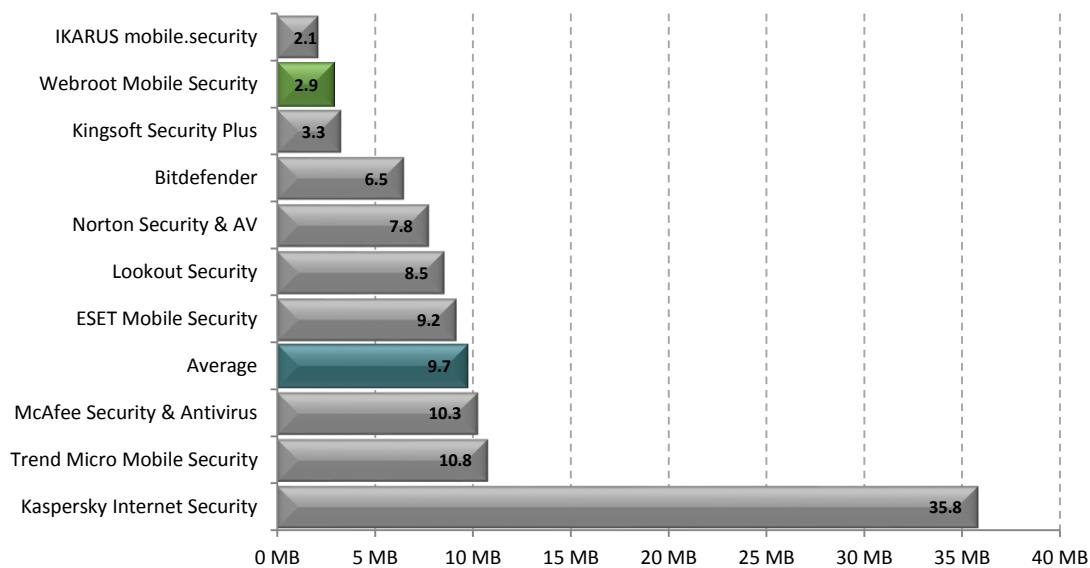
This metric measures the total data usage by the security app over a period of 7-days.

Test Results

In the following charts, we have highlighted the results we obtained for Webroot Mobile Security in green. The competitor average has also been highlighted in blue for ease of comparison.

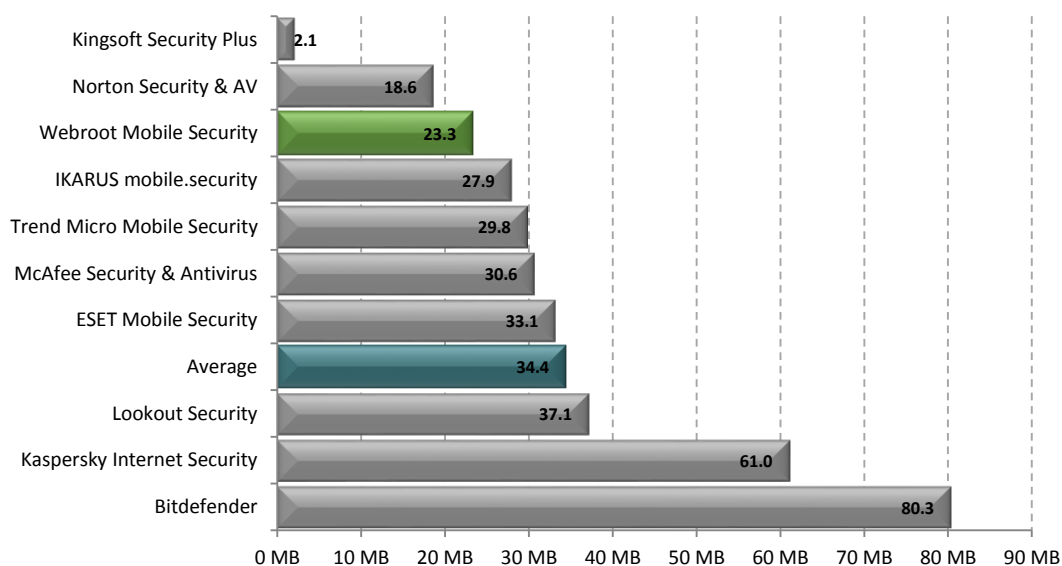
Benchmark 1 – Package Size (MB)

The following chart compares each security app's download package size as advertised by the Google Play store. Products with lower package sizes are considered better performing products in this category.



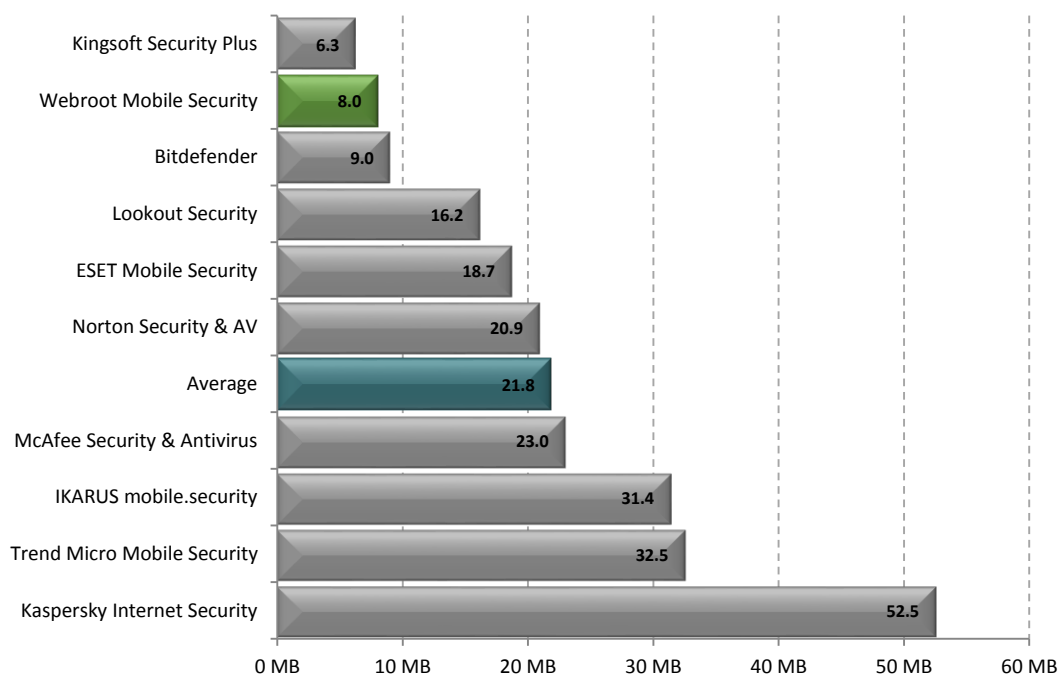
Benchmark 2 – Post Installation Inflation (MB)

The following chart compares the decrease in available space before and after each security app has been installed on the device. Products with lower inflation values are considered better performing products in this category.



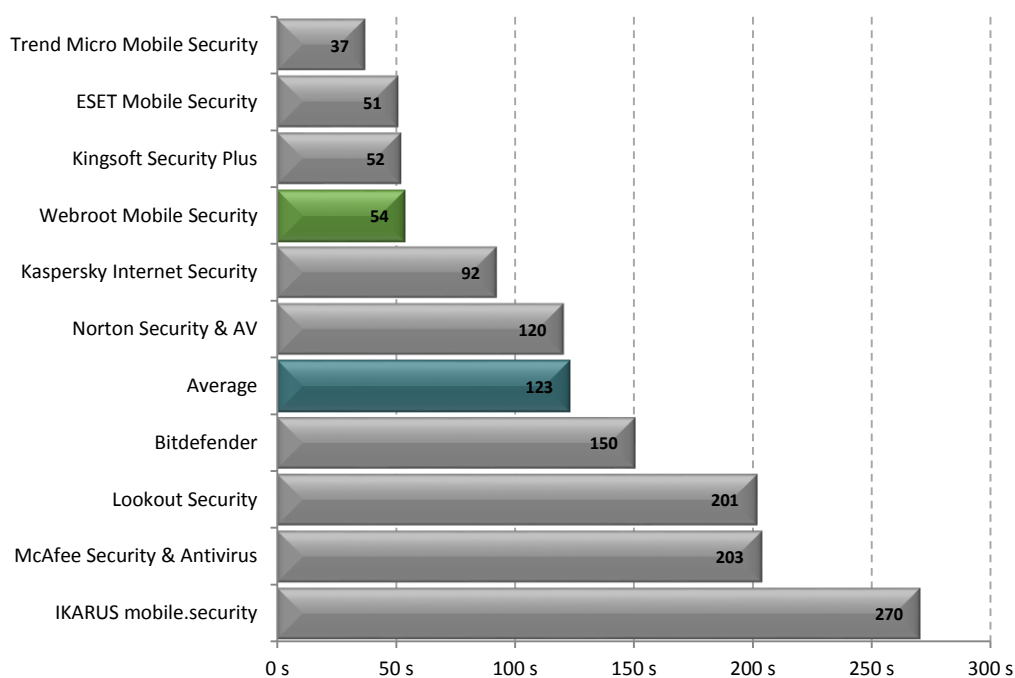
Benchmark 3 – Size on Disk (MB)

The following chart compares the total size of files added during the installation of each security app. Products with lower installation sizes are considered better performing products in this category.



Benchmark 4 – Installation Time (sec)

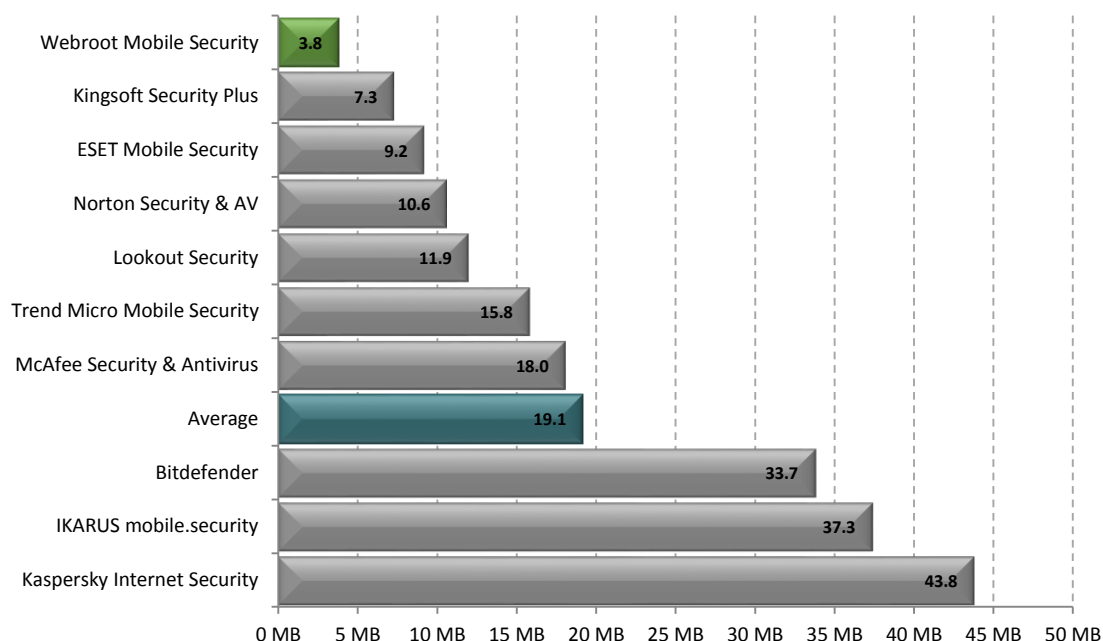
The following chart compares the time it takes for each security app to be fully functional and ready for use by the end-user. Products with lower installation times are considered better performing products in this category.



Benchmark 5 – Network Usage During Installation (MB)

The following chart compares the total data usage on the device over the course of each security app's installation.

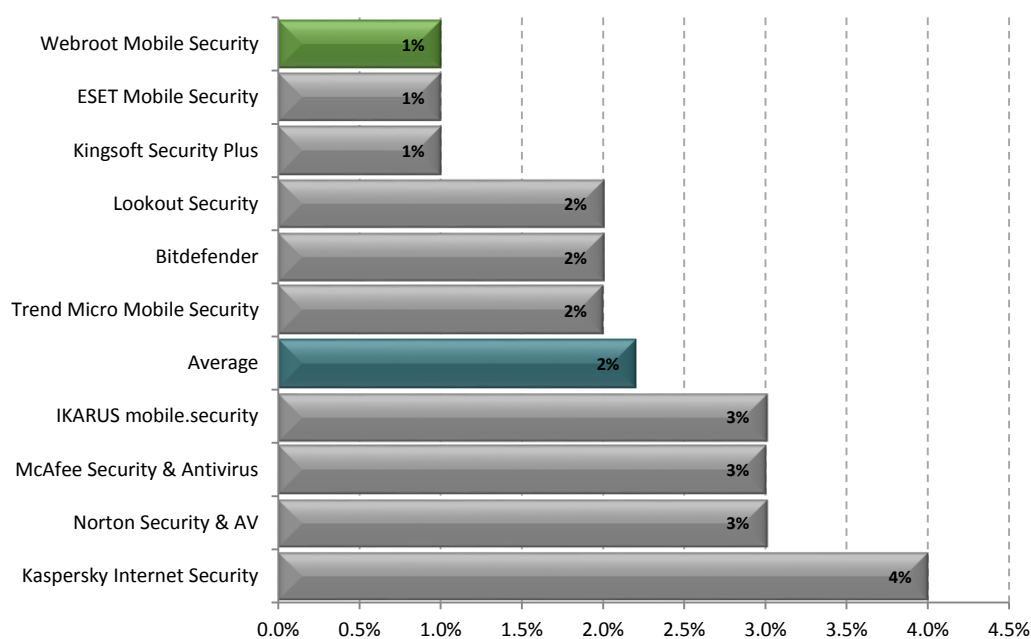
Products with lower usages are considered better performing products in this category.



Benchmark 6 – Battery Usage During Installation (%)

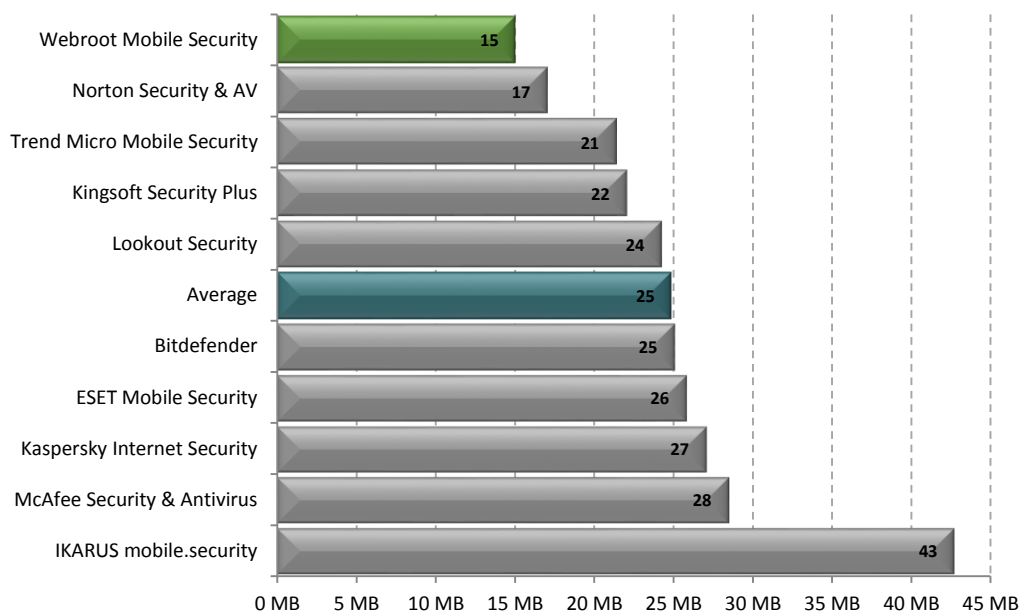
The following chart compares the decrease in battery charge over the course of each security app's installation.

Products with lower usages are considered better performing products in this category.



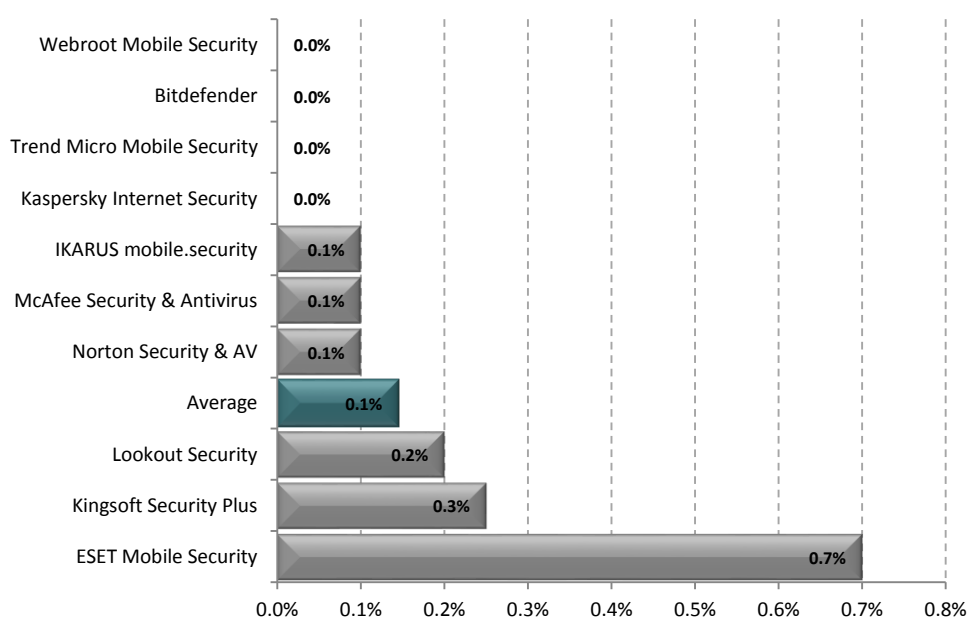
Benchmark 7 – Memory Usage During Idle (MB)

The following chart compares the average amount of RAM used by each security app during a 2 hour period of idle. Products with lower memory usages are considered better performing products in this category.



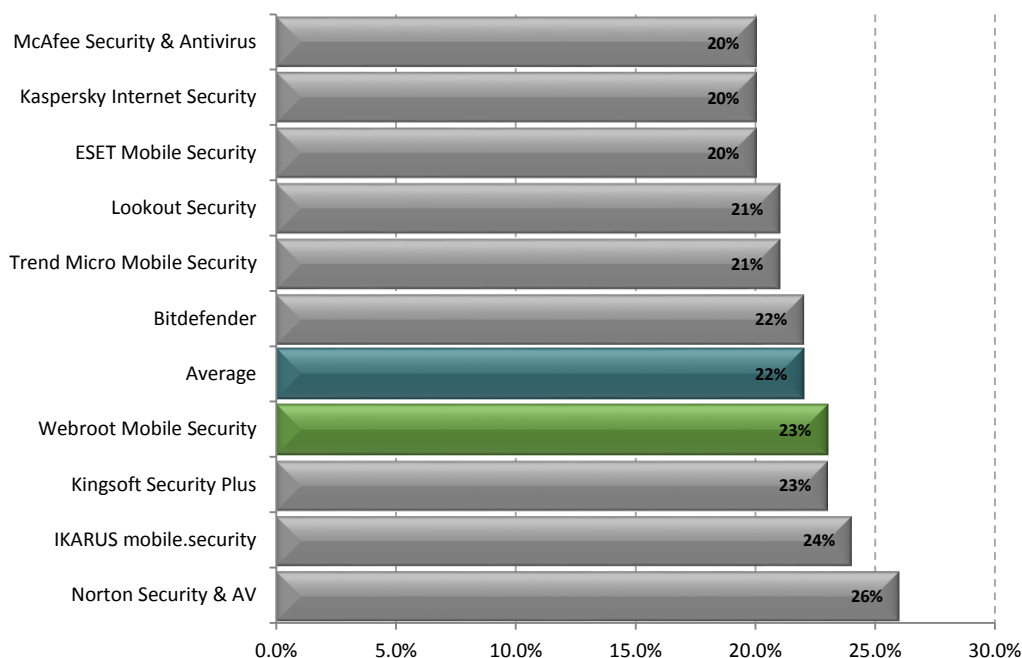
Benchmark 8 – CPU Usage During Idle (%)

The following chart compares the average CPU usage by each security app during a 2 hour period of idle. Products with lower CPU usages are considered better performing products in this category.



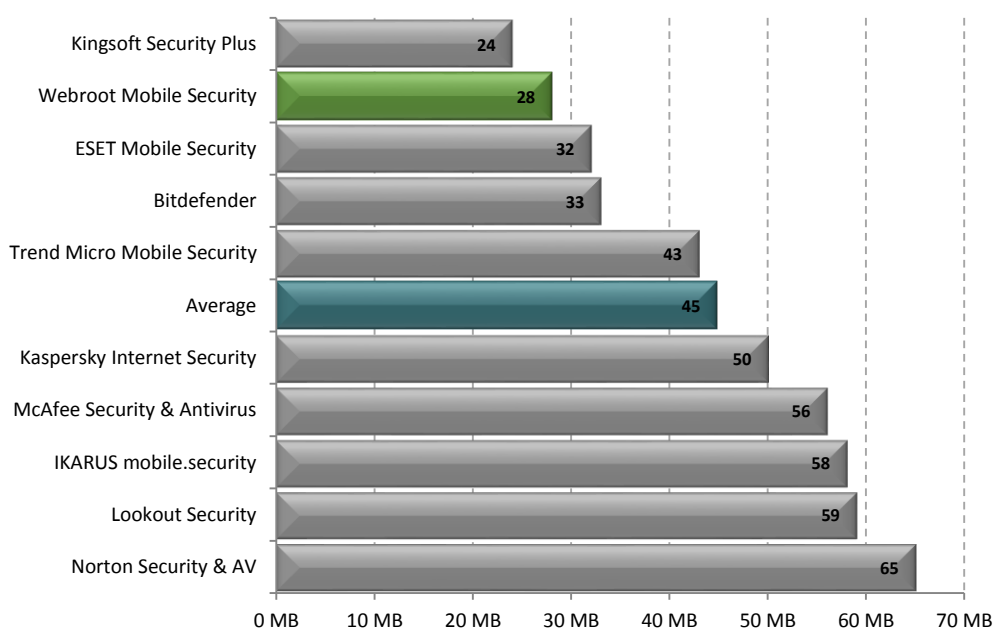
Benchmark 9 – Battery Usage During Idle (%)

The following chart compares, for each security app, the overall battery impact on the device over a 2-day period of idle. Products with lower battery usages are considered better performing products in this category.



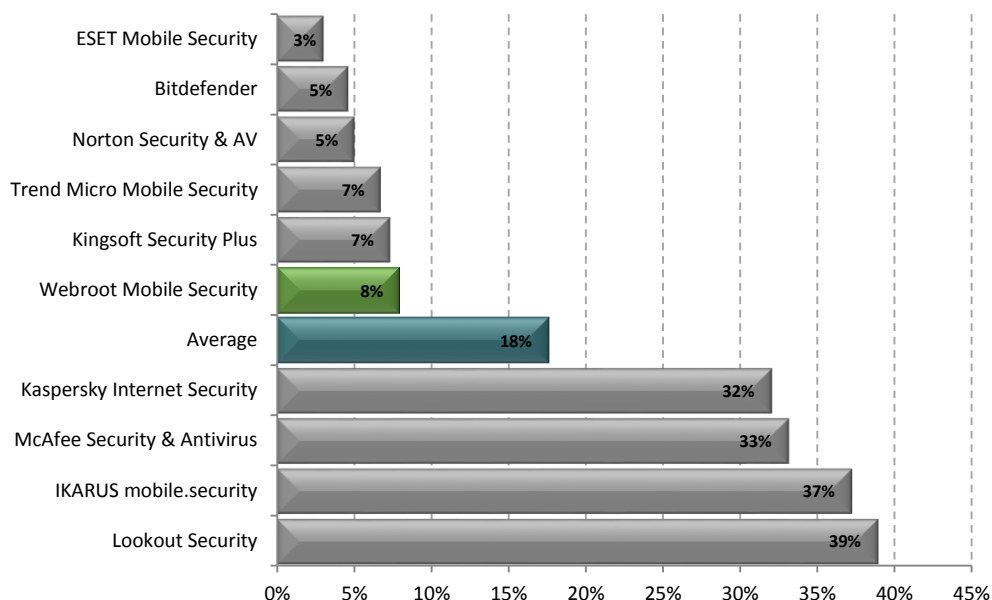
Benchmark 10 – Memory Usage During App Activity (MB)

The following chart compares the average amount of RAM used by each security app during a period in which the user carries out tasks with the app. Products with lower memory usages are considered better performing products in this category.



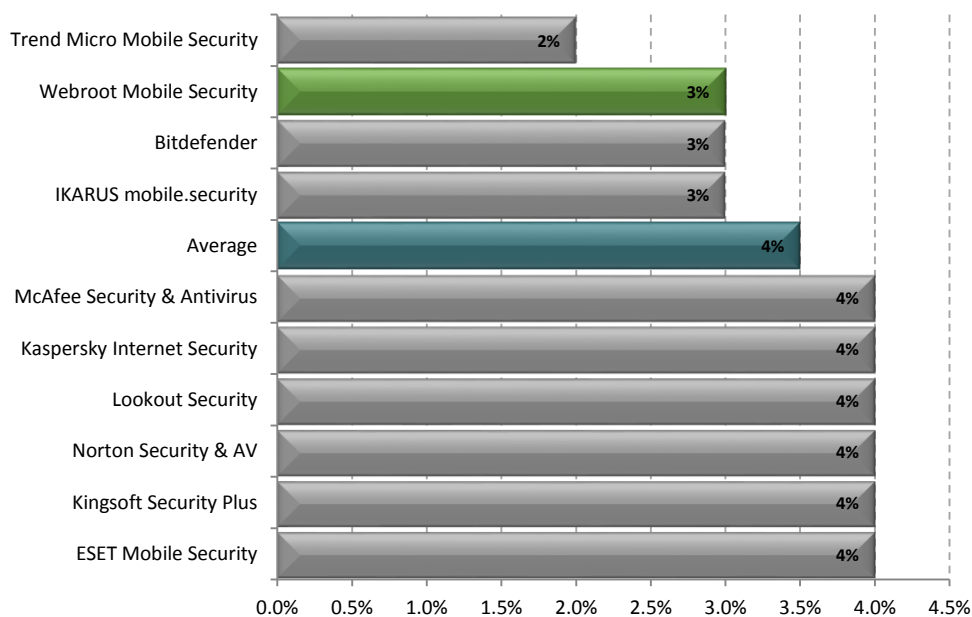
Benchmark 11 – CPU Usage During App Activity (%)

The following chart compares the average amount of CPU usage by each security app during a period in which the user carries out tasks with the app. Products with lower CPU usages are considered better performing products in this category.



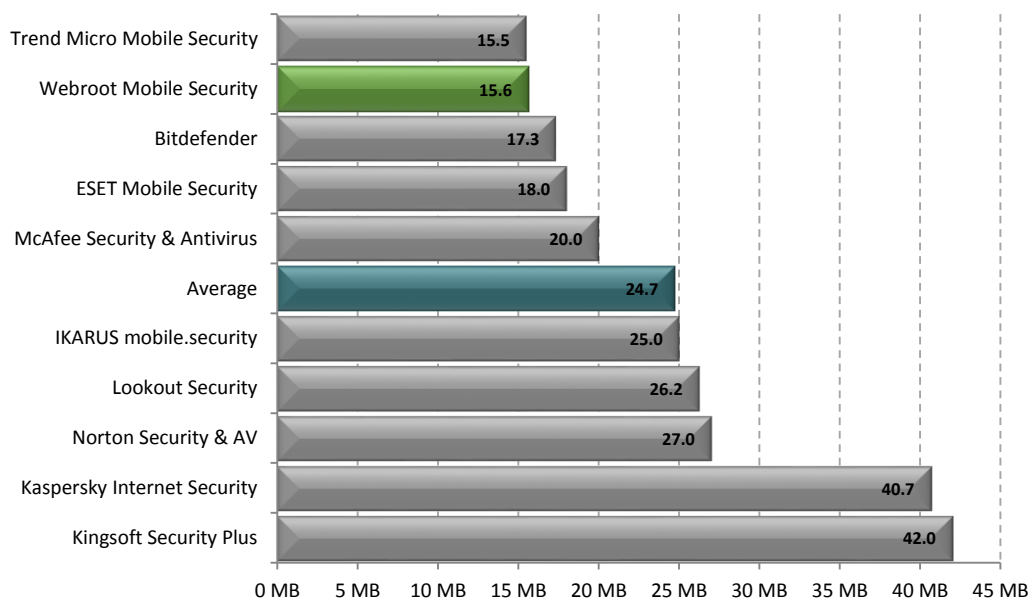
Benchmark 12 – Battery Usage During App Activity (%)

The following chart compares, for each security app, the average amount of battery charge used on the device during a period in which the user carries out tasks with the app. Products with lower battery usages are considered better performing products in this category.



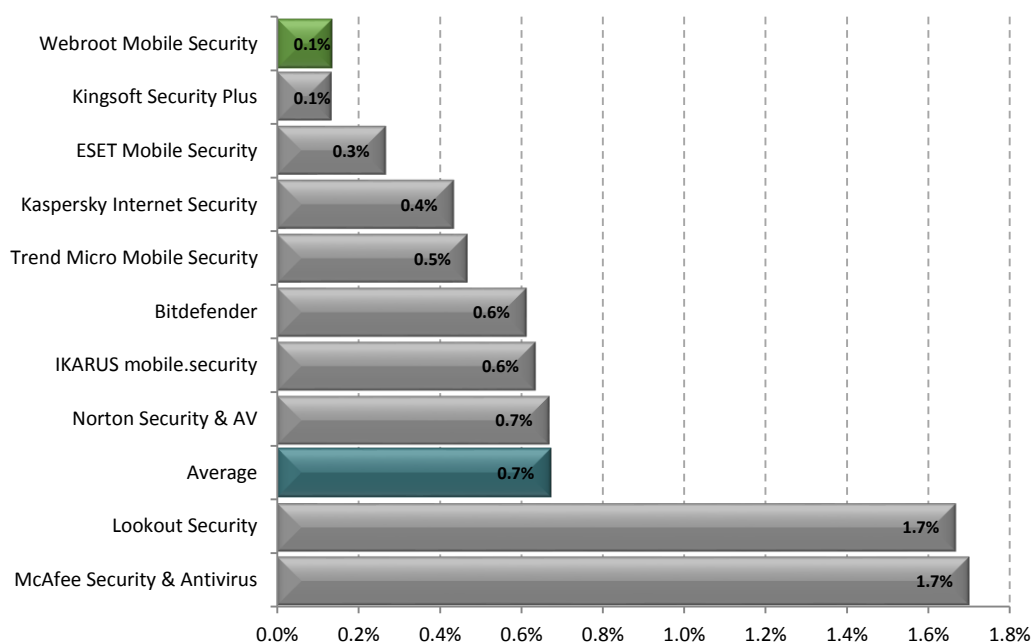
Benchmark 13 – Memory Usage During Device Activity (MB)

The following chart compares the average amount of RAM in use by each security app during a period in which typical activity occurs on the device. This includes web browsing, installing apps, and a scheduled security scan. Products with lower memory usages are considered better performing products in this category.



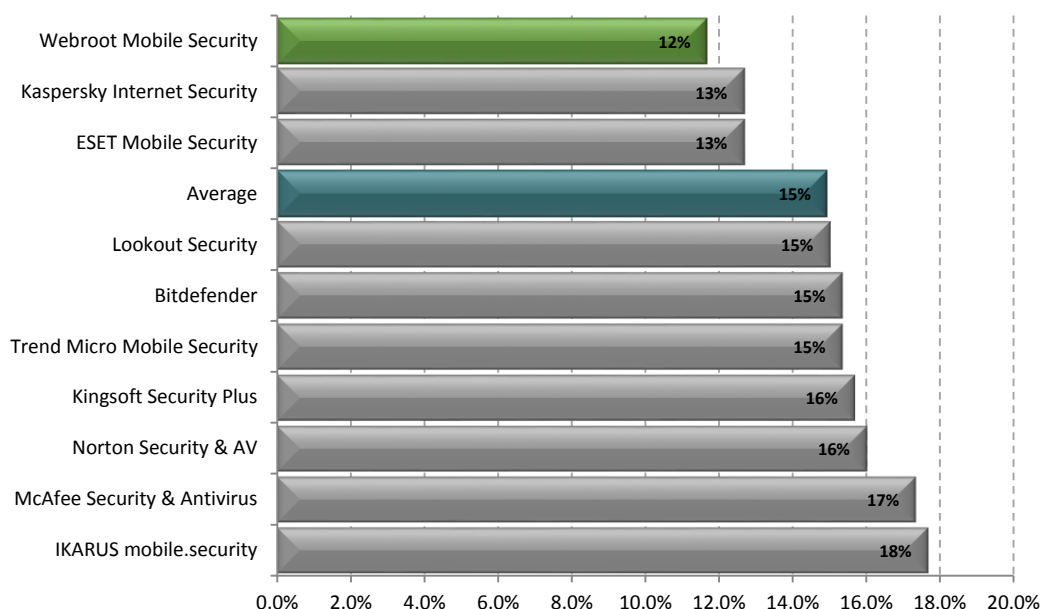
Benchmark 14 – CPU Usage During Device Activity (%)

The following chart compares the average CPU usage by each security app during a period in which typical activity occurs on the device. This includes web browsing, installing apps, and a scheduled security scan. Products with lower CPU usages are considered better performing products in this category.



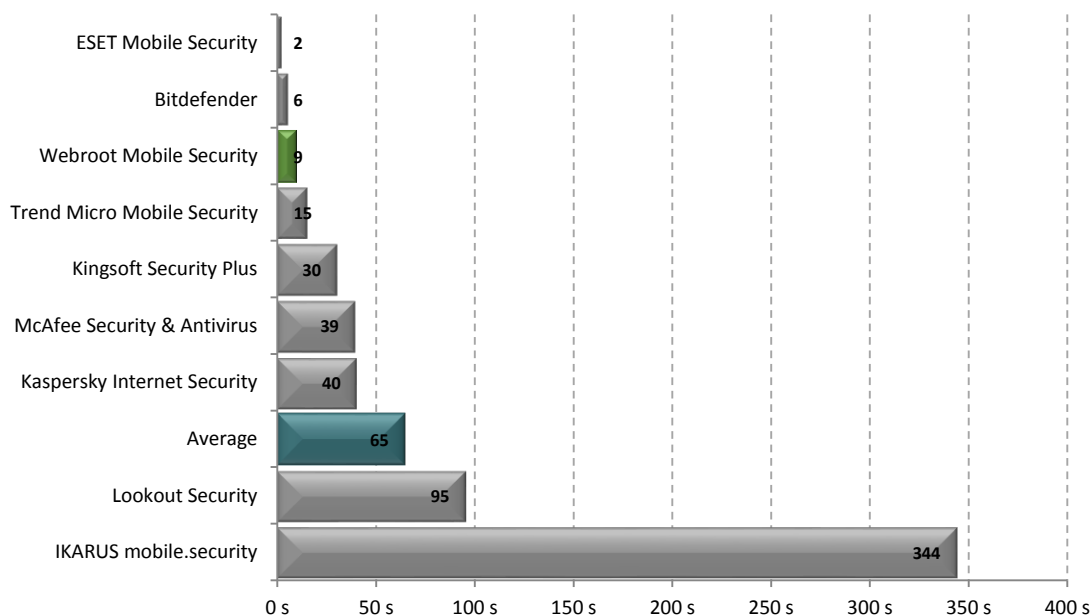
Benchmark 15 – Battery Usage During Device Activity (%)

The following chart compares, for each security app, the overall battery impact on the device during a period in which typical activity occurs on the device. This includes web browsing, installing apps, and a scheduled security scan. Products with lower battery usages are considered better performing products in this category.



Benchmark 16 – Scan Time (sec)

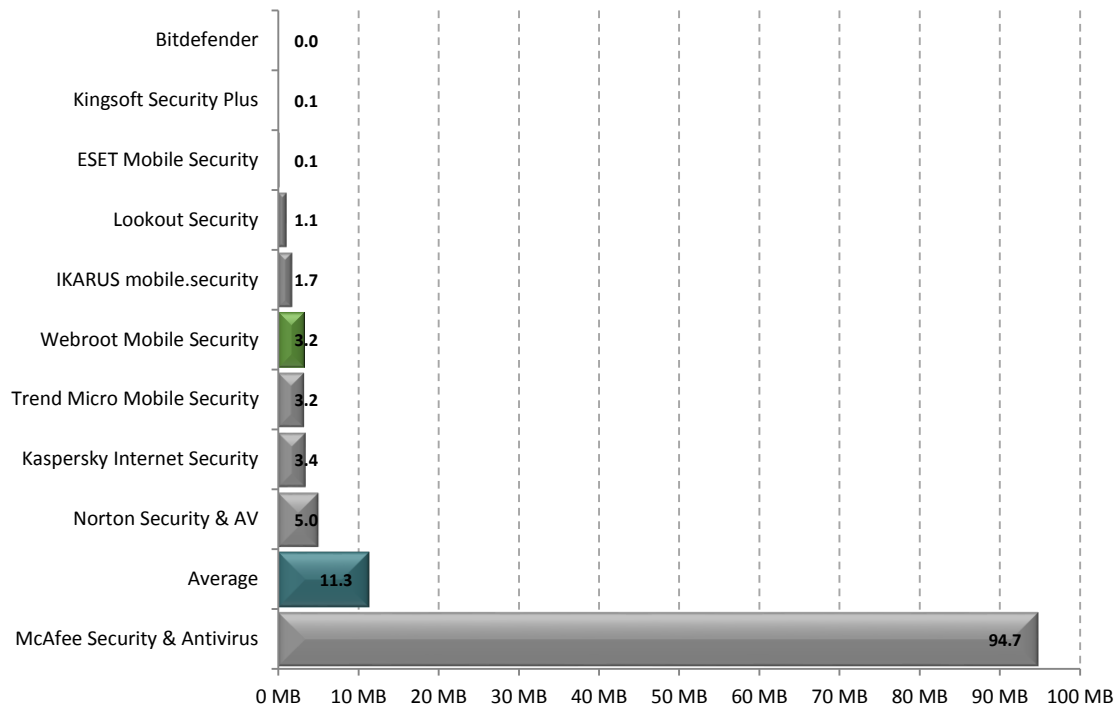
The following chart compares the average time taken to scan the entire device using each security app's on-demand scan function. Products with lower scan times are considered better performing products in this category.*



*Norton was omitted from this chart as it did not appear to have an on-demand scan functionality.

Benchmark 17 – 7-day Network Usage (MB)

The following chart compares the total data usage by each security app over a 7-day period. Products with lower usages are considered better performing products in this category.



Disclaimer and Disclosure

This report only covers versions of products that were available at the time of testing. The tested versions are as noted in the “Products and Versions” section of this report. The products we have tested are not an exhaustive list of all products available in these very competitive product categories.

Disclaimer of Liability

While every effort has been made to ensure that the information presented in this report is accurate, PassMark Software Pty Ltd assumes no responsibility for errors, omissions, or out-of-date information and shall not be liable in any manner whatsoever for direct, indirect, incidental, consequential, or punitive damages resulting from the availability of, use of, access of, or inability to use this information.

Disclosure

Webroot Inc. funded the production of this report. The list of products tested and the metrics included in the report were selected by Webroot.

Trademarks

All trademarks are the property of their respective owners.

Contact Details

PassMark Software Pty Ltd

Level 5

63 Foveaux St.

Surry Hills, 2010

Sydney, Australia

Phone + 61 (2) 9690 0444

Fax + 61 (2) 9690 0445

Web www.passmark.com

Appendix 1 – Test Environment

For our testing, we used five (5) mobile devices to test five (5) products at a time. All devices were identical and had the following specifications:

Mobile Device

Model:	Samsung Galaxy S4
Removable Storage:	microSD card
RAM:	2048MB
Android OS Version:	Android 4.4.2 (KitKat)
CPU:	Quad core, 1900 MHz, Krait 300
USB:	USB 2.0

Samsung Galaxy S4 Test Environment Setup

The following process was used to create a “clean” baseline state to be restored to each device at the start of each product test round. This process reduces unwanted variation in the test environment between product test rounds.

The steps taken to achieve the baseline device state are as follows:

1. Restore device to factory settings.
2. Disable automatic GPS location.
3. Sign in to Google Account.
4. Disable automatic software updates.
5. Disable automatic app updates in Google Play store.
6. Install Apps required for testing.
7. Place sample data files on device’s internal storage (see [Data Set #3](#)).
8. Turn on Developer mode and enable USB debugging (for process stats analysis).
9. Charge battery to full capacity and unplug from charger.

Appendix 2 – Methodology Description

Download Package Size (MB)

This is the size of the download package required from the **Google Play** store to install the app. The app's download page in the **Google Play** store provides the size in megabytes under "Read more".

Post Installation Inflation (MB)

This metric measures the decrease in available storage space on the device before and after the security app has been installed. The total free internal storage available is recorded in megabytes both before and after the app is installed, using the third-party app **SystemPanel**. The difference is then calculated to give the result.

Size on Disk (MB)

This metric measures the total space taken up by the security app on the device. The result is taken from the "Total", given in megabytes, under the security app's App info profile within **Application Manager**.

Installation Time (seconds)

The total installation process is defined as the "end-to-end experience" and is split into the following three phases:

1. Download – This is the time taken to download the app from the **Google Play** store. The time is started as soon as the "Accept" button is pressed. The phase finishes as soon as the progress bar changes to say "Installing...".
2. Installing, First app launch and Setup – This phase begins as soon as the progress bar says "Installing...". Once a notification is given to say that the app has been installed, the app is launched and the necessary setup steps are completed. This may include signing up for an account, product activation, settings configuration, and mandatory updates.
3. First device scan – This is the time taken to complete the first scan on the device, whether automatic or manual. For some products, the first device scan is run automatically immediately after the application has completed setup. Otherwise, the scan is initiated manually via the app's interface.

Each phase is timed manually using a stopwatch. The total of all three phases in seconds gives the result.

Network Usage During Installation (MB)

This metric measures the total network usage on the device over the course of the security app's installation. The total network usage given under **Connections -> Data Usage**, given in megabytes, is measured both before and after the installation (the total end-to-end experience) is complete. The difference in these two values is then calculated to give the result.

Battery Usage During Installation (%)

This metric measures the overall battery impact on the device over the course of the security app's installation. The battery charge is recorded both before and after the security app has been installed, and the difference is calculated to give the result in megabytes.

Memory Usage During Idle (MB)

This metric measures the average memory usage by the security app over a 2 hour period of idle. Before the period begins, the device is rebooted and then left idle for 2 hours with automatic updates and idle scans turned off where possible. Once the 2 hour idle period is complete, the device is connected to a machine with the **Android SDK** installed. The **Android Debug Bridge** (adb) command line tool is used to connect to the device using the command:

```
adb devices
```

Using the system service **procstats**, the memory information for the security app's service over the last 2 hours is dumped and saved. For example, when testing Lookout Security, the following command was used:

```
adb shell dumpsys procstats com.lookout --hours 2
```

For more information on obtaining memory dumps using **procstats**, please see [Reference #1](#).

The overall average (avgUSS) for that service is then taken as the result. USS is the private memory assigned to the process, whereas PSS includes the proportional size its shared libraries. The latter may be misleading in this context and thus the former is used for our result.

CPU Usage During Idle (%)

This metric measures the average CPU usage by the security app during a 2 hour period of idle. In the third-party app **SystemPanel**, Monitoring is enabled and then the device is rebooted. The device is then left idle for 2 hours with automatic updates and idle scans turned off where possible. Once the period is finished, **SystemPanel** gives the security app's average CPU Consumption as a percentage (under *Active Applications -> Security App Name -> Current Session -> CPU Usage -> Average Consumption*).

Battery Usage During Idle (%)

This metric measures the change in battery charge on the device with the app installed over a 2-day period. Automatic updates and idle scans are switched off where possible, and the device is charged to 100% battery capacity. The charger is then unplugged and left for a period of 2-days. Once the 2-day idle period is complete, the battery charge is recorded. The decrease in battery charge gives the result as a percentage.

Memory Usage During App Activity (MB)

This metric measures the average memory usage by the security app during a 15 minute period in which the app is being actively used. The device is rebooted and then typical user tasks are carried out manually back-to-back within the security app (such as on-demand scans, updates, and settings configuration) for a period of 15 minutes. Once the 15 minute period is complete, the device is connected to a machine with the **Android SDK** installed. The **Android Debug Bridge** (adb) command line tool is used to connect to the device using the command:

```
adb devices
```

Using the system service **procstats**, the memory information for the security app's service over the last hour (which is the minimum time interval that can be specified) is dumped and saved using a command like the following:

```
adb shell dumpsys procstats com.lookout --hours 1
```

For more information on obtaining memory dumps using **procstats** please see [Reference #1](#).

To reduce unwanted variability from idle time that may occur in between tasks as a result of the manual nature of this test, the average (avgUSS) has been taken only over the time in which the service has the highest CPU usage on the device. This is given in the row labelled "Top" in the **procstats** dump.

CPU Usage During App Activity (%)

This metric measures the average CPU usage by the security app during a 15 minute period in which the app is being actively used. In the third party app **SystemPanel**, monitoring is enabled and then the device is rebooted. Typical user tasks are then carried out manually back-to-back within the security app (such as on-demand scans, updates, and settings configuration) for a period of 15 minutes. Once the period is finished, **SystemPanel** then gives the security app's average CPU Consumption as a percentage (under *Active Applications -> Security App Name -> Current Session -> CPU Usage -> Average Consumption*).

Battery Usage During App Activity (%)

This metric measures the change in battery charge on the device with the app installed over a 15 minute period in which the app is being actively used. Typical user tasks are then carried out manually back-to-back within the security app (such as on-demand scans, updates, and settings configuration) for a period of 15 minutes. The battery charge both before and after the period is recorded, and then the decrease is calculated to give the result as a percentage.

Memory Usage During Device Activity (MB)

This metric measures the average memory usage by the security app during a period of typical device activity in which the security app is used implicitly. This metric takes the average of two separate measurements:

1. The first measurement is the average memory usage over a 2 hour period in which the following tasks are carried out:
 - a. The user browses a list of 30 websites over 30 minutes (1 website per minute) using the default browser.
 - b. The user then installs a list of 10 third party apps from the **Google Play** store over 1.5 hours (about 10 minutes apart).

For these lists please see [Data Set #1](#) and [Data Set #2](#). The device is rebooted and then the above tasks are carried out manually. **Procstats** is then used in the same way as previously described (see [Memory Usage During Idle](#)) to obtain the memory usage by the security app over the past 2 hours. The average total USS is taken as the result.

2. The second measurement is the average memory usage by the security app when an automatic daily scheduled scan runs in the background. The task is configured via the security app's UI. In the case where

apps did not have a scheduled scan functionality, a scheduled update was enabled instead. Some security apps don't offer the option of specifying a time of a scheduled scan, so it was necessary to leave the device for at least one day in order to ensure that the scan has been carried out. Once the one day period has passed, **Procstats** is used in the same way as described above (see [Memory Usage During Idle](#)) to obtain the memory usage by the security app over the past 24 hours. To omit the large proportion of idle time in which the product is not carrying out the scheduled task, the average USS is taken from the row labelled "Top" in the **procstats** dump. This is the state in which the process is using the most amount of CPU on the device.

The weighted average of the above two measurements (two-thirds weighting to 1. and one-third weighting to 2.) is calculated to give the result in megabytes.

CPU Usage During Device Activity (%)

This metric measures the average CPU usage by the security app during a period of typical device activity in which the security app is used implicitly. This metric takes the average of two separate measurements:

1. The first measurement is the average CPU usage over a 2 hour period in which the following tasks are carried out:
 - a. The user browses a list of 30 websites over 30 minutes (1 website per minute) using the default browser.
 - b. The user then installs a list of 10 third party apps from the **Google Play** store over 1.5 hours (about 10 minutes apart).

For these lists please see [Data Set #1](#) and [Data Set #2](#). Before performing the above, monitoring is enabled in the third party app **SystemPanel** and the device is then rebooted. The above tasks are then carried out manually. Once they are complete, **SystemPanel** is used to give the security app's average CPU consumption as a percentage (under Active Applications -> Security App Name -> Current Session -> CPU Usage -> Average Consumption).

2. The second measurement is the average CPU usage by the security app when an automatic daily scheduled scan runs in the background. The task is configured via the security app's UI. In the case where apps did not have a scheduled scan functionality, a scheduled update was enabled instead. Some security apps don't offer the option of specifying a time of a scheduled scan, so it was necessary to leave the device for at least one day in order to ensure that the scan has been carried out. To omit the large proportion of idle time CPU Usage logged over the 1-day period, the peak CPU consumption is taken by examining the security app's historical CPU Usage chart in **SystemPanel**.

The weighted average of the above two measurements (two-thirds weighting to 1. and one-third weighting to 2.) is calculated to give the result as a percentage.

Battery Usage During Device Activity (%)

This metric measures the total battery usage on the device during a period of typical device activity in which the security app is used implicitly. This metric takes the average of two separate measurements:

1. The first measurement is the average CPU usage over a 2 hour period in which the following tasks are carried out:

- a. The user browses a list of 30 websites over 30 minutes (1 website per minute) using the default browser.
- b. The user then installs a list of 10 third party apps from the Google Play store over 1.5 hours (about 10 minutes apart). For these lists please see [Data Set #1](#) and [Data Set #2](#).

The battery charge both before and after the above tasks are carried out is recorded. The decrease in battery charge is then calculated as the result.

2. The second measurement is the average CPU usage by the security app when an automatic daily scheduled scan runs in the background. The task is configured via the security app's UI. In the case where apps did not have a scheduled scan functionality, a scheduled update was enabled instead. Some security apps don't offer the option of specifying a time of a scheduled scan, so it was necessary to leave the device for at least one day in order to ensure that the scan has been carried out. The battery charge both before and after the 1-day period is recorded and the decrease is calculated as the result.

The weighted average of the above two measurements (two-thirds weighting to 1. and one-third weighting to 2.) is calculated to give the result as a percentage.

Scan Time (sec)

This metric measures the average on-demand device scan time by the security app. Sample files are placed on the device's internal storage before testing begins (see [Data Set #3](#)) which may be included in the scan scope of some security apps depending on the security app's default behavior. Before each run, the device is rebooted to remove caching effects. The scan is then initiated from the security app's UI and the time to complete the scan is taken either manually or from the UI where reliable. A total of 10 scans are run and the average is calculated as the result in seconds.

7-day Network Usage (MB)

This metric measures the total network usage by the security app over a 7-day period. The security app's automatic updates are enabled and the device is left for a period of 7-days. The device's data usage for the security app (under *Connections* -> *Data Usage* -> *Security App Name*) is recorded daily. Throughout the test it is ensured that the device has enough battery to complete the test. The overall increase in data usage for the security app is calculated as the result in megabytes.

Data Set #1

This Data set consists of the following 30 popular websites.

1. <http://www.about.com>
2. <http://www.alibaba.com>
3. <http://www.amazon.com>
4. <http://www.apple.com>
5. <http://www.buzzfeed.com>
6. <http://www.cnet.com>
7. <http://www.craigslist.com>
8. <http://www.doubleclick.com>
9. <http://www.dslreports.com>
10. <http://www.eharmony.com>

11. <http://www.expedia.com>
12. <http://www.facebook.com>
13. <http://www.google.com>
14. <http://www.jcpenney.com>
15. <http://www.linkedin.com>
16. <http://www.mlslistings.com>
17. <http://www.netflix.com>
18. <http://www.noaa.gov>
19. <http://www.nordstrom.com>
20. <http://www.pitchfork.com>
21. <http://www.sca.com>
22. <http://www.scottrade.com>
23. <http://www.shacknews.com>
24. <http://www.sigalert.com>
25. <http://www.slashdot.com>
26. <http://www.target.com>
27. <http://www.t-mobile.com>
28. <http://www.wikipedia.com>
29. <http://www.webmd.com>
30. <http://www.yelp.com>

Data Set #2

This data set consists of the following 10 apps available on the Google App Store (as of July 31st 2015):

1. Netflix
2. Ebay
3. Esty
4. Airbnb
5. Fruit Ninja Free
6. 8 Ball Pool
7. Photo Grid
8. Spotify
9. OI File Manager
10. My Android

Data Set #3

This data set consists of a mix of picture files, music files, and documents files in the following amounts:

excel	425 KB
jpg	94.5 MB
mp3	420 MB
pdf	11.8 MB
png	28.5 MB
txt	12.6 MB
Video (.mov, .wmv, .avi, .mp4)	374 MB
wav	259 MB

word	8.03 MB
zip files	8.06 MB