

Consumer Security Products Performance Benchmarks (Edition 1)

Antivirus & Internet Security Windows 10

October 2019

Document: Consumer Security Products Performance Benchmarks (Edition 1)
Authors: J. Han, D. Wren
Company: PassMark Software
Date: 28 October 2019
Edition: 1
File: Consumer_Security_Products_Performance_Benchmarks_2020_Ed_1.docx

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	11
BENCHMARK 1 – BOOT TIME (SECONDS)	11
BENCHMARK 2 – SCAN TIME (SECONDS)	12
BENCHMARK 3 – USER INTERFACE LAUNCH TIME (MILLISECONDS)	13
BENCHMARK 4 – MEMORY USAGE DURING SYSTEM IDLE (MEGABYTES)	14
BENCHMARK 5 – MEMORY USAGE DURING INITIAL SCAN (MEGABYTES)	15
BENCHMARK 6 – BROWSE TIME (SECONDS).....	16
BENCHMARK 7 – EDGE LAUNCH TIME (MILLISECONDS)	17
BENCHMARK 8 – INSTALLATION TIME (SECONDS)	18
BENCHMARK 9 – INSTALLATION SIZE (MEGABYTES)	19
BENCHMARK 10 – FILE COPY, MOVE AND DELETE (SECONDS).....	20
BENCHMARK 11 –THIRD-PARTY APPLICATIONS INSTALLATION TIME (SECONDS)	21
BENCHMARK 12 – NETWORK THROUGHPUT (SECONDS)	22
BENCHMARK 13 – FILE FORMAT CONVERSION (SECONDS).....	23
BENCHMARK 14 – FILE COMPRESSION AND DECOMPRESSION (SECONDS).....	24
BENCHMARK 15 – FILE DOWNLOAD (SECONDS)	25
BENCHMARK 16 – PE SCAN TIME (SECONDS)	26
BENCHMARK 17 – FILE COPY DISK TO DISK (SECONDS)	27
BENCHMARK 18 – FILE COPY OVER NETWORK (SECONDS)	28
BENCHMARK 19 – PCMARK 8 HOME SCORE.....	29
BENCHMARK 20 – WORD DOCUMENT LAUNCH AND OPEN TIME (MILLISECONDS)	30
BENCHMARK 21 – RUN EXCEL MACRO (SECONDS)	31
BENCHMARK 22 – SAVE WORD DOCUMENT TO PDF (SECONDS).....	32
BENCHMARK 23 – USB 3.0 FILE COPY (SECONDS)	33
DISCLAIMER AND DISCLOSURE	34
CONTACT DETAILS	34
APPENDIX 1 – TEST ENVIRONMENT	35
APPENDIX 2 – METHODOLOGY DESCRIPTION	36

Revision History

Rev	Revision History	Date
Edition 1	Initial version of report	28 October 2019

References

Ref #	Document	Author	Date
1	What Really Slows Windows Down (URL)	O. Warner, The PC Spy	2006

Executive Summary

PassMark Software® conducted objective performance testing on ten (10) security products. This report presents our results and findings as a result of performance benchmark testing conducted on these products.

For more details on which versions were tested, please see the section "[Products and Versions](#)".

Testing included twenty-three (23) performance metrics. These performance metrics are as follows:

- Boot Time
- Scan Time
- User Interface Launch Time
- Memory Usage during System Idle
- Memory Usage during Initial Scan
- Browse Time
- Edge Launch Time
- Installation Time
- Installation Size
- File Copy, Move and Delete
- Third-Party Applications Installation Time
- Network Throughput
- File Format Conversion
- File Compression and Decompression
- PE Scan Time
- File Copy Disk to Disk
- File Copy Over Network
- File Download
- PCMark 8 Home Score
- Word Document Launch and Open Time
- Run Excel Macro
- Save Word Document as PDF
- USB 3.0 File Copy

Overall Score

PassMark Software assigned every product a score depending on its ranking in each metric compared to other products in the same category.

Security Software

In the following table the highest possible score attainable is 230; in a hypothetical situation where a product has attained first place in all 23 metrics. Security products have been ranked by their overall scores:

Product Name	Overall Score
Norton Security	191
ESET Internet Security	161
Kaspersky Internet Security	159
Windows Defender	128
AVG Internet Security	115
Trend Micro Internet Security	115
Trend Micro Virus Buster	112
Avast Internet Security	111
Bitdefender Internet Security	84
SourceNext ZERO Super Security	79

Products and Versions

The names and versions of the ten security products tested are as follows:

Manufacturer	Product Name	Release Year	Product Version	Date Tested
Symantec Corporation	Norton Security	2019	22.19.8.65	Sep 2019
Avast Software s.r.o.	AVG Internet Security	2019	19.6.3098	Jul 2019
SourceNext Corporation	SourceNext ZERO Super Security	2019	23.0.19.85	Jul 2019
ESET, spol. s r.o.	ESET Internet Security	2019	12.1.34.0	Jul 2019
Trend Micro Inc.	Trend Micro Internet Security	2019	16.0.1146	Oct 2019
Trend Micro Inc.	Trend Micro Virus Buster	2019	15.0.1231	Aug 2019
Kaspersky Lab	Kaspersky Internet Security	2019	20.0.14.1085(c)	Jul 2019
Avast Software s.r.o.	Avast Internet Security	2019	19.6.2383	Jul 2019
Microsoft Corporation	Windows Defender	2019	4.18.1909.6	Oct 2019
Bitdefender	Bitdefender Internet Security	2019	24.0.6.31	Sep 2019

Performance Metrics Summary

We have selected a set of objective metrics which provide a comprehensive and realistic indication of the areas in which a security product may impact system performance for end users. Our metrics test the impact of the security software on common tasks that end-users would perform on a daily basis.

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 – Boot Time

This metric measures the amount of time taken for the machine to boot into the operating system. Security software is generally launched at Windows startup, adding an additional amount of time and delaying the startup of the operating system. Shorter boot times indicate that the application has had less impact on the normal operation of the machine.

Benchmark 2 – Scan Time

All security solutions have functionality designed to detect viruses and various other forms of malware by scanning files on the system. This metric measures the amount of time required to scan a set of clean files. Our sample file set comprises a total file size of 982 MB and is made up of files that would typically be found on end-user machines, such as media files, system files and Microsoft Office documents.

Benchmark 3 – User Interface Launch Time

This metric provides an objective indication as to how responsive a security product appears to the user, by measuring the amount of time it takes for the user interface of the security software to launch from Windows. To allow for caching effects by the operating system, both the initial launch time and the subsequent launch times are measured. Our final result is an average of these two measurements.

Benchmark 4 – Memory Usage during System Idle

This metric measures the amount of memory (RAM) used by the product while the machine and security software are in an idle state. The total memory usage is calculated by identifying all security software processes and the amount of memory used by each process.

The amount of memory used while the machine is idle provides a good indication of the amount of system resources being consumed by the security software on a permanent basis. Better performing products occupy less memory while the machine is idle.

Benchmark 5 – Memory Usage during Initial Scan

This metric measures the amount of memory (RAM) used by the product during an initial security scan. The total memory usage is calculated by identifying all security software processes and the amount of memory used by each process during the scan.

Benchmark 6 – Browse Time

It is common behavior for security products to scan data for malware as it is downloaded from the internet or intranet. This behavior may negatively impact browsing speed as products scan web content for malware. This metric measures the time taken to browse a set of popular internet sites to consecutively load from a local server in a user's browser window.

Benchmark 7 – Edge Launch Time

This metric is one of many methods to objectively measure how much a security product impacts on the responsiveness of the system. This metric measures the amount of time it takes to launch the user interface of Microsoft Edge. To allow for caching effects by the operating system, both the initial launch time and the subsequent launch times are measured. Our final result is an average of these two measurements.

Benchmark 8 – Installation Time

The speed and ease of the installation process will strongly influence the user's first impression of the security software. This test measures the minimum installation time required by the security software to be fully functional and ready for use by the end user. Lower installation times represent security products which are quicker for a user to install.

Benchmark 9 – Installation Size

In offering new features and functionality to users, security software products tend to increase in size with each new release. Although new technologies push the size limits of hard drives each year, the growing disk space requirements of common applications and the increasing popularity of large media files (such as movies, photos and music) ensure that a product's installation size will remain of interest to home users.

This metric aims to measure a product's total installation size. This metric is defined as the total disk space consumed by all new files added during a product's installation.

Benchmark 10 – File Copy, Move and Delete

This metric measures the amount of time taken to move, copy and delete a sample set of files. The sample file set contains several types of file formats that a Windows user would encounter in daily use. These formats include documents (e.g. Microsoft Office documents, Adobe PDF, Zip files, etc.), media formats (e.g. images, movies and music) and system files (e.g. executables, libraries, etc.).

Benchmark 11 –Third-Party Applications Installation Time

This metric measures the amount of time taken to install and uninstall third-party programs. The installation speed of third-party applications may be impacted by security behavior such as heuristics or real time malware scanning.

Benchmark 12 – Network Throughput

The metric measures the amount of time taken to download a variety of files from a local server using the Hypertext Transfer Protocol (HTTP), which is the main protocol used on the web for browsing, linking and data transfer. Files used in this test include file formats that users would typically download from the web, such as images, archives, music files and movie files.

Benchmark 13 – File Format Conversion

This test measures the amount of time taken to convert an MP3 file to a WAV and subsequently, convert the same MP3 file to a WMA format.

Benchmark 14 – File Compression and Decompression

This metric measures the amount of time taken to compress and decompress different types of files. File formats used in this test include documents, movies and images.

Benchmark 15 – File Download

This test measures the amount of time taken to download a set of setup files from a local server using the Hypertext Transfer Protocol (HTTP). The data set comprises a total file size of 290MB, and the formats used include executables and Microsoft installation packages.

Benchmark 16 – PE Scan Time

All security solutions have functionality designed to detect viruses and various other forms of malware by scanning files on the system. This metric measures the amount of time required to scan a set of PE (Portable Executable) files. Our sample file set comprises a total file size of 2.03GB and consisted of .exe (329MB), .dll (920MB) and .sys files (827MB).

Benchmark 17 – File Copy Disk to Disk

This test measures the amount of time taken to copy files between two local drives. The data set comprises a total file size of 5.44GB, and the formats used include documents, movies, images and executables.

Benchmark 18 – File Copy Over Network

This test measures the amount of time taken to copy files from a local drive to a local server. The data set comprises a total file size of 5.44GB, and the formats used included documents, movies, images and executables.

Benchmark 19 – PCMark 8 Home Score

This test measures the overall performance as an overall score over a range of areas, including memory, video, gaming, music, communications, productivity and HDD performance.

Benchmark 20– Word Document Launch and Open Time

This test measures how much security software impacts on the responsiveness and performance of the system. This metric measures the amount of time it takes to open a large, mixed media document with Microsoft Word. To allow for caching effects by the operating system, both the initial launch time and the subsequent launch times are measured. Our final result is an average of these two measurements.

Benchmark 21 – Run Excel Macro

This test measures the amount of time taken to open an Excel document and run a macro to perform a range of operations. The test is run five times with a reboot in between each run. Our final result is taken as an average of these five measurements.

Benchmark 22 – Save Word Document as PDF

This test measures the amount of time taken to open a large Word document and save it in PDF format. The test is run five times with a reboot in between each run. Our final result is taken as an average of these five measurements.

Benchmark 23 – USB 3.0 File Copy

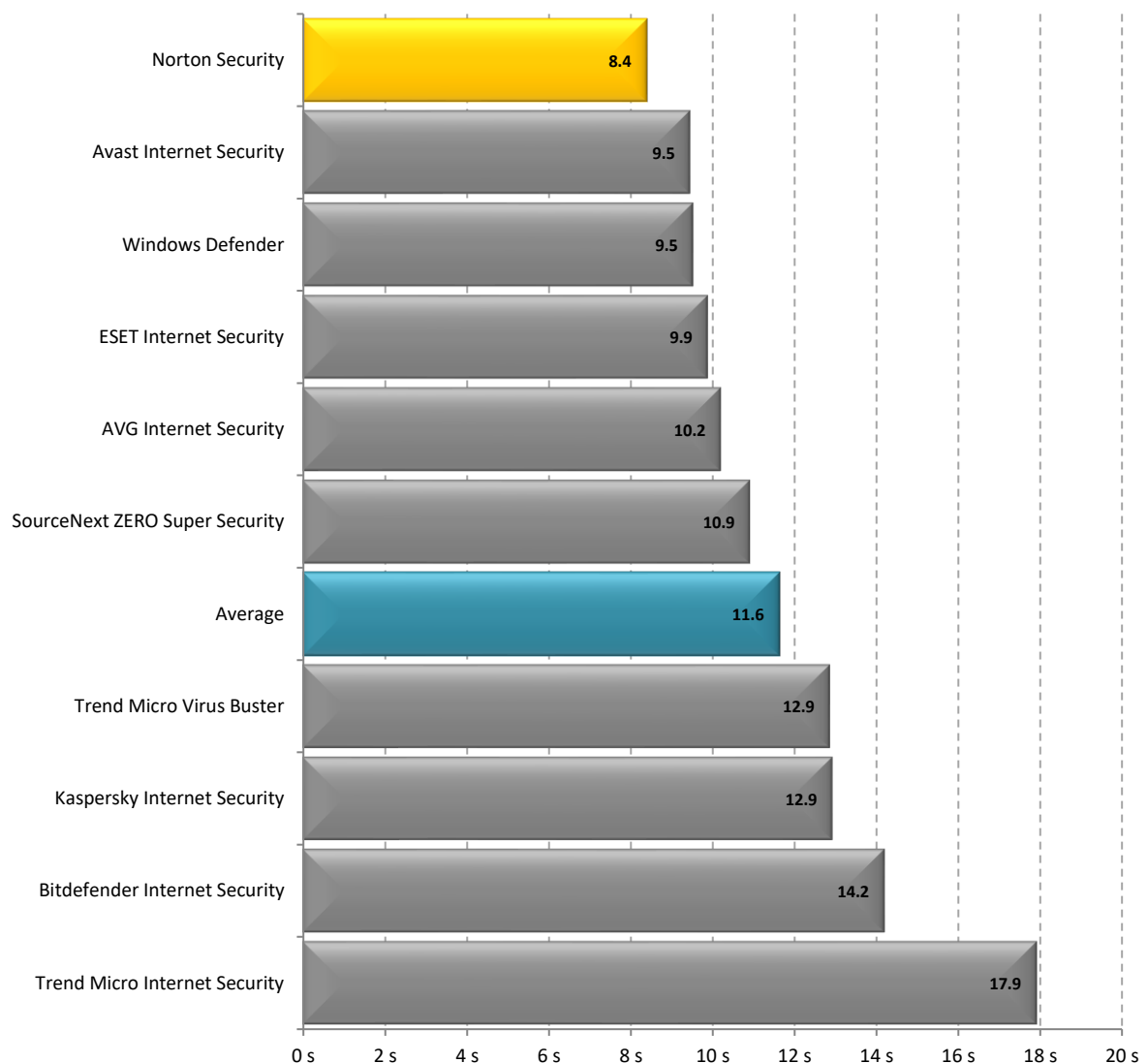
This test measures the amount of time taken to copy files from a USB 3.0 drive to a local disk. The data set comprises a total file size of 6.06GB, and the formats used include documents, movies, system files and executables.

Test Results

In the following charts, we have highlighted the results we obtained for Norton Security in yellow. The average has also been highlighted in blue for ease of comparison.

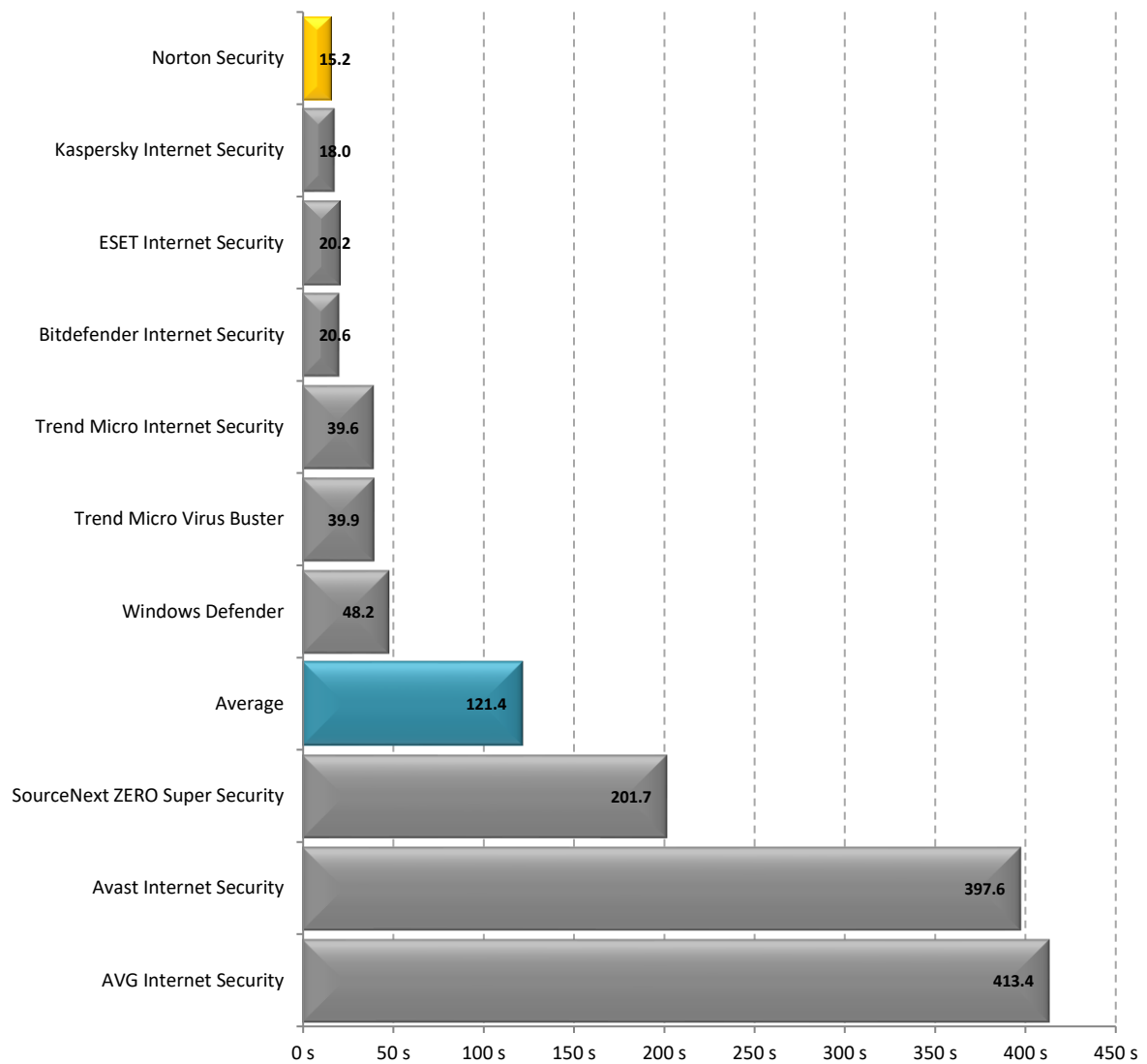
Benchmark 1 – Boot Time (seconds)

The following chart compares the average time taken for the system to boot (from a sample of five fast boots) for each Internet Security product tested. Products with lower boot times are considered better performing products in this category.



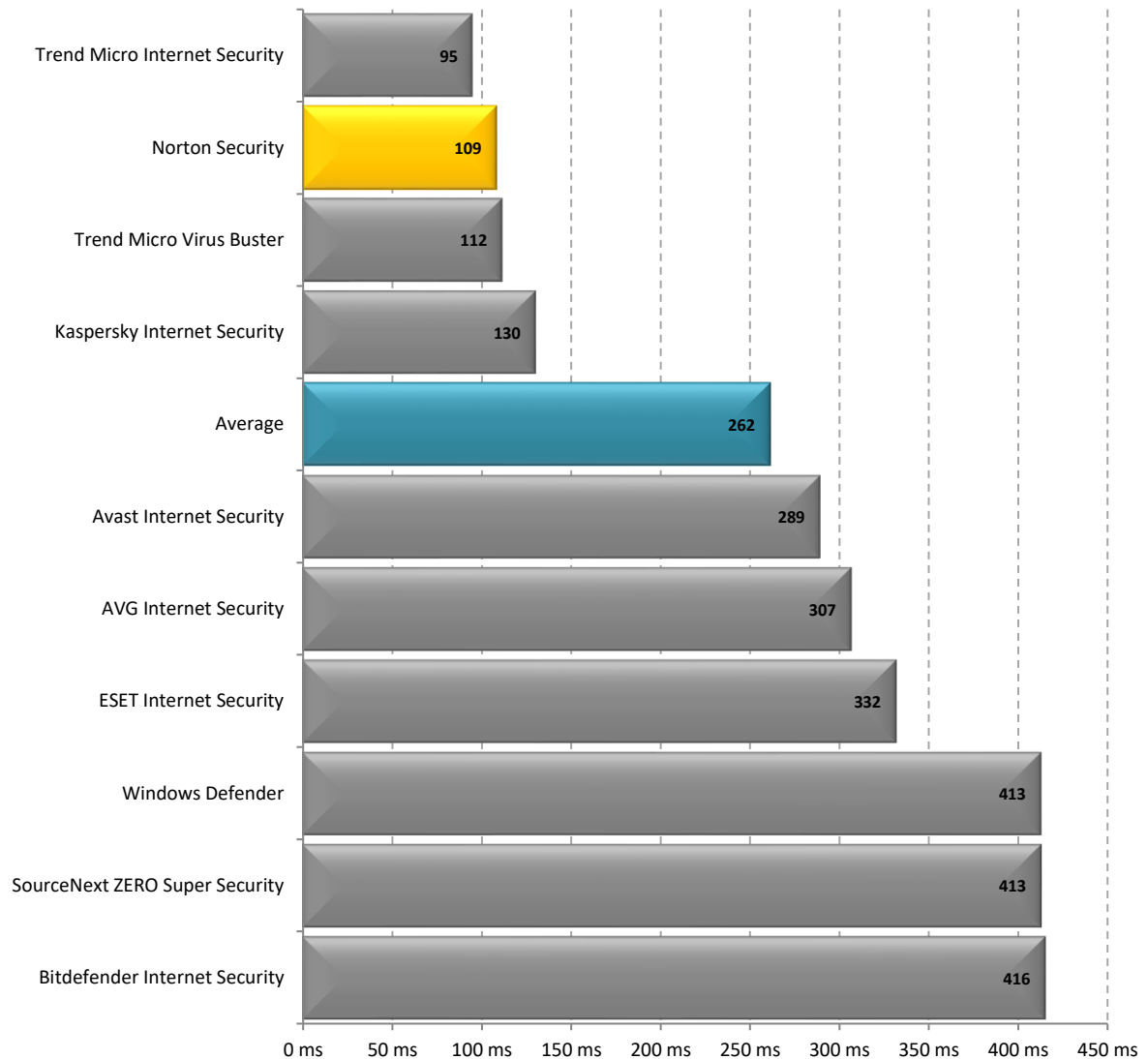
Benchmark 2 – Scan Time (seconds)

The following chart compares the average time taken to scan a set of 6159 files (totaling 982 MB) for each Internet Security product tested. This time is calculated by averaging the initial (Run 1) and subsequent (Runs 2-5) scan times. Products with lower scan times are considered better performing products in this category.



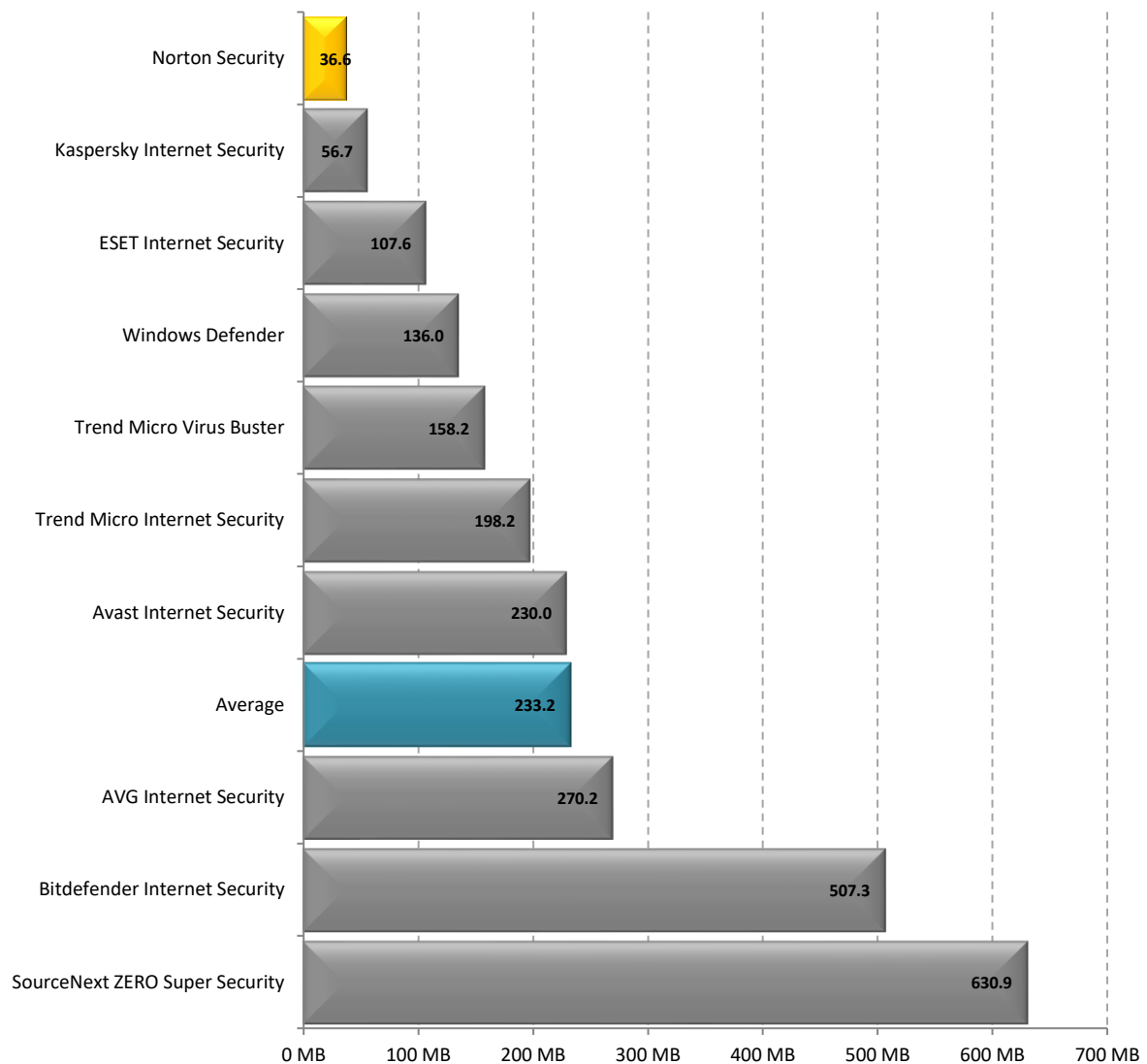
Benchmark 3 – User Interface Launch Time (milliseconds)

The following chart compares the average time taken to launch a product's user interface. Products with lower launch times are considered better performing products in this category.



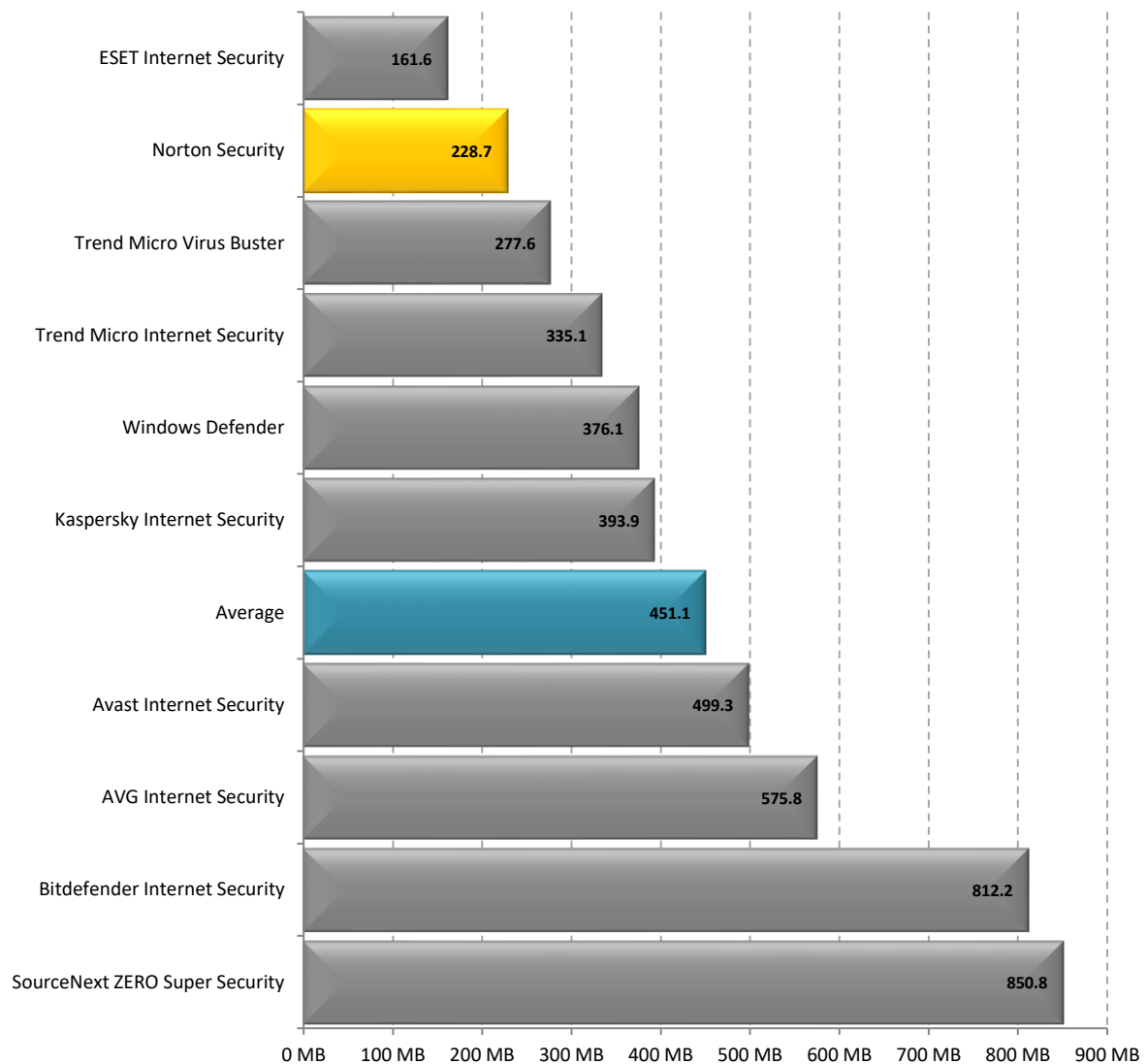
Benchmark 4 – Memory Usage during System Idle (megabytes)

The following chart compares the average amount of RAM in use by an Internet Security product during a period of system idle. This average is taken from a sample of ten memory snapshots taken at roughly 60 seconds apart after reboot. Products with lower idle RAM usage are considered better performing products in this category.



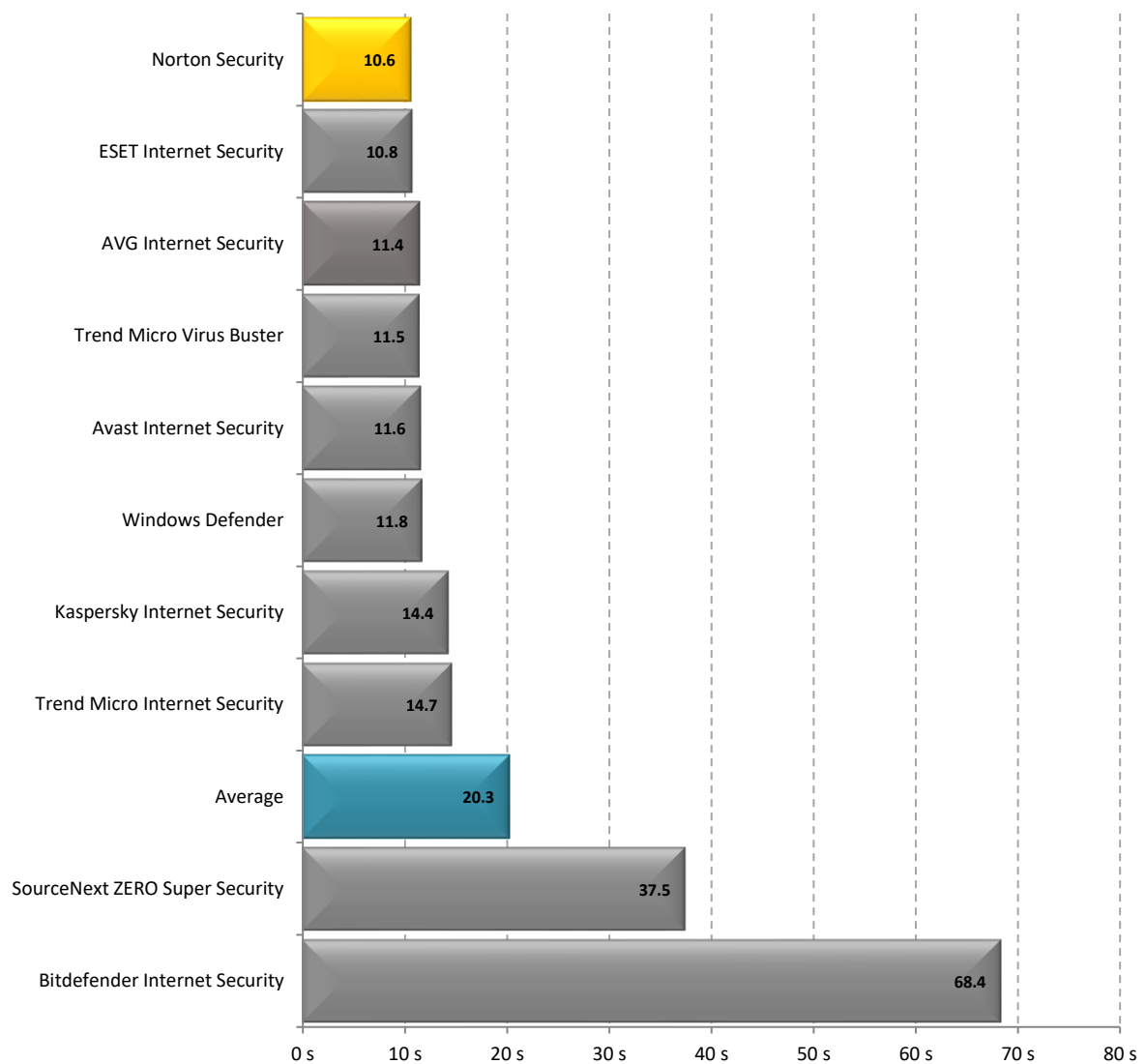
Benchmark 5 – Memory Usage during Initial Scan (megabytes)

The following chart compares the average amount of RAM in use by an Internet Security product during a scan. This average is taken from a sample of ten memory snapshots taken at roughly 12 seconds apart. Products with lower idle RAM usage are considered better performing products in this category.



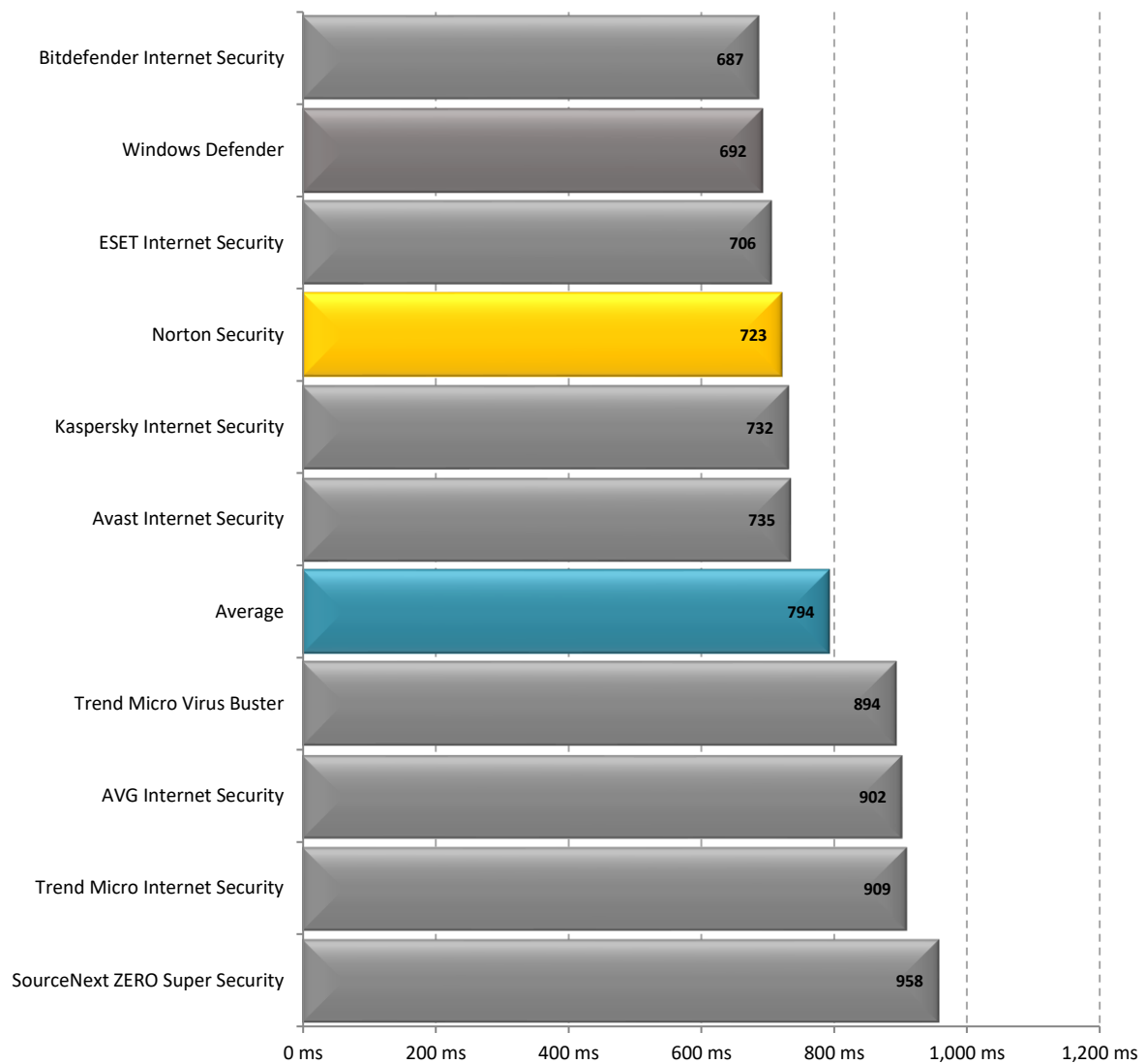
Benchmark 6 – Browse Time (seconds)

The following chart compares the average time taken for the default browser Microsoft Edge to successively load a set of popular websites through the local area network from a local server machine. Products with lower browse times are considered better performing products in this category.



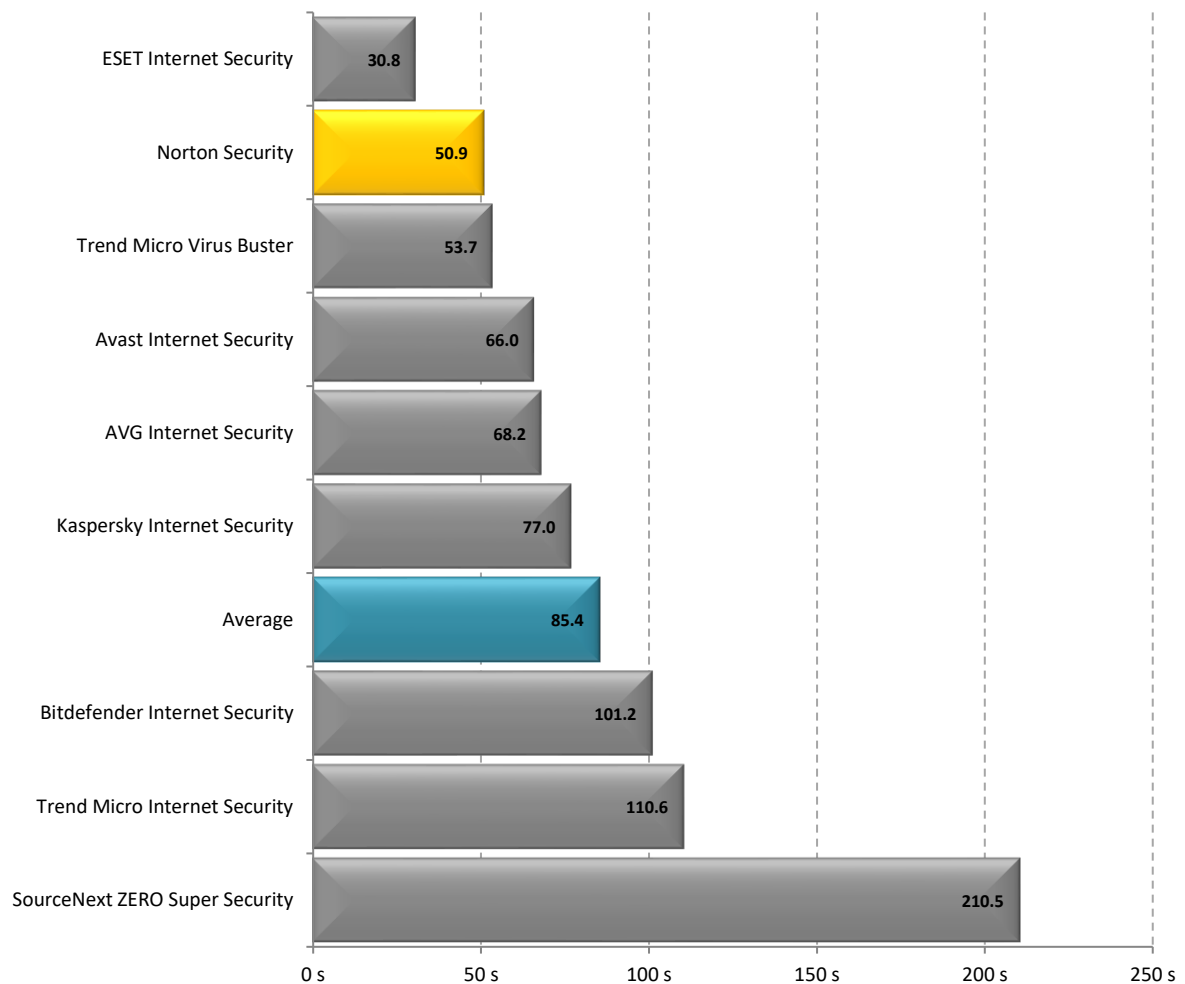
Benchmark 7 – Edge Launch Time (milliseconds)

The following chart compares the average launch times of the Microsoft Edge after rebooting the machine for each Internet Security product we tested. Products with lower launch times are considered better performing products in this category.



Benchmark 8 – Installation Time (seconds)

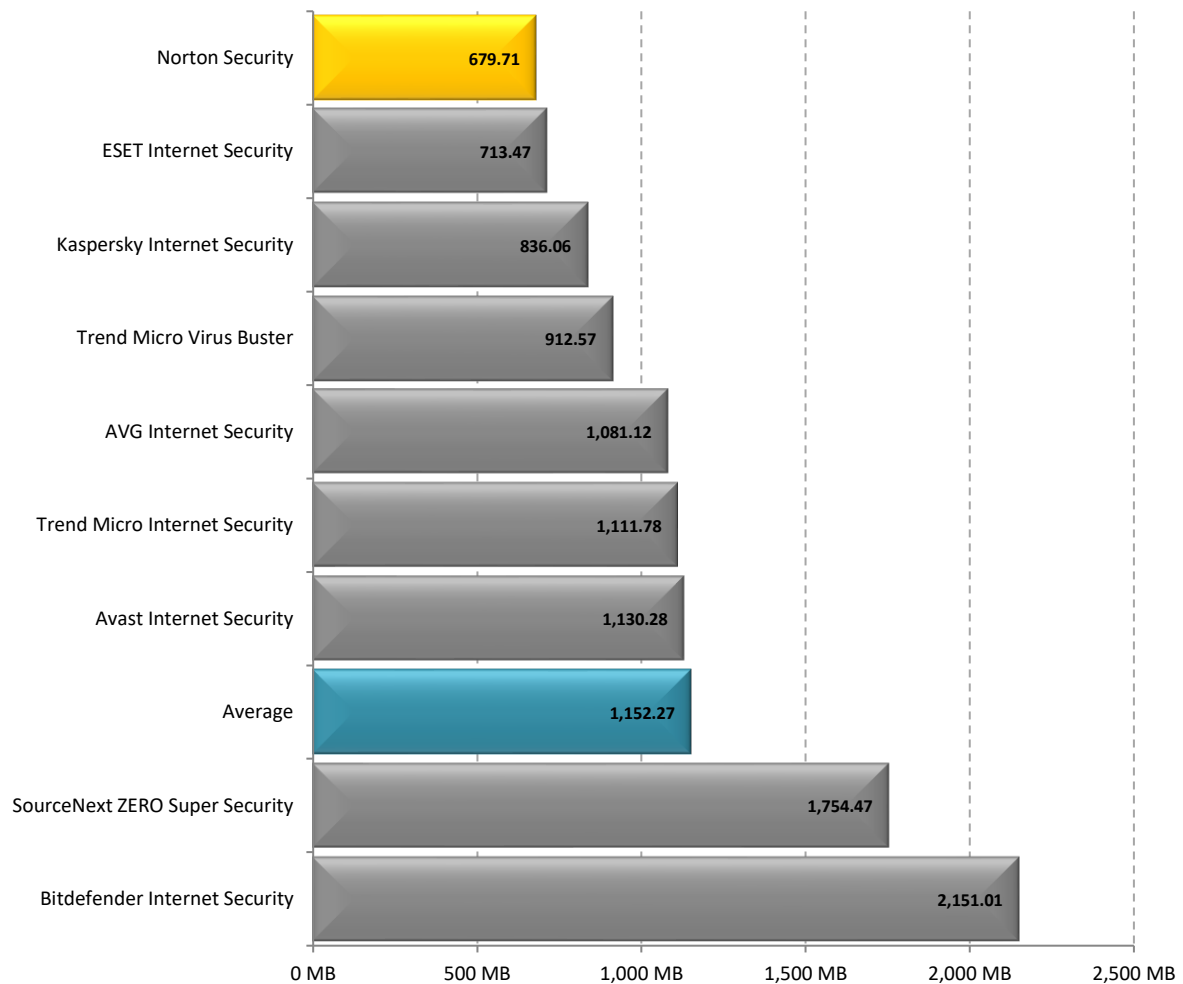
The following chart compares the minimum time it takes for security products to be fully functional and ready for use by the end user. Products with lower installation times are considered better performing products in this category.¹



¹ Windows Defender was excluded from this test as it is built in to Windows 10 (Does not require installation).

Benchmark 9 – Installation Size (megabytes)

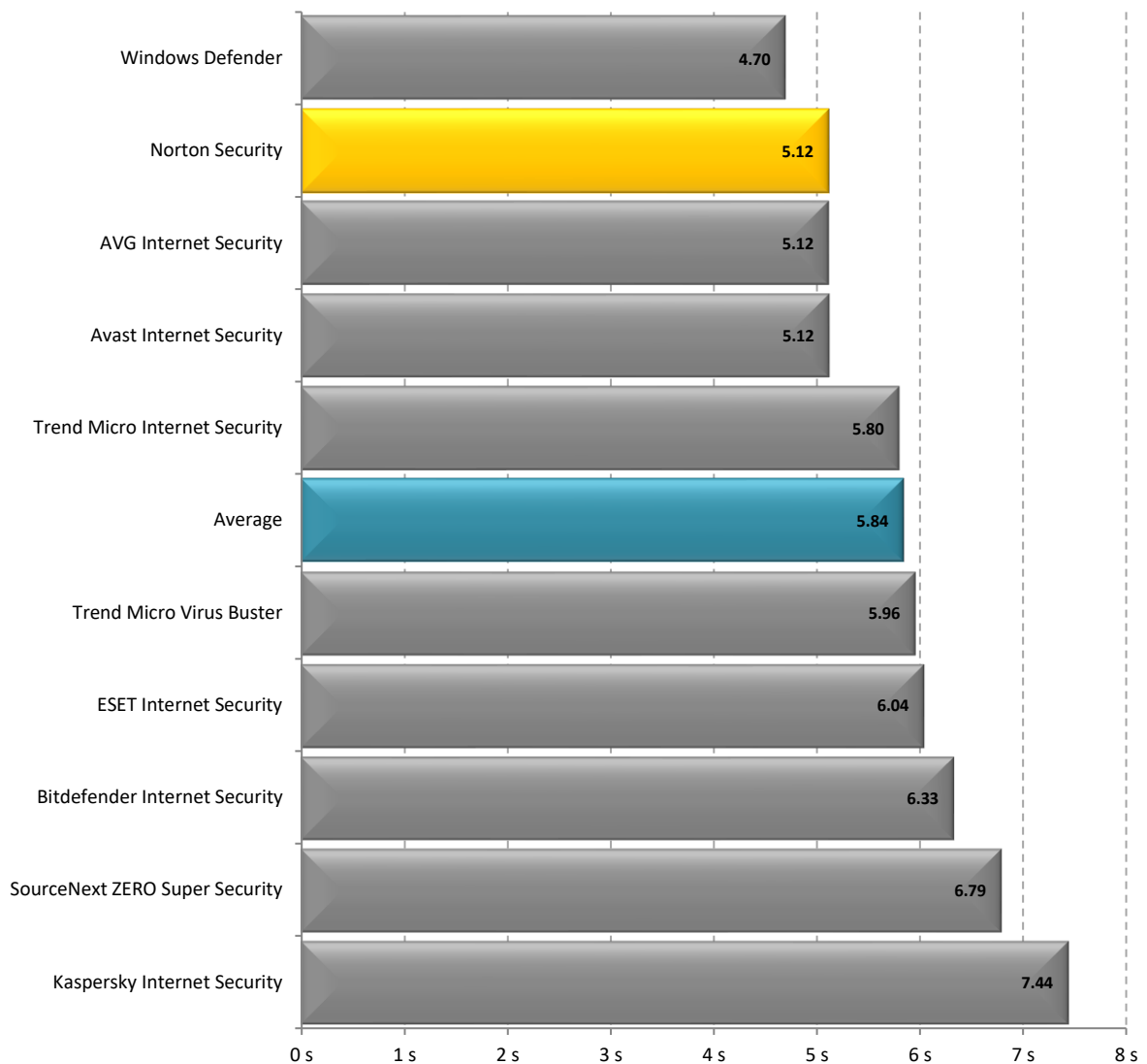
The following chart compares the total size of files added during the installation of security products. Products with lower installation sizes are considered better performing products in this category.²



² Windows Defender was excluded from this test as it is built in to Windows 10 (Does not require installation).

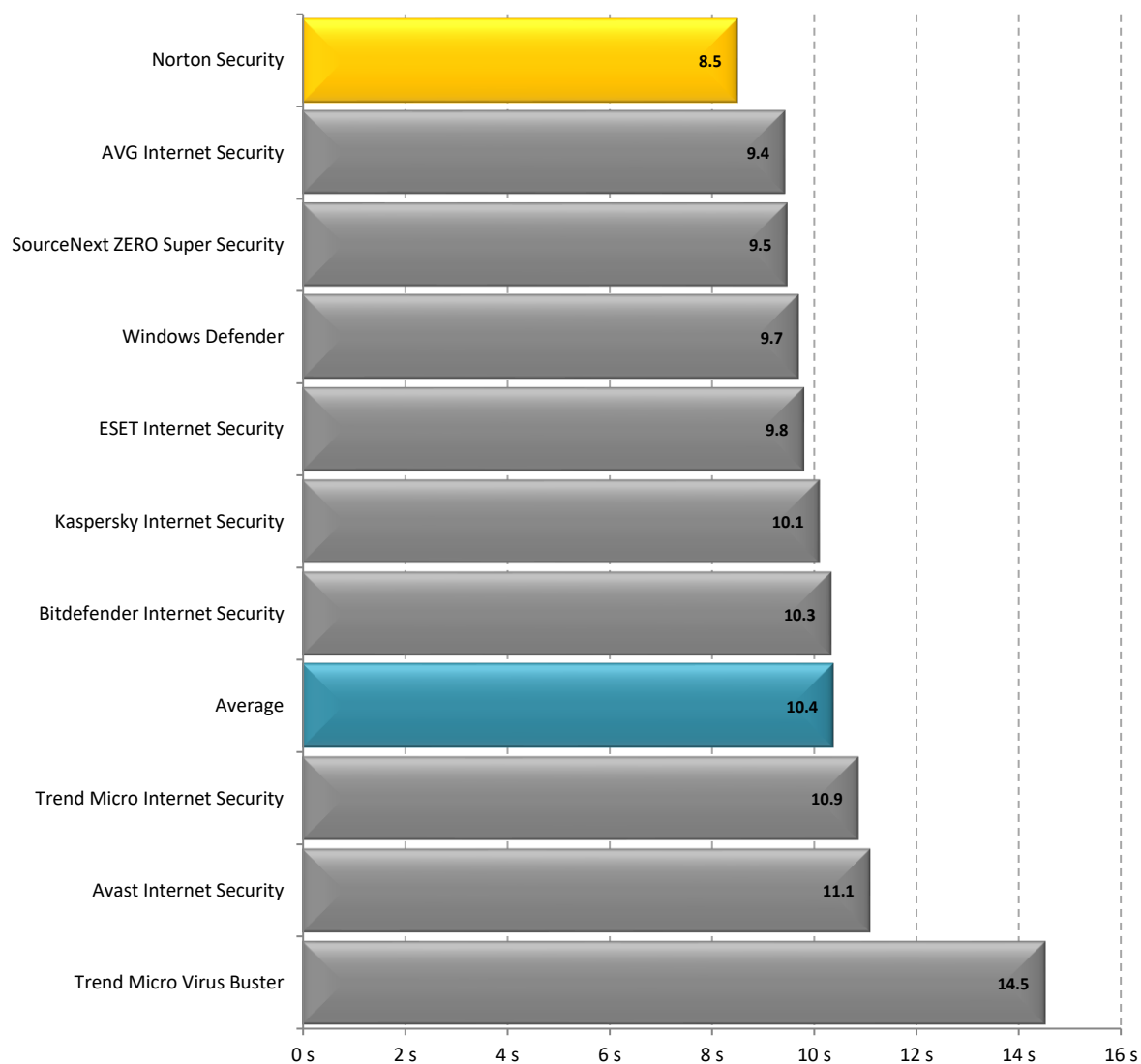
Benchmark 10 – File Copy, Move and Delete (seconds)

The following chart compares the average time taken to copy, move and delete several sets of sample files for each Internet Security product tested. Products with lower times are considered better performing products in this category.



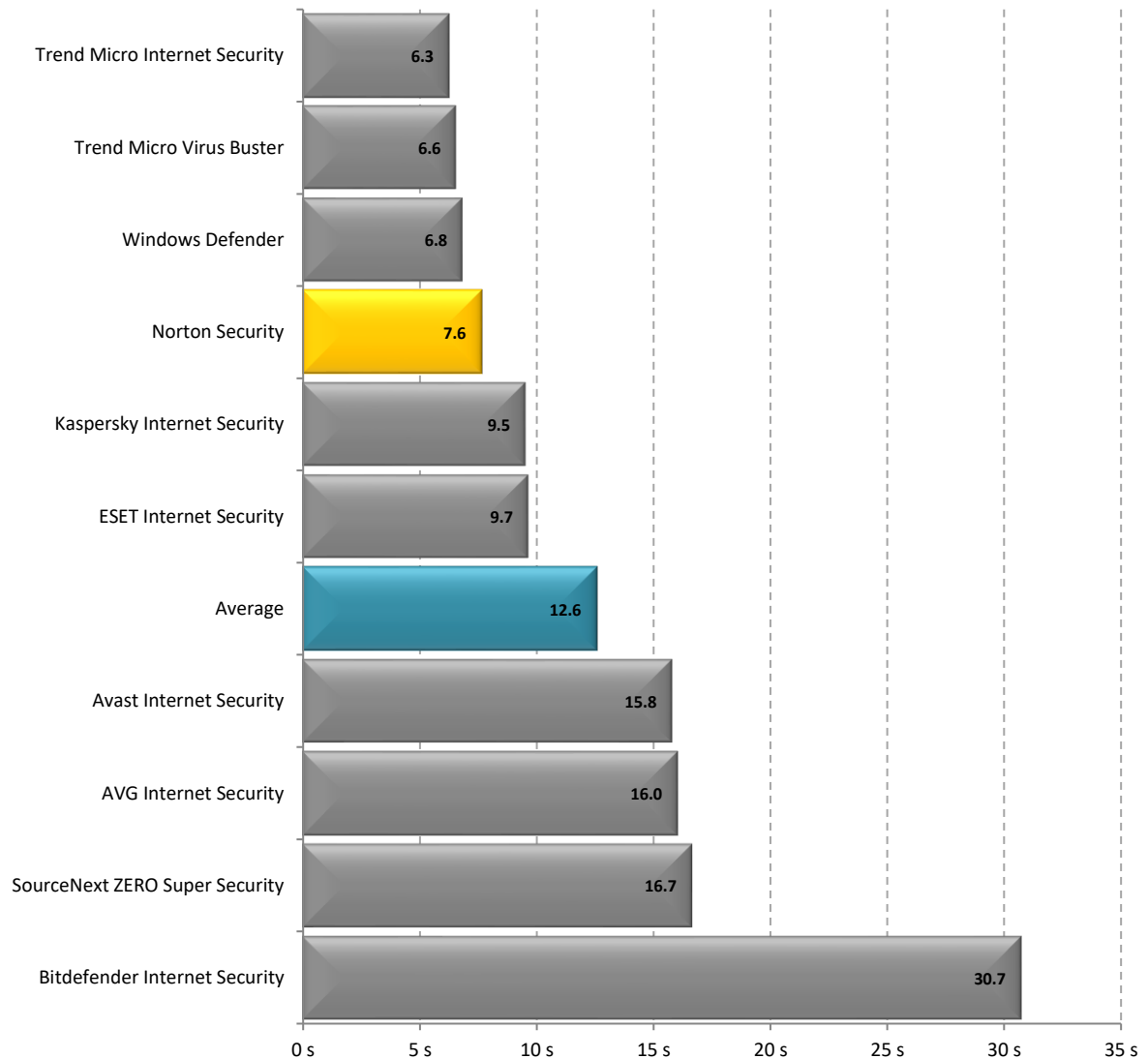
Benchmark 11 –Third-Party Applications Installation Time (seconds)

The following chart compares the average time taken to install 3 different third-party applications for each Internet Security product tested. Products with lower times are considered better performing products in this category.



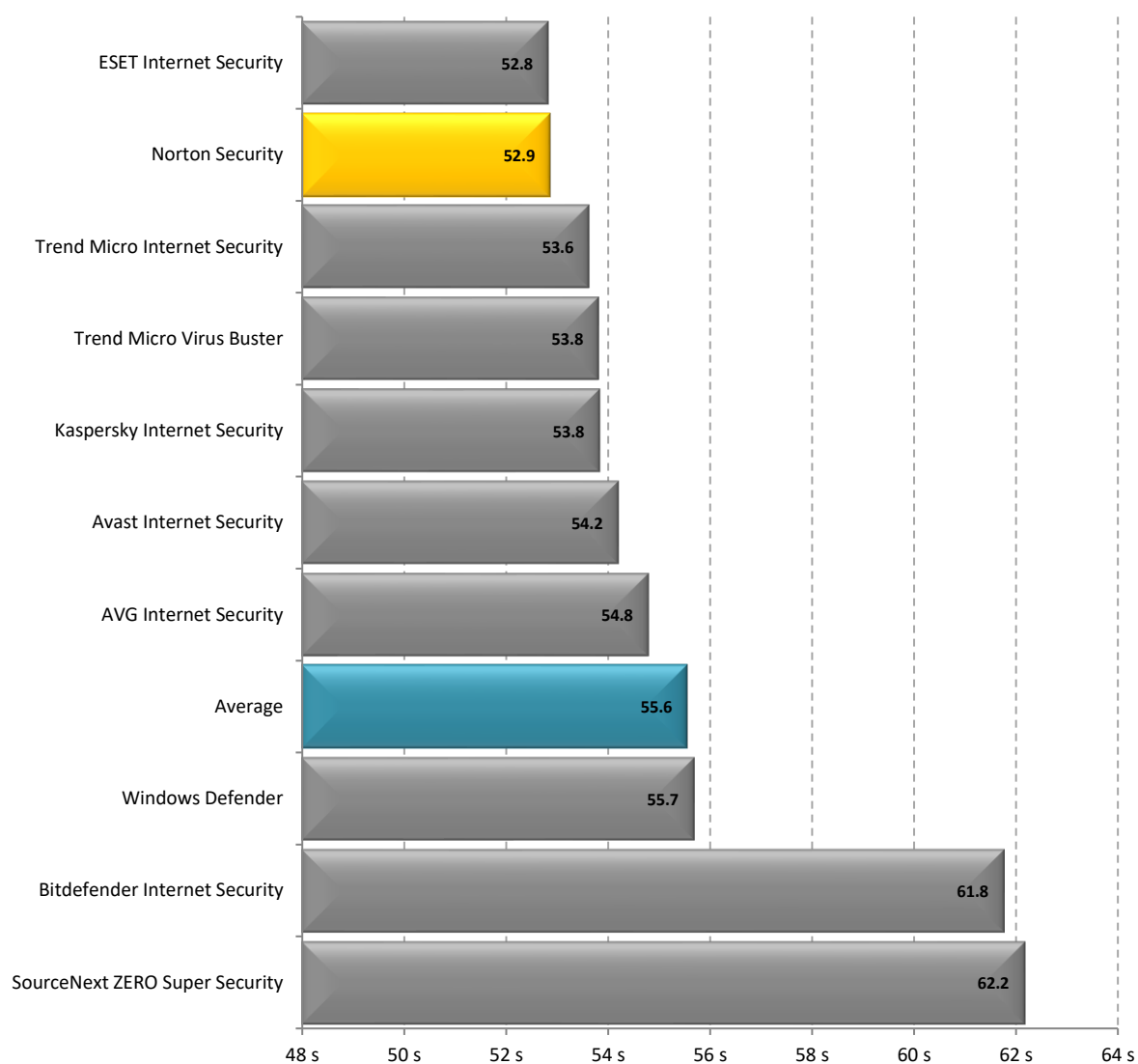
Benchmark 12 – Network Throughput (seconds)

The following chart compares the average time to download a sample set of common file types for each Internet Security product tested. Products with lower times are considered better performing products in this category.



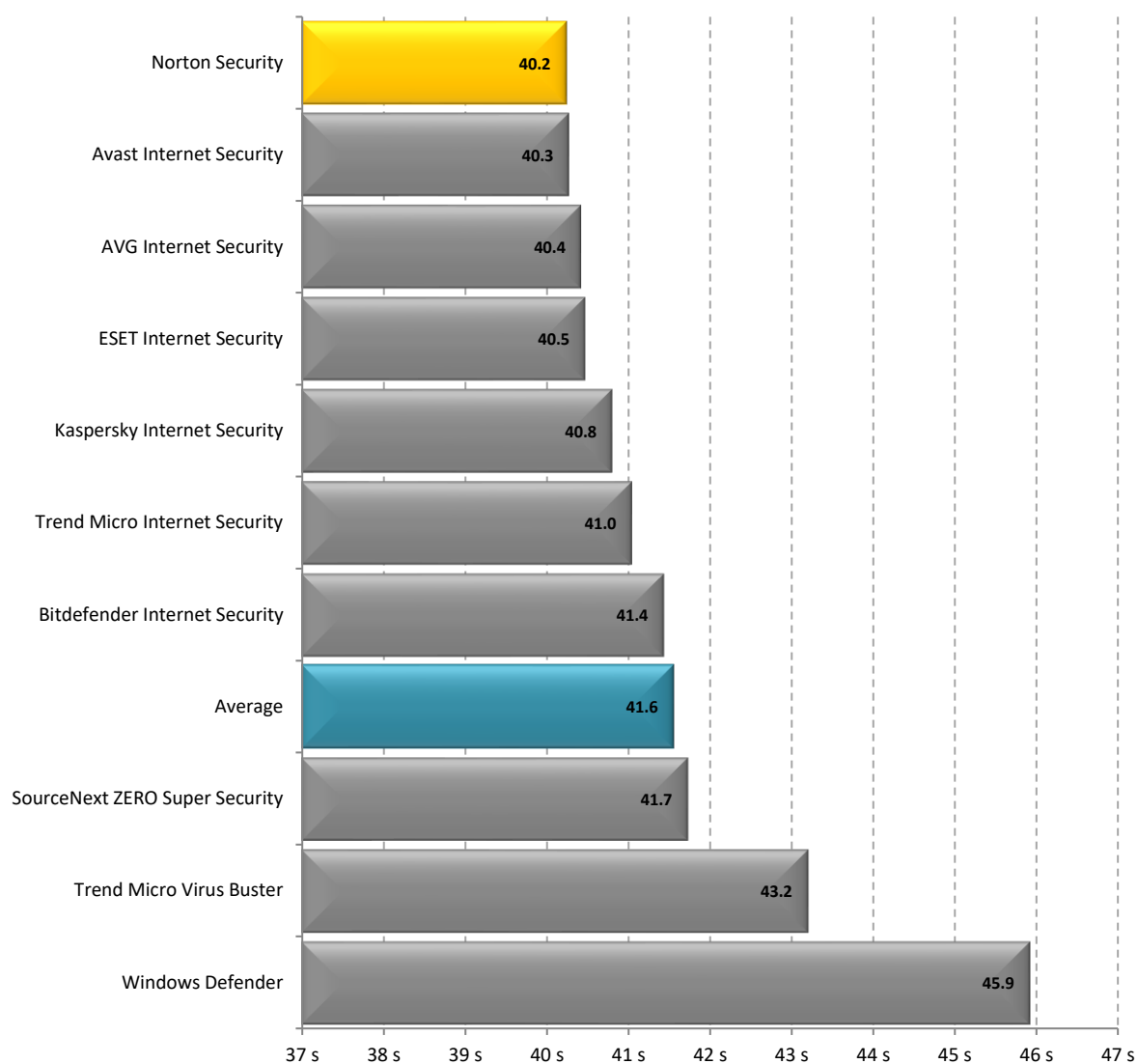
Benchmark 13 – File Format Conversion (seconds)

The following chart compares the average time it takes for five sample files to be converted from one file format to another (MP3 ↔ WMA, MP3 ↔ WAV) for each Internet Security product tested. Products with lower times are considered better performing products in this category.



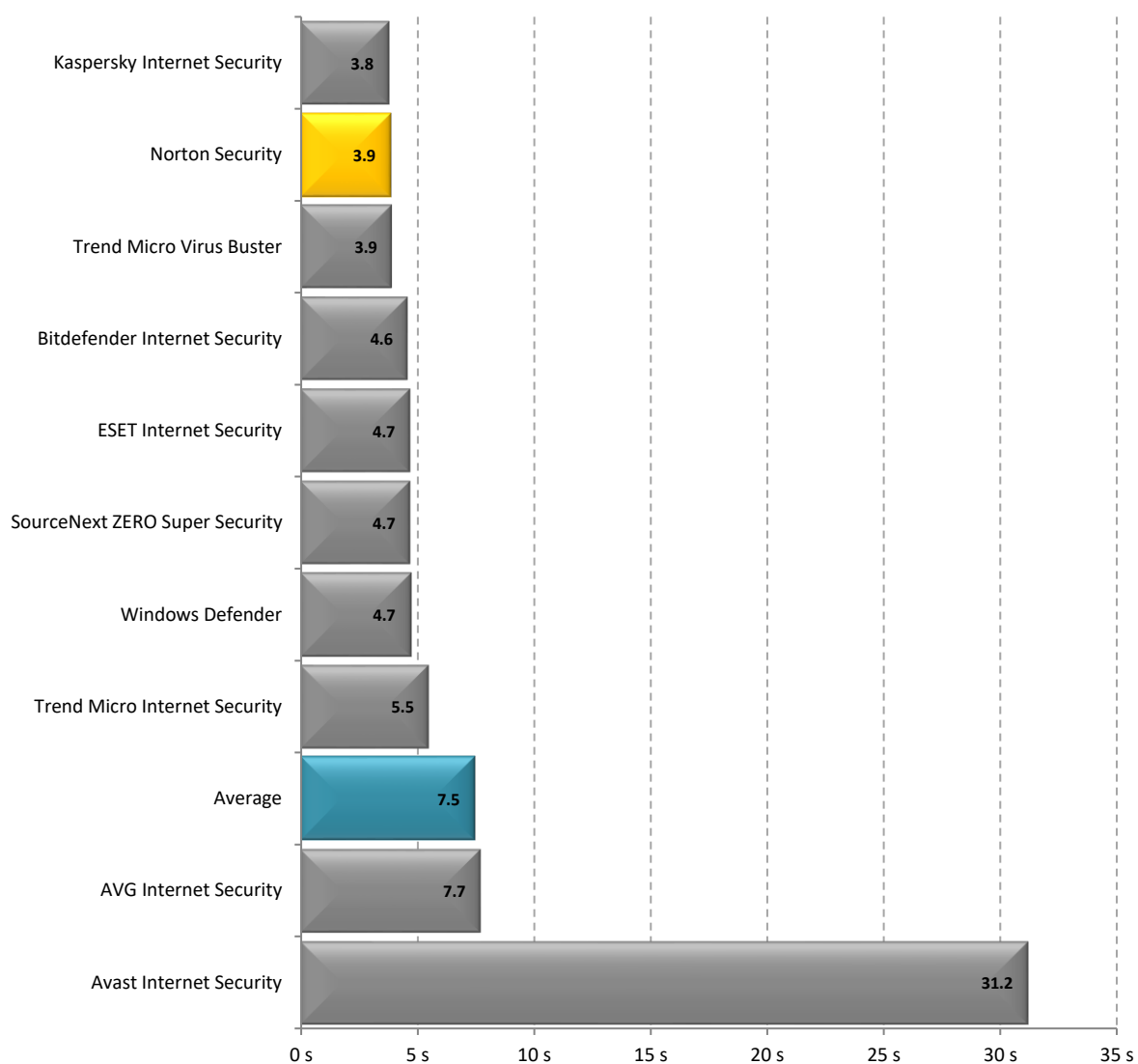
Benchmark 14 – File Compression and Decompression (seconds)

The following chart compares the average time it takes for sample files to be compressed and decompressed for each Internet Security product tested. Products with lower times are considered better performing products in this category.



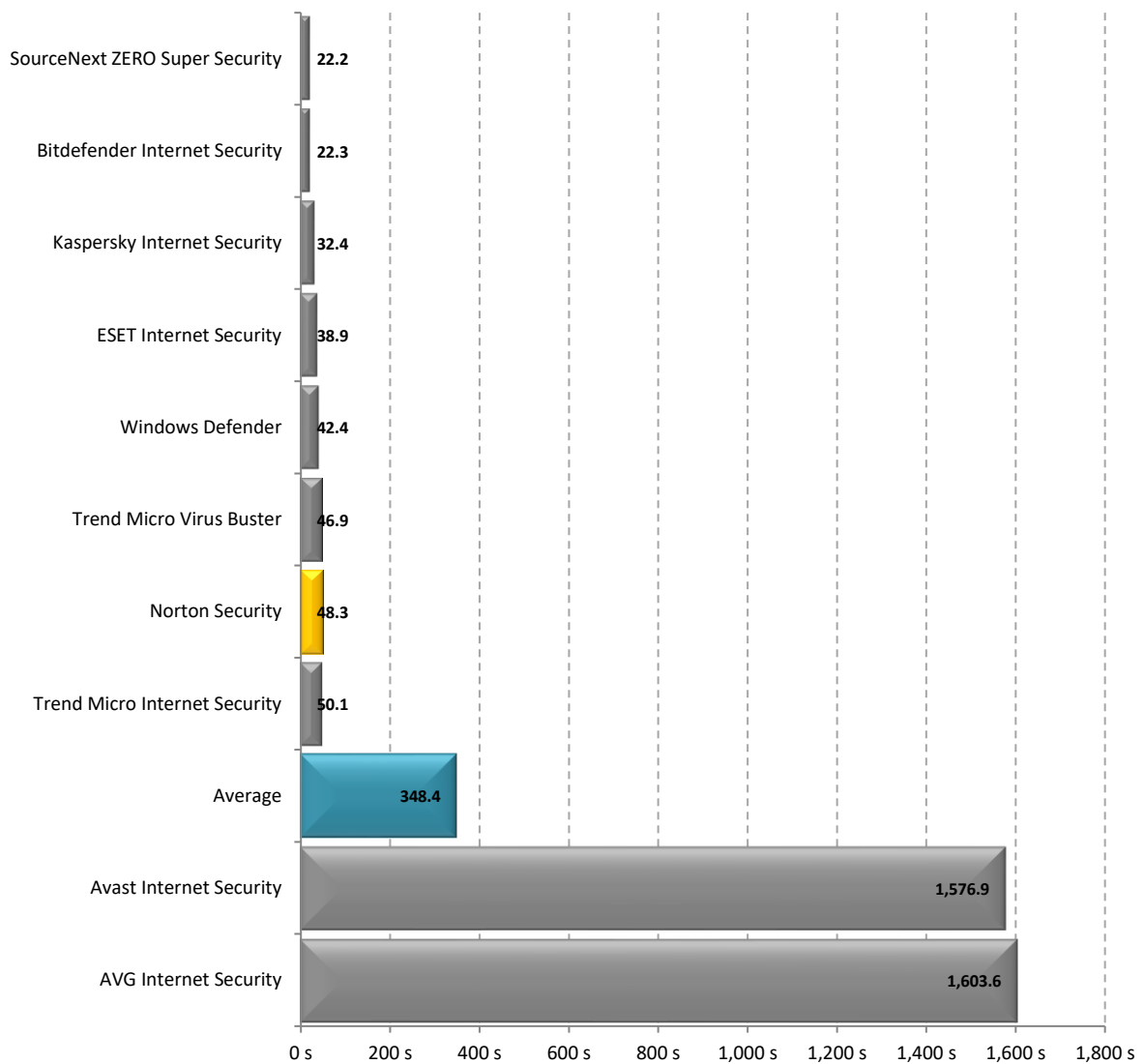
Benchmark 15 – File Download (seconds)

The following chart compares the average time taken to download a set of setup files from a local server. The test was performed 5 times, and the average of all 5 runs was taken as the result. Products with lower times are considered better performing products in this category.



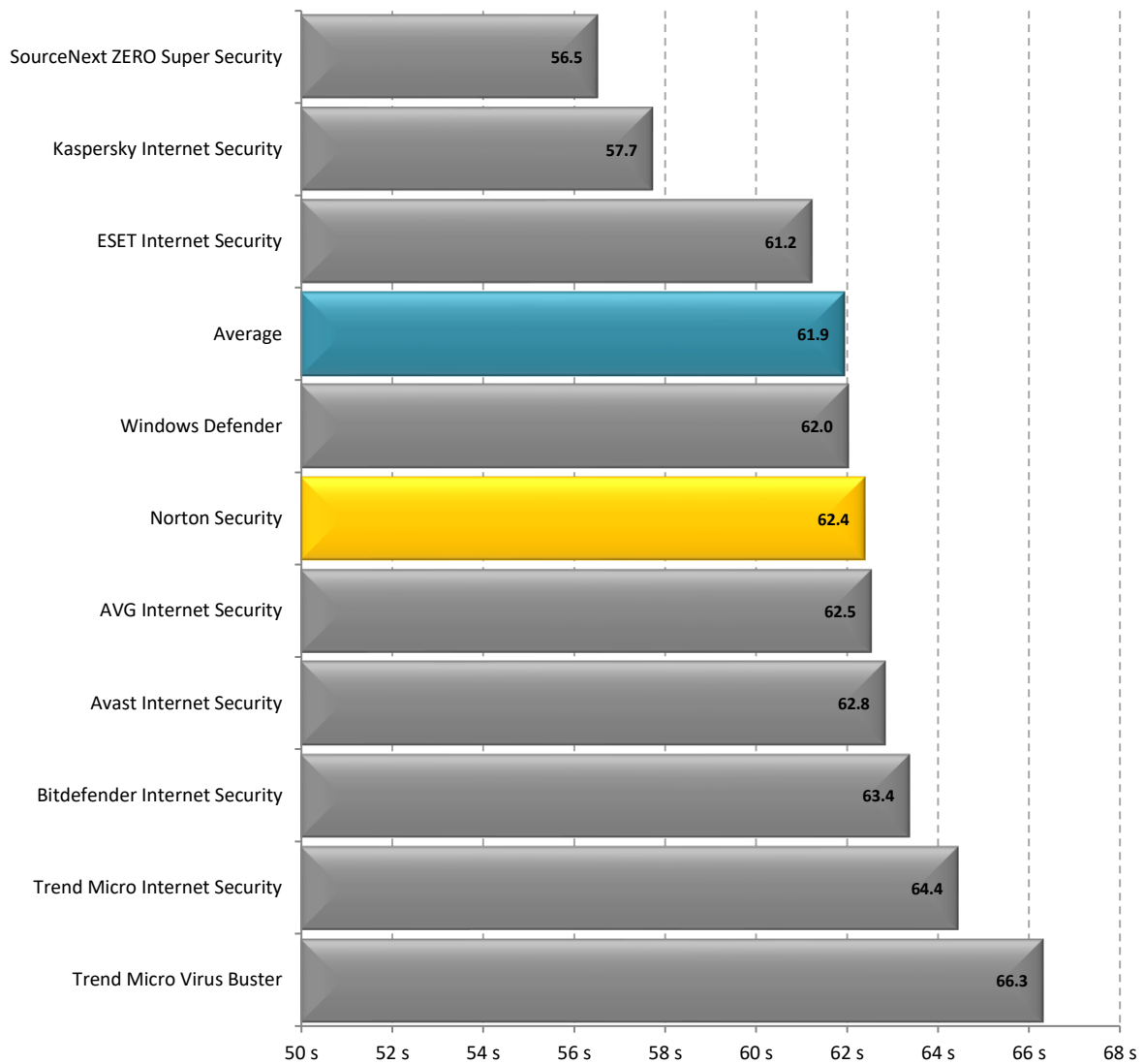
Benchmark 16 – PE Scan Time (seconds)

The following chart compares the average time taken to scan a set of 6,351 portable executable files (totaling 2,076 MB) for each Internet Security product tested. This time is calculated by averaging the initial (Run 1) and subsequent (Runs 2-5) scan times. Products with lower scan times are considered better performing products in this category.



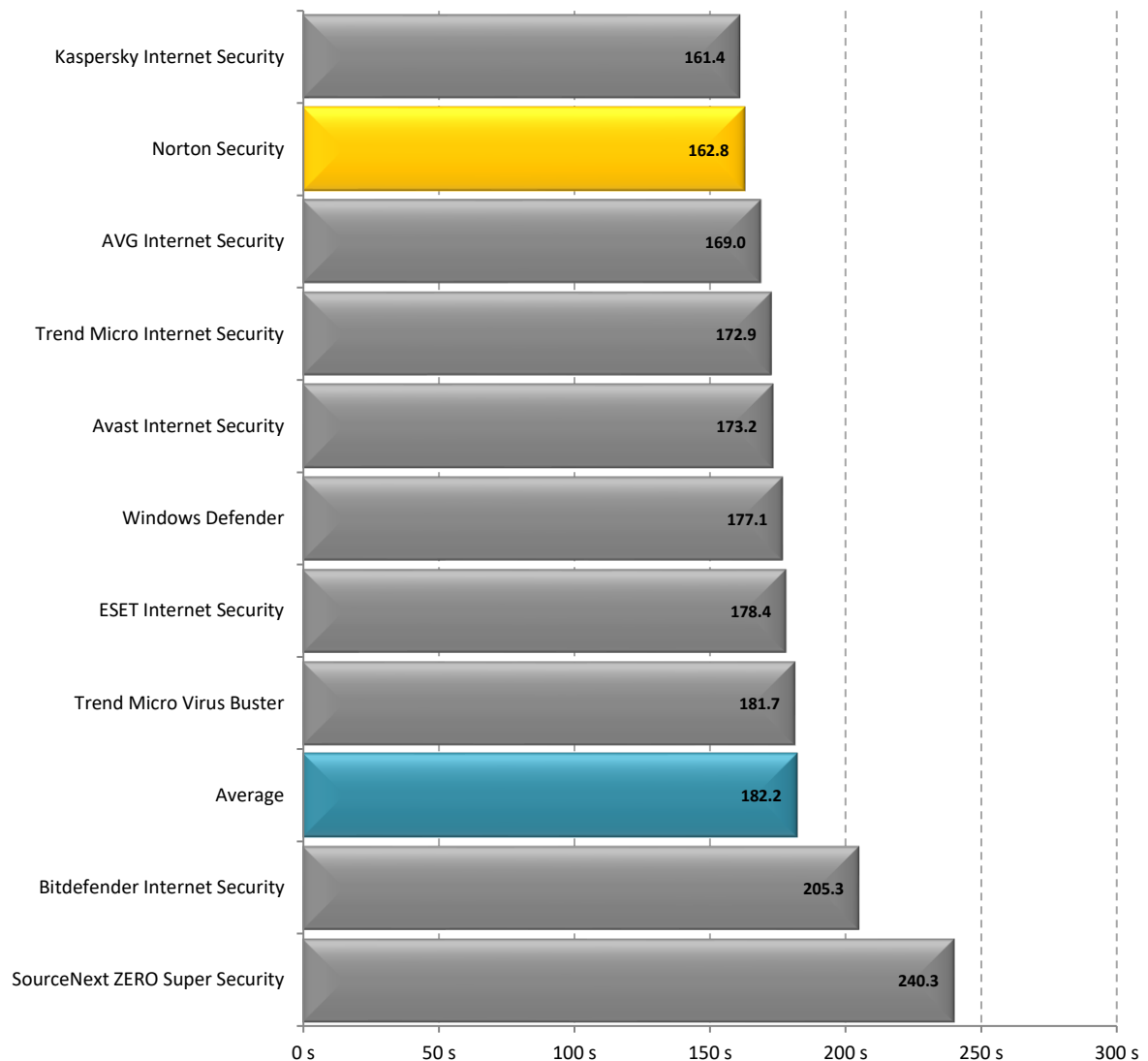
Benchmark 17 – File Copy Disk to Disk (seconds)

The following chart compares the average time taken to copy a total of 8,501 files, with a total file size of 5,995 MB files, from one local drive to another local drive for each Internet Security product tested. The test was performed 5 times, and the average of all 5 runs was taken as the result. Products with lower times are considered better performing products in this category.



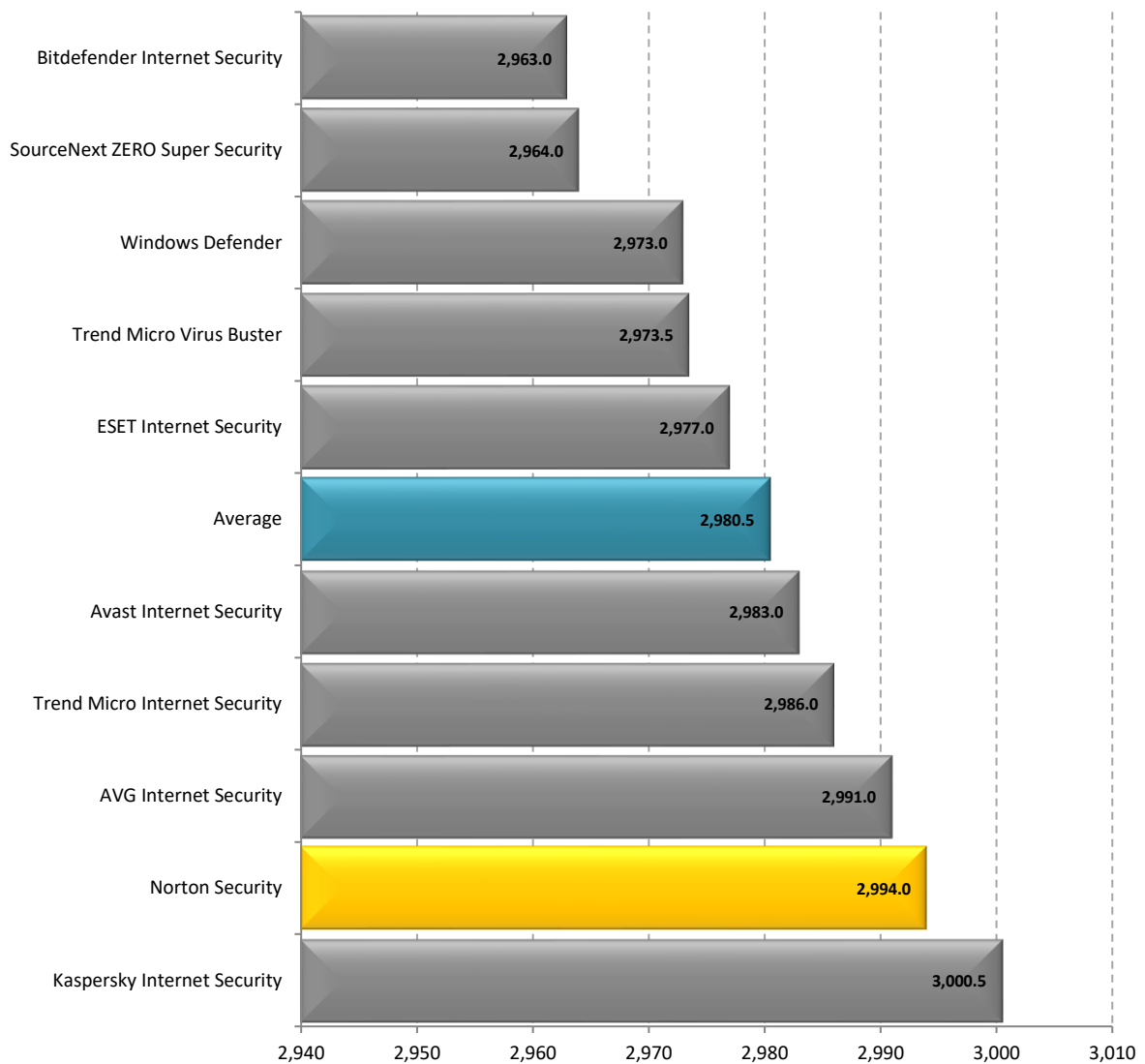
Benchmark 18 – File Copy Over Network (seconds)

The following chart compares the average time taken to transfer a total of 8,501 files over the local network, with a total file size of 5,995 MB files, from a local drive on the test machine to a local server. The test was performed 5 times, and the average of all 5 runs was taken as the result. Products with lower times are considered better performing products in this category.



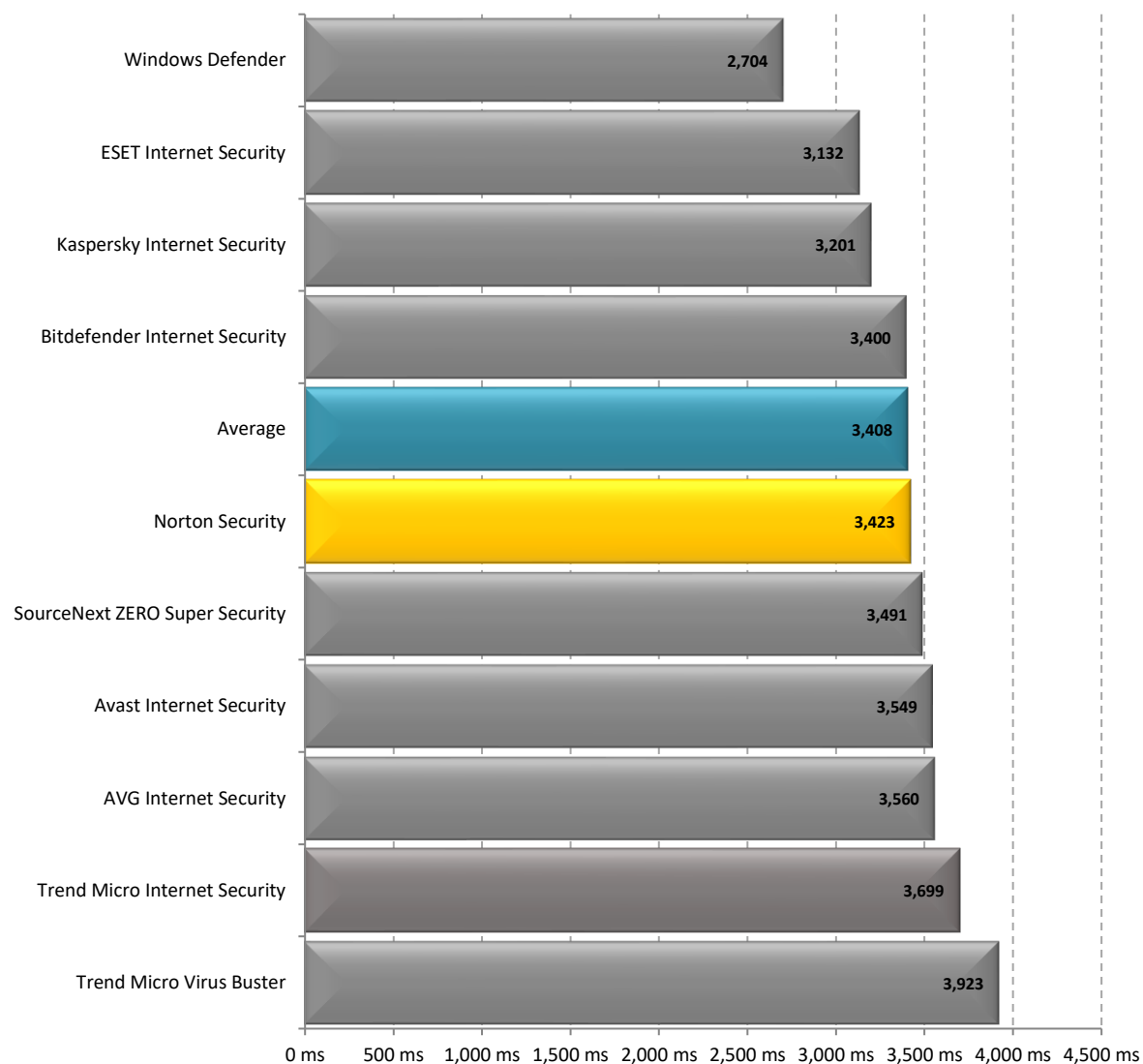
Benchmark 19 – PCMark 8 Home Score

The following chart compares the average PCMark 8 Home score for each Internet Security product tested. The test was performed 2 times, and the average of both runs was taken as the result. Products with higher scores are considered better performing products in this category.



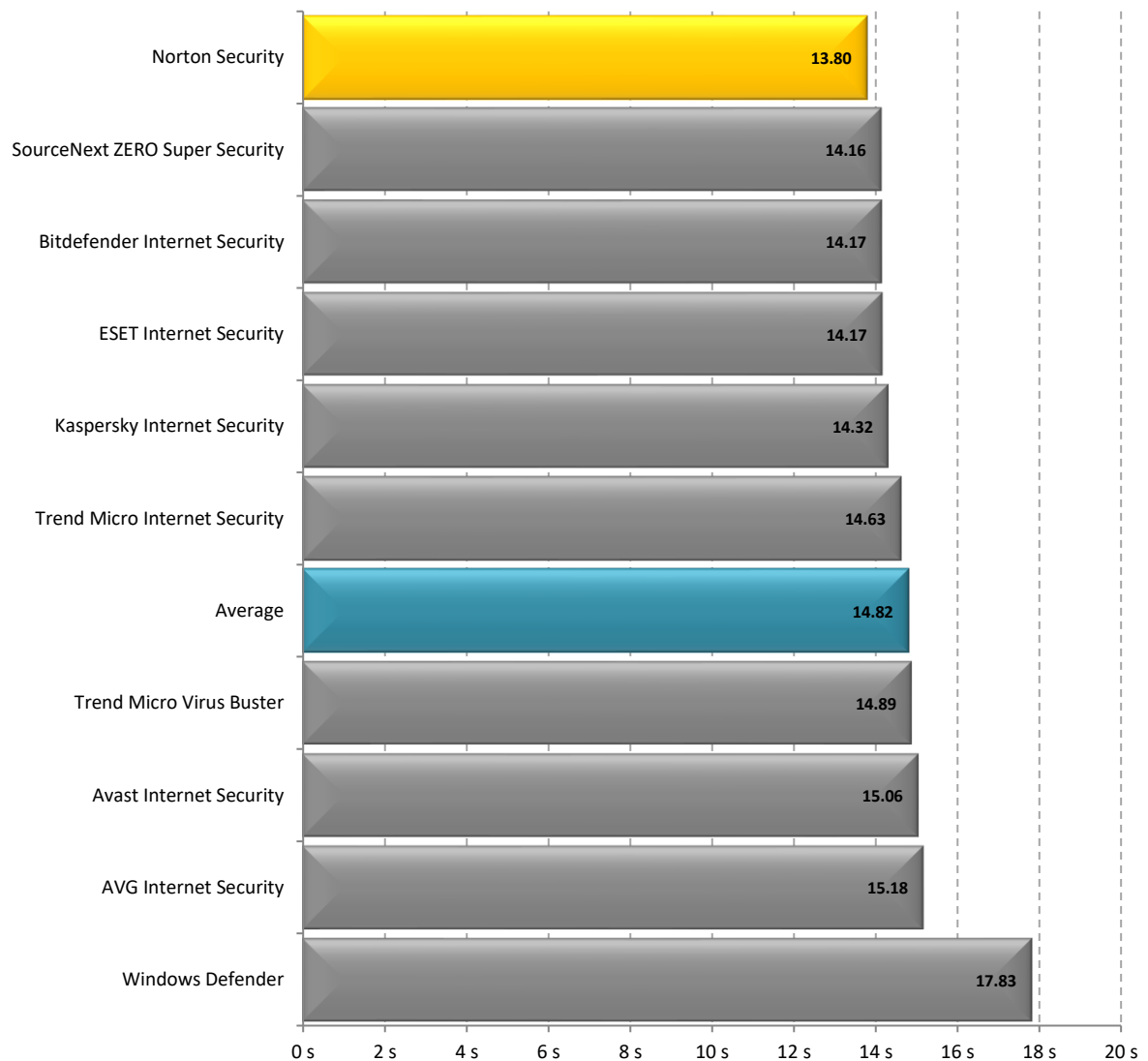
Benchmark 20 – Word Document Launch and Open Time (milliseconds)

The following chart compares the average time taken to launch Microsoft Word and open a 10MB document. Products with lower launch times are considered better performing products in this category.



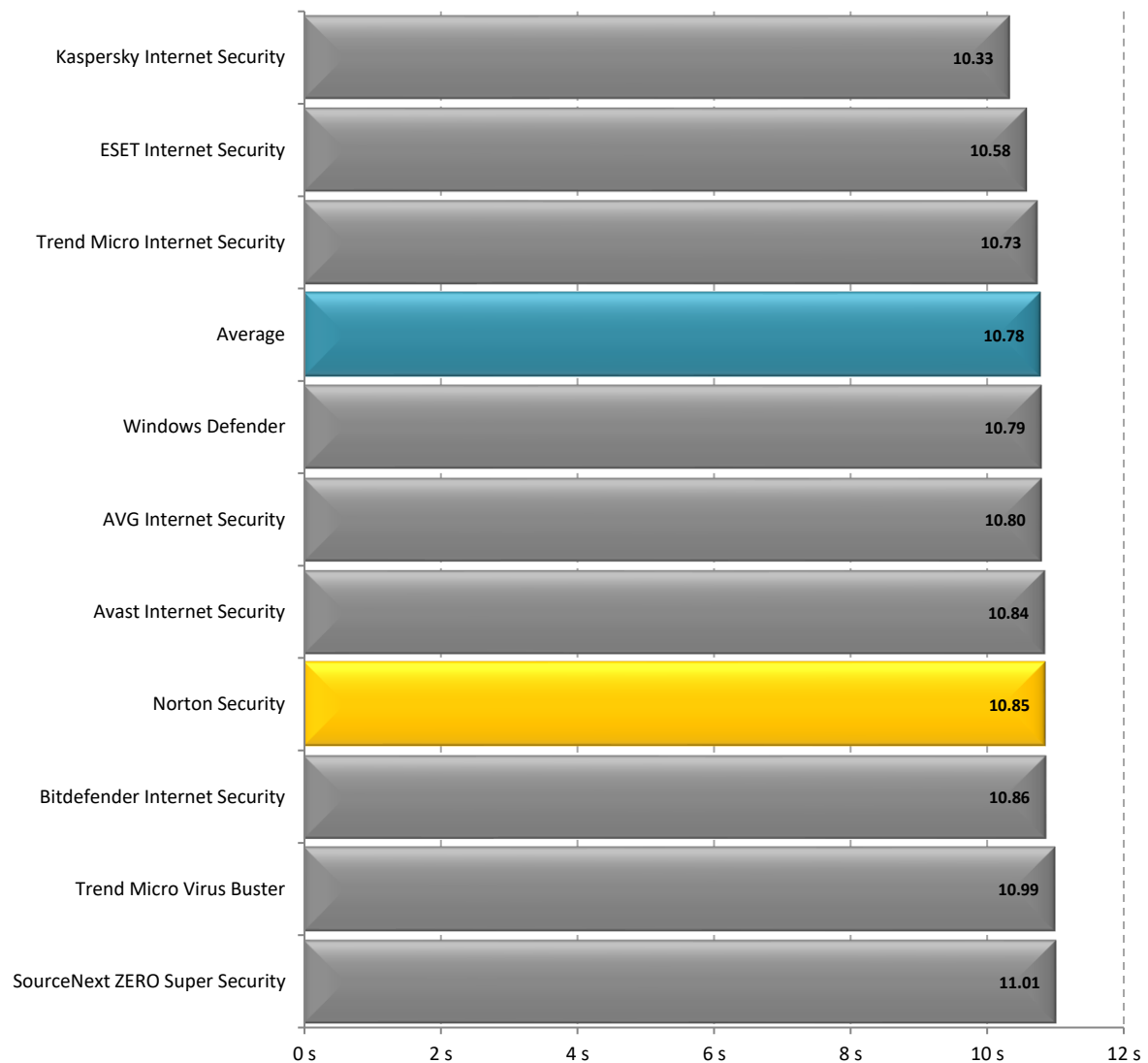
Benchmark 21 – Run Excel Macro (seconds)

The following chart compares the average time taken to launch Microsoft Excel and run a macro. Products with lower times are considered better performing products in this category.



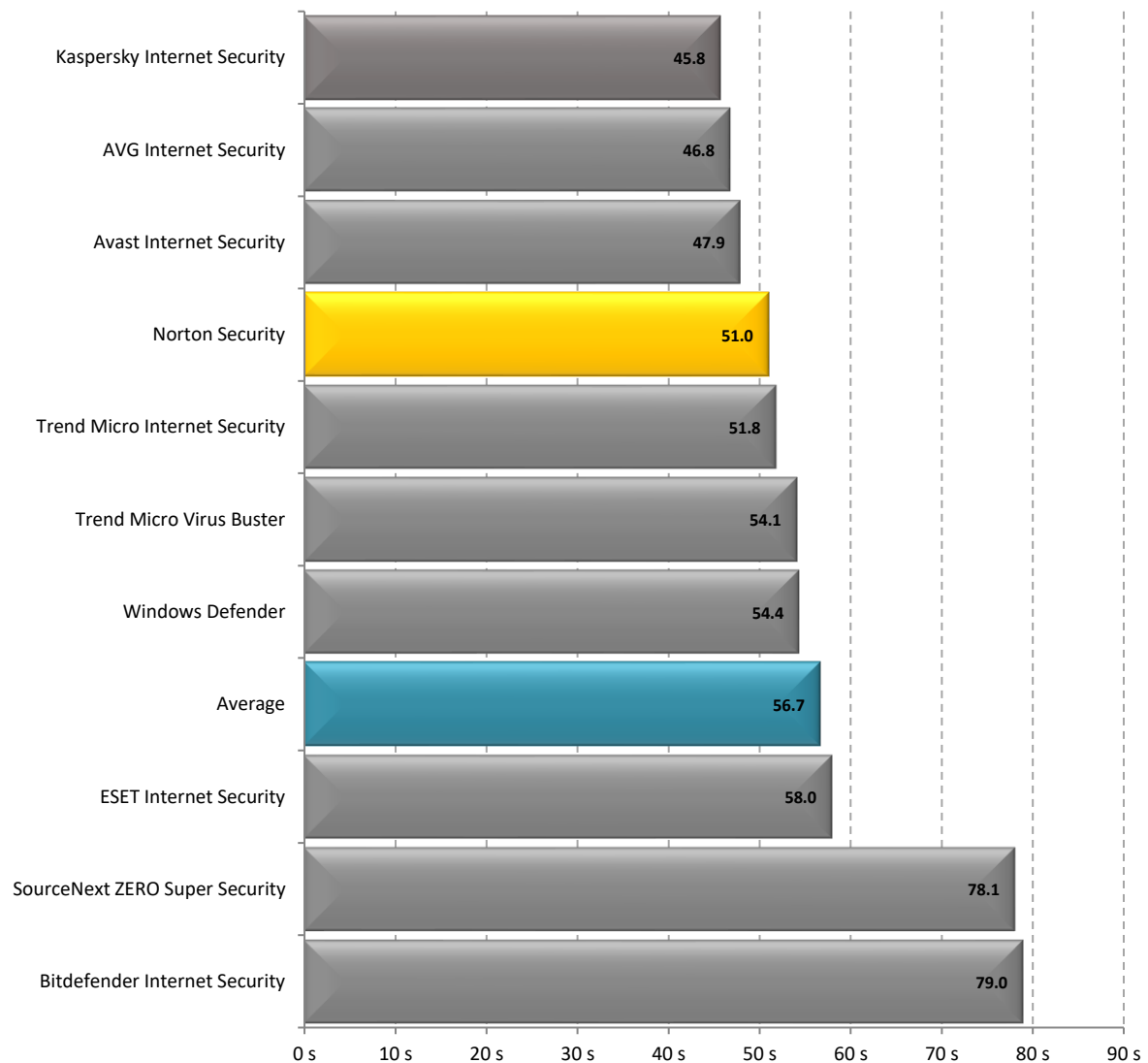
Benchmark 22 – Save Word Document to PDF (seconds)

The following chart compares the average time taken to launch Microsoft Word and open a 10MB document and save it as a PDF. Products with lower times are considered better performing products in this category.



Benchmark 23 – USB 3.0 File Copy (seconds)

The following chart compares the average time taken to copy a set of files from an external USB 3.0 drive to a local disk. Products with lower times are considered better performing products in this category.



Disclaimer and Disclosure

This report only covers the versions that are listed in the “Products and Versions” section of this report.

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

Symantec Corporation funded the production of this report, selected the test metrics and list of products to be included in this report, and supplied some of the test scripts used for the tests.

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, PassMark Software used a test environment running Windows 10 Pro (64-bit) with the following hardware specifications:

Testing Machine

Model:	Lenovo H50W-50 i5
CPU:	Intel Core i5-4460 CPU @ 3.20GHz
Video Card:	NVIDIA GeForce GT 705
RAM:	8GB DDR3 RAM
Main Boot Drive:	Samsung SSD 850 PRO 512GB
2nd Drive:	Samsung SSD 850 PRO 512GB
3rd Drive:	Seagate HDD 1TB (7200 RPM)
Network:	1Gbps (Connected via Gigabit Ethernet cable)
O/S:	Windows 10 Pro v1803 (Build 17134.228)

For network tests (wget, http), PassMark Software used a server with the following specifications:

Testing Server

Motherboard:	Intel S1200BTL Server Motherboard
CPU:	Intel Xeon E3-1220 V2 @ 3.10 GHz
Video Card:	Intel Integrated Graphics Card
RAM:	8GB (2x4GB) ECC RAM, 1333 MHz
Hard Drive:	SSD 128GB
Network:	1Gbps (Connected via Gigabit Ethernet cable)
O/S:	Windows Server 2012 Standard

Appendix 2 – Methodology Description

Windows 10 Image Creation

A bootable version of Macrium Reflect (Backup software) was used to create a “clean” baseline image prior to testing. Our aim was to create a baseline image with the smallest possible footprint and reduce the possibility of variation caused by external operating system factors.

The baseline image was restored prior to testing of each different product. This process ensures that we install and test all products on the same, “clean” machine.

The steps taken to create the base Windows 10 image were as follows:

1. Installation and activation of **Windows 10 Pro** Edition.
2. Disable Automatic Updates.
3. Change User Account Control settings to “Never Notify”.
4. Disable Windows Defender automatic scans to avoid unexpected background activity.
5. Disable the Windows firewall to avoid interference with security software.
6. Disable *Superfetch* to ensure consistent results.
7. Install *HTTP Watch* for Browse Time testing.
8. Install *Windows Performance Toolkit x64* for Boot Time testing.
9. Install Active Perl for interpretation of some test scripts.
10. Install OSForensics for testing (Installation Size) purposes.
11. Install Macrium Reflect for backup and restore of baseline image.
12. Update Windows.
13. Disable Windows Automatic Updates.
14. Create a baseline image of the boot drive using Macrium Reflect.

Benchmark 1 – Boot Time

PassMark Software used tools available from the *Windows Performance Toolkit* (as part of the Microsoft Windows 10 ADK obtainable from the [Microsoft Website](#)).

The Boot Time test was conducted as an individual assessment via the Windows Assessment Console. The network connection was disabled, and the login password was removed to avoid interruption to the test. The final result was taken as the total boot duration excluding BIOS load time.

Benchmark 2 – Scan Time

Scan Time is the time it takes for each product to scan a set of sample files. The sample used was identical in all cases and contained a mixture of system files and Office files. In total there were 6,159 files whose combined size was 982 MB. Most of these files came from the Windows system folders. As the file types can influence scanning speed, the breakdown of the main file types, file numbers and total sizes of the files in the sample set is given here:

File Extension	Number of Files	File Size (MB)
.dll	2,589	490
.exe	695	102
.sys	332	23
.gif	302	1
.doc	281	64
.wmf	185	2
.png	149	2
.html	126	1
.nls	80	6
.jpg	70	1
.ini	59	2
.ico	58	<1
.mof	43	6
.ax	39	4
.xls	38	3
.ime	35	5
.drv	31	1
.txt	31	1
.chm	30	6
.cpl	29	4
.mfl	29	3
.inf	26	2
.hlp	22	3
.imd	20	18
.py	20	<1
.msc	18	1
.vbs	18	1
.xml	18	1
.rtf	16	62
.ocx	16	4
.tsp	14	1
.com	14	<1
.xsl	14	<1
.h	13	<1
.vsd	12	2
.scr	12	2
.aw	12	2
.js	12	1
.zip	11	25

.lex	9	10
.ppt	9	4
.acm	9	1
.wav	7	5
Total	6,159	982

This scan was run without launching the product's user interface, by right-clicking the test folder and choosing the "Scan Now" option. To record the scan time, we have used product's built-in scan timer or reporting system. Where this was not possible, scan times were taken manually with a stopwatch.

In previous years of testing, we noticed many more products showing a substantial difference between the initial scan time (first scan) and subsequent scan times (scans 2 to 5). We believe this behavior is due to products themselves caching recently scanned files. As a result of this mechanism, we have averaged the four subsequent scan times to obtain an average subsequent scan time. Our final result for this test was an average of the subsequent scan average and the initial scan time.

Benchmark 3 – User Interface Launch Time

The launch time of a product's user interface was taken using *AppTimer* (v1.0.1006). For each product tested, we obtained a total of fifteen samples from five sets of three UI launches, with a reboot before each set to clear caching effects by the operating system. When compiling the results, the first of each set was separated out so that there was a set of values for the initial launch after reboot and a set for subsequent launches.

We have averaged the subsequent launch times to obtain an average subsequent launch time. Our final result for this test was an average of the subsequent launch average and the initial launch time.

In some cases, *AppTimer* did not correctly record the time taken for UI launch. For instance, some applications would open their window and look like they were ready, but then continued to be unresponsive. Where the measurement from *AppTimer* appeared inaccurate, we have taken the time manually with a stopwatch.

AppTimer is publicly available from the [PassMark Website](#).

Benchmark 4 – Memory Usage during System Idle

The *MemLog++* utility was used to record process memory usage on the system at boot, and then every minute for another fifteen minutes after. This was done only once per product and resulted in a total of 15 samples. The first sample taken at boot was discarded.

The *MemLog++* utility records memory usage of all processes, not just those of the anti-malware product. As a result of this, an anti-malware product's processes needed to be isolated from all other running system processes. To isolate relevant process, we used a program called *Process Explorer* which was run immediately upon the completion of memory usage logging by *MemLog++*. *Process Explorer* is a Microsoft Windows Sysinternals software tool which shows a list of all DLL processes currently loaded on the system.

Benchmark 5 – Memory Usage during Initial Scan

The **MemLog ++** utility was used to record process memory during an on-demand scan of the boot drive over a 2-minute period. This was done only once per product and resulted in a total of 10 samples each.

The **MemLog ++** utility records memory usage of all processes, not just those of the anti-malware product. As a result of this, an anti-malware product's processes needed to be isolated from all other running system processes. To isolate relevant process, we used a program called **Process Explorer** which was run immediately upon the completion of memory usage logging by **MemLog ++**. **Process Explorer** is a Microsoft Windows Sysinternals software tool which shows a list of all DLL processes currently loaded on the system.

Benchmark 6 – Browse Time

We used JavaScript to load a list of 103 'popular' websites consecutively from a local server.

On each page in the sample data, a few lines of JavaScript were added to the website's html to execute the JavaScript script that loads the next website in the chain. To begin with, once the first website had been loaded completely, the script was executed to load the second website in the chain. Once this had finished loading, the script was executed to then load the third website in the chain. This process was repeated until the final website in the chain had been loaded.

The start time and end time of this process were recorded, and the difference was calculated in seconds to get the final result.

For this test, we have used the Windows default browser **Microsoft Edge** (Version 42.17134.1.0).

Front pages of high traffic websites were used in this test including shopping, social, news, finance and reference websites.

The Browse Time test was executed five times and our final result was an average of these five samples. The local server was restarted between different products and one initial 'test' run was conducted.

Benchmark 7 – Edge Launch Time

The average launch time of Microsoft Edge interface was taken using **AppTimer**. This test was practically identical to the User Interface launch time test. For each product tested, we obtained a total of fifteen samples from five sets of three Edge launches, with a reboot before each set to clear caching effects by the operating system. When compiling the results, the first of each set was separated out so that there was a set of values for the initial launch after reboot and a set for subsequent launches.

For this test, we have used **Microsoft Edge** (Version 42.17134.1.0) as our test browser.

We have averaged the subsequent launch times to obtain an average subsequent launch time. Our final result for this test was an average of the subsequent launch average and the initial launch time.

Benchmark 8 – Installation Time

This test measured the minimum Installation Time a product requires to be fully functional and ready for use by the end user. Installation time can usually be divided in three major phases:

- The **Extraction and Setup phase** consists of file extraction, the EULA prompt, product activation and user configurable options for installation.
- The **File Copy phase** occurs when the product is being installed; usually this phase is indicated by a progress bar.
- The **Post-Installation phase** is any part of the installation that occurs after the File Copy phase. This phase varies widely between products; the time recorded in this phase may include a required reboot to finalize the installation or include the time the program takes to become idle in the system tray.

To reduce the impact of disk drive variables, each product was copied to local drive before initializing installation. Each step of the installation process was manually timed with a stopwatch and recorded in as much detail as possible. Where input was required by the end user, the stopwatch was paused, and the input noted in the raw results in parenthesis after the phase description.

Where possible, all requests by products to pre-scan or post-install scan were declined or skipped. Where it was not possible to skip a scan, the time to scan was included as part of the installation time. Where an optional component of the installation formed a reasonable part of the functionality of the software, it was also installed (e.g. website link checking software as part of an Internet Security Product).

Installation time includes the time taken by the product installer to download components required in the installation. This may include mandatory updates or the delivery of the application itself from a download manager. We have noted in our results where a product has downloaded components for product installation.

We have excluded product activation times due to network variability in connecting vendor servers or time taken in account creation.

Benchmark 9 – Installation Size

A product's Installation Size was previously defined as the difference between the initial snapshot of the Disk Space (C:\ drive) before installation and the subsequent snapshot taken after the product was installed on the system. Although this was a widely used methodology, we noticed that the results it yielded were not always reproducible in Vista due to random OS operations that may take place between the two snapshots. We improved the Installation Size methodology by removing as many Operating System and disk space variables as possible.

Using PassMark **OSForensics** we created initial and post-installation disk signatures for each product. These disk signatures recorded the number of files and directories, and complete details of all files on that drive (including file name, file size, checksum, etc.) at the time the signature was taken.

The initial disk signature was taken immediately prior to installation of the product. A subsequent disk signature was taken immediately following a manual update and system reboot after product installation. Using **OSForensics**, we compared the two signatures and calculated the total disk space consumed by files that were

new, modified, and deleted during product installation. Our result for this metric reflects the total size of all newly added files during installation.

OSForensics detects the hard links created during the installation and they were excluded to the count.

Benchmarks 10 – File Copy, Move and Delete

This test measured the amount of time required for the system to copy, move and delete samples of files in various file formats. This sample was made up of 812 files over 743,034.8 KB and can be categorized as documents [26% of total], media files [54% of total] and PE files (i.e. System Files) [20% of total].

The breakdown of the main file types, file numbers and total sizes of the files in the sample set is shown in the following table:

File Format	Number of Files	File Size (KB)
DOC	8	29,736.5
DOCX	4	13,205.48
PPT	3	5,634
PPTX	3	4,049.24
XLS	4	2,598
XLSX	4	1,392.63
PDF	73	133,103.56
ZIP	4	6,148.42
7Z	1	90.08
JPG	351	30,639.9
GIF	6	144.71
MOV	7	56,015.99
RM	1	5,526.02
AVI	8	76,858.8
WMV	5	45,045.08
MP3	28	187,090.22
EXE	19	2,883.71
DLL	104	28,575.75
AX	1	18
CPL	2	2,060
CPX	2	4.28
DRV	10	151.23
ICO	1	105.1
MSC	1	40.61
NT	1	1.65
ROM	2	35.75
SCR	2	2,197.5

SYS	1	36,648.53
TLB	3	132.4
TSK	1	1.13
UCE	1	22.45
EXE	19	2,883.71
DLL	104	28,575.75
AX	1	18
CPL	2	2,060
CPX	2	4.28
DRV	10	151.23
ICO	1	105.1
MSC	1	40.61
NT	1	1.65
ROM	2	35.75
SCR	2	2,197.5
SYS	1	36,648.53
TLB	3	132.4
TSK	1	1.13
UCE	1	22.45
Total	812	743,034.8

This test was conducted five times to obtain the average time to copy, move and delete the sample files, with the test machine rebooted between each sample to remove potential caching effects.

Benchmark 11 – Third-Party Applications Installation Time

This test measured how much time was required to install and uninstall a third-party application. For this test, *CommandTimer.exe* timed how long it took to install and uninstall the following applications on the test machine:

- Firefox 3.6.3 (11,909 KB) (MSI File)
- Microsoft .NET 3.5 (34,121 KB) (MSI File)
- Steam (1,551 KB) (MSI File)

This test was conducted five times to obtain the average time to install/uninstall the above third-party programs, with the test machine rebooted between each sample to remove potential caching effects.

Benchmark 12 – Network Throughput

This benchmark measured how much time was required to download a sample set of binary files of various sizes and types over a 100MB/s network connection. The files were hosted on a server machine running Windows Server 2012. *CommandTimer.exe* was used in conjunction with *GNU Wget* (version 1.10.1) to time and conduct the download test.

The complete sample set of files was made up of 527.99 MB over 484 files and two file type categories: media files [74% of total] and documents [26% of total]. The breakdown of the file types, file numbers and total sizes of the files in the sample set is shown in the following table:

File Format	Number of Files	File Size (MB)
JPEG	343	29.25
GIF	9	0.34
PNG	5	0.47
MOV	7	54.7
RM	1	5.4
AVI	8	75.06
WMV	5	43.99
MP3	28	182.71
PDF	73	129.98
ZIP	4	6
7Z	1	0.09
Total	484	527.99

This test was conducted five times to obtain the average time to download this sample of files, with the test machine rebooted between each sample to remove potential caching effects.

Benchmark 13 – File Format Conversion (MP3 → WAV, MP3 → WMA)

This test measured how much time was required to convert five (5) different MP3 files into WAV files and subsequently, convert the same MP3 samples into a WMA files. The total size of the files was 24.68 MB.

To encode the MP3 into another format, we used an application called *ffmpeg.exe*. The format conversion process was timed using *CommandTimer.exe*.

This test was conducted five times to obtain the average conversion speed between these formats, with the test machine rebooted between each sample to remove potential caching effects.

Benchmark 14 – File Compression and Decompression

This test measured the amount of time required to compress and decompress a sample set of files. For this test, we used a subset of the media and documents files used in the *File Copy, Move and Delete* benchmark. *CommandTimer.exe* recorded the amount of time required for *7zip.exe* to compress the files into a *.zip and subsequently decompress the created *.zip file.

This subset comprised 1,218 files over 783 MB. The breakdown of the file types, file numbers and total sizes of the files in the sample set is shown in the following table:

File Extension	Number of Files	File Size (MB)
.xls	13	9.23
.xlsx	9	3.51
.ppt	9	7.37
.pptx	11	17.4
.doc	17	35.9
.docx	19	24.5
.gif	177	1.10
.jpg	737	66.2
.png	159	48.9
.mov	7	54.7
.rm	1	5.39
.avi	46	459
.wma	11	48.6
.avi	46	459
.wma	11	48.6
Total	1,218	783

This test was conducted five times to obtain the average file compression and decompression speed, with the test machine rebooted between each sample to remove potential caching effects.

Benchmark 15 – File Download

This test measured the time it took to complete an http download of a sample set of installation files over a 100MB/s network connection. The files were hosted on a local server machine running Windows Server 2012. *CommandTimer.exe* was used in conjunction with the native .NET *DownloadFile()* method to time and conduct the download test. The file set includes the following:

- Firefox Setup 31.0 (30.7 MB) (EXE file)
- GIMP Setup 2.8.10 (86.2 MB) (EXE file)
- LibreOffice Installer Package 4.2.5 (209 MB) (MSI file)
- Adobe Reader Installer 3.5.4.25 (1 MB) (EXE file)

This test was conducted five times to obtain the average time to download this sample of files, with the test machine rebooted between each sample to remove potential caching effects.

Benchmark 16 – PE Scan Time

This test measured the on-demand scan times of a file set comprised only of executable files (.exe, .dll and .sys files). We performed five scans of the sample file set, with a machine restart between each scan to remove possible caching effects. The time taken to scan the files was taken from a security product's scan logs, or where logs were not available, manually with a stopwatch. Scans were launched by right clicking on the folder to be scanned.

A breakdown of the sample file set is as follows:

File Type	Number of Files	File Size (MB)
SYS Files	2,174	329
DLL Files	2,037	920
EXE Files	2,140	827
Total	6,351	2,076

We have averaged the four subsequent scan times to obtain an average subsequent scan time. Our final result for this test was an average of the subsequent scan average and the initial scan time.

Benchmark 17 – File Copy Disk to Disk

This test measured the amount of time taken to copy files between two local drives. The data set comprised of 8,501 files with a total file size of 5,995 MB, and the formats used included documents, movies, images and executables. A breakdown of the sample file set is given below:

File Extension	Number of Files	File Size (MB)
.jpg	2,903	588
.dll	773	25
.exe	730	197
.gif	681	63
.wav	430	260
.sys	501	79
.png	451	27
.mp3	333	2,157
.wma	585	925
.docx	267	81
.avi	247	1,079
.doc	160	57
.xls	329	132
.ppt	97	148
.zip	14	177
Total	8,501	5,995

A total of five runs of this test were performed, with a machine restart between each run. The time taken to copy files was measured and recorded by *CommandTimer.exe*. All the files were copied between a folder on the local drive and a 2nd folder on a different drive. Files were deleted from the 2nd drive once the copy was complete. The final result was calculated as an average of the five samples.

Benchmark 18 – File Copy Over Network

This test measured the amount of time taken to transfer files from a local drive on the test machine to a shared folder on the local server. The data set comprised of 8,501 files with a total file size of 5,995 MB, and the formats used included documents, movies, images and executables. A breakdown of the sample file set is given below:

File Extension	Number of Files	File Size (MB)
.jpg	2,903	588
.dll	773	25
.exe	730	197
.gif	681	63
.wav	430	260
.sys	501	79
.png	451	27
.mp3	333	2,157
.wma	585	925
.docx	267	81
.avi	247	1,079
.doc	160	57
.xls	329	132
.ppt	97	148
.zip	14	177
Total	8,501	5,995

A total of five runs of this test were performed, with a machine restart between each run. The time taken to transfer files was measured and recorded by *CommandTimer.exe*. Files were deleted from the server once the copy was complete. The final result was calculated as an average of the five samples.

Benchmark 19 – PCMark 8 Home Score

This test aims to benchmark the test machine's performance over a series of common home computing tasks. This includes web browsing, writing, photo editing, video chat, and casual gaming. For this benchmark we have installed *PCMark 8 Professional Edition* (Version 2.7.613). From the interface, the PCMark 8 Home test was launched in conventional mode. This Home test was run twice, with a reboot in between. The final result was an average of the overall scores of the two iterations.

Benchmark 20 – Word Document Launch and Open Time

The average launch time of Word interface was taken using *AppTimer*. This included the time to launch the Word application and open a 10 MB document. This test was practically identical to the User Interface launch time test. For each product tested, we obtained a total of fifteen samples from five sets of three Word launches, with a reboot before each set to clear caching effects by the operating system. When compiling the results, the first of each set was separated out so that there was a set of values for the initial launch after reboot and a set for subsequent launches.

We have averaged the subsequent launch times to obtain an average subsequent launch time. Our final result for this test was an average of the subsequent launch average and the initial launch time.

Benchmark 21 – Run Excel Macro

This test measured the time taken to open an excel document and run an excel macro. The macro performed a range of operations, including reading data from the file system, mathematical calculations, and writing data to the spreadsheet. This test was run using a Windows PowerShell script. We obtained a total of five samples with a reboot in between each to clear caching effects by the operating system. The final result was calculated as an average of the five samples.

Benchmark 22 – Save Word Document to PDF

This test measured the time taken to open a large Word Document (~10 MB) and save it as a PDF document. This test was run using a Windows PowerShell script. We obtained a total of five samples with a reboot in between each to clear caching effects by the operating system. The final result was calculated as an average of the five samples.

Benchmark 23 – USB 3.0 File Copy

This test measured the amount of time taken to transfer files from an external drive to a local disk via a USB 3.0 connection. The data set comprised of 6,834 files with a total file size of 6,177.78 MB, and the formats used included documents, movies, images, executables and system files. A breakdown of the sample file set is given below:

File Extension	Number of Files	File Size (MB)
.dll	1,429	627
.doc	496	164
.docx	324	76
.exe	280	1,228.8
.jpg	1,564	469
.mp3	394	1,085.44
.pdf	567	246
.pptx	556	1,085.44
.pst	2	59.1
.sys	568	145
.xlsx	584	81
.zip	70	911
Total	6,834	6,177.78

A total of five runs of this test were performed, with a machine restart between each run. The time taken to transfer files was measured and recorded by *CommandTimer.exe*. Files were deleted from the local disk once the copy was completed. The final result was calculated as an average of the five samples.