COMPARATIVE REVIEW

RED HAT LINUX 9

Matt Ham

Since *Virus Bulletin*'s last *Linux* comparative review (see *VB*, May 2003, p.18), the *Linux* bandwagon has rolled along steadily, gaining momentum and providing ever more financial reason for vendors to provide a product for the platform. Only 11 products were submitted for last year's *Linux* review, two of which do not reappear this time (*GeCAD* having been absorbed by *Microsoft* and *Norman* not having submitted on this occasion) – however there are five newcomers: *SOFTWIN*, *NAI*, *CAT*, *Grisoft* and *Eset*.

Only two VB 100% awards were achieved in the 2003 *Linux* comparative, with the on-access components of the products proving the largest source of trouble. A year later, it is clear that the feasibility of various scanning methods has been tested in the marketplace, and there is an appearance of greater homogeneity in the methods used. 'Appearance' is the key word, however, because the scanning methods used in several products are not mentioned in any detail, leaving only guesswork to determine how scanning is performed. Seasoned campaigners would, at this point, berate me with cries of 'RTFM!' – which would be easier if manuals were always provided, but more of that later.

The primary methods for on-access scanning on *Samba* shares (this being the chosen area for the tests) can be divided into those in which the scanning is performed only on the *Samba* share, and those in which all files are scanned. Where only the *Samba* share is scanned, the predominant method is the insertion of 'vfs object = <filename.so>' into the smb.conf file. This can be applied globally or on a per-share basis. Where scanning of all file accesses is desired a kernel object may be inserted and scanning performed by means of a daemon. The most popular way of doing this is via the *Dazuko* module.

Some problems occurred in on-access scanning. The problems with the vfs object method of scanning seemed primarily due to overloading of the *Samba* processes. This resulted in slowed scanning, the creation of large numbers of *Samba* processes and permanent or temporary termination of the *Samba* connection. *VirusBuster*'s developers warned that there were known, temporary problems with their product where 10,000 infected file accesses were exceeded. Other products demonstrated similar problems without prior warning.

More problems resulted from the old chestnut of insufficient information. In some cases documentation was lacking and in other cases it was hidden. For many products the final destination of the installed files was a mystery, which made finding and activating on-access scanning unneccesarily difficult. Simple tasks, such as loading daemons, may seem obvious to a developer, but for a user who is not even sure how on-access scanning is intended to operate, the absence of instructions for such tasks is infuriating.

DAZUKO

Available from http://www.dazuko.org/, *Dazuko* is an open source file access control interface, designed to be used over a full range of *Linux* and *BSD* platforms. While it is linked with *H+BEDV*, which has provided much of the funding for the project, *Dazuko* demonstrates sufficient independence for other companies to have felt no qualms about using its functionality. The total package is less than 60 KB in size, so distribution is easy from a logistical point of view.

Since such low-level interaction with the file system is not possible without direct reference to the kernel on the current machine, *Dazuko* must be compiled locally before it can be used. It was with a degree of trepidation that I noted in the readme for the module that the instructions were for a 'quick and dirty' installation. From past experience these words can be translated as 'this won't work, refer to the 1000-page manual for a better way'. On this occasion, however, the quick and dirty method proved simply to be quick. All that was required was to allow the module to configure before making it, inserting it and activating it – each process being a matter of one command line which, for the truly lazy, can be cut and pasted from the readme. In all, *Dazuko* was a pleasure to work with.

Where *Dazuko* was concerned, the low-level nature of the scanning initiated by the module was something of a problem when interacting with on-demand scanners. By choice, the on-access components of products are disabled whenever on-demand scans are carried out during comparative tests. However, sanity-checking exercises on single files demonstrated that there were cases where on-demand scanning would show no detections, since the on-access portion of the scanner was denying access to infected objects. This was not a major problem during testing, though in real-world situations this could be more of an issue.

THE TESTS

In general the testing methodology varied little from the standard methods used for the *Windows* tests. *Samba* testing was an exception, since it is unique to these *Linux* tests. The test client runs *Windows NT4 SP 6* and is configured to access the collection of infected files in addition to various directories used in file transfers. These transfers are for the installation of the applications on the *Linux* machines and extraction of results – during testing of the on-access components there is no other file activity between the two machines. Detection is considered to be confirmed when file access through fopen() is denied to an infected file. In cases where this does not trigger detection, denial of file copying is considered to be equivalent. For products which are unstable over large test sets the *Samba* process was restarted between tests.

Alwil Avast! 0.2.0

ItW File	100.00%	**Macro**	99.56%
ItW File (o/a)	100.00%	**Standard**	99.36%
Linux	70.00%	**Polymorphic**	93.58%

One of the two products which gained a VB 100% award in the last *Linux* comparative, *Avast!* is the first of those in this review that use the *Dazuko* module. The installation method is via shell scripts and is slightly long-winded as a result. This consists of installing *Dazuko*, installing the core *Avast!* engine modules, installing the scanner modules and finally activating the scanning daemon manually. Thankfully, the documentation was thorough.

On-demand scanning ran without any problems at all. However, there were some issues with on-access scanning when the whole test set was passed through in one batch. This caused a scattering of unblocked files distributed randomly across the set. With a slightly slower throughput, however, the detection became consistent and approached that of the on-demand scans. Historically, such cases have resulted in a VB 100% award, along with the caveat that, in this version at least, some files may be missed. It should be noted, however, that this version of the product is not fully released as yet, so some problems would be expected.

CAT QuickHeal X Gen Ver 7.01

ItW File	100.00%	**Macro**	97.51%
ItW File (o/a)	100.00%	**Standard**	83.42%
Linux	40.00%	**Polymorphic**	91.84%

Another *Dazuko*-based scanner, *QuickHeal* also uses shell scripts to perform installation. However, there was some initial confusion in the installation procedure, since the installation shell script was not tagged as an executable file. This was easy to correct, if mystifying, and once this obstacle had been overcome the process was completed quickly. With full detection of viruses in the In the Wild (ItW) test set and no false positives, *QuickHeal* gains a VB 100% award.

DialogueScience Dr.Web 4.31.1

ItW File	100.00%	**Macro**	100.00%
ItW File (o/a)	100.00%	**Standard**	100.00%
Linux	100.00%	**Polymorphic**	100.00%

Dr.Web consists of two RPM packages: one for the command line version and another for the *Samba* functionality. The scanning of *Samba* accesses is performed via the insertion of a vfs object reference in the smb.conf file, thus offering on-access detection only for *Samba* accesses. *Dr.Web* flagged 12 files as suspicious, but no full-blown false positives. The product's detection rate was much more impressive, with all infected files detected. *Dr.Web* thus overcomes its recent blip in performance and adds another VB 100% to its current collection.

Eset NOD32 2.01 1.650

ItW File	100.00%	**Macro**	100.00%
ItW File (o/a)	100.00%	**Standard**	99.91%
Linux	100.00%	**Polymorphic**	100.00%

The installation of *NOD32* involves two RPM files and an additional shell script, on top of the required *Dazuko* compilation. As far as on-demand scanning was concerned matters were simple enough, with only one miss in the whole test set. This was of W32/Lovelorn.A in its DLL form – a new entry in the standard test set and not a worrying miss. Matters were not so trouble-free, however, where on-access scanning was concerned. The documentation provided was copious in quantity, but lacking any form of troubleshooting information where scanning failed to initialise. The developers were consulted, and the problem investigated further – happily for *Eset* the result was a VB 100% award.

FRISK F-Prot Antivirus 4.3.5 3.14.8

ItW File	100.00%	**Macro**	100.00%
ItW File (o/a)	100.00%	**Standard**	99.82%
Linux	66.67%	**Polymorphic**	99.91%

Having been reviewed as a standalone product recently (see *VB*, December 2003, p.14), the installation of *F-Prot Antivirus* has become something of a routine. The insertion of the preload instructions into the *Samba* configuration file must be performed manually, but it is documented well enough for this to be only a minor chore.

On-access tests	ItW file		Macro		Polymorphic		Standard		Linux	
	Number missed	%	Number missed	%	Number missed	%	Number missed	%	Number missed	%
Alwil Avast!	0	100.00%	18	99.56%	112	93.58%	18	99.42%	9	80.00%
CAT QuickHeal	0	100.00%	102	97.51%	1277	91.84%	315	83.30%	26	40.00%
DialogueScience Dr.Web	0	100.00%	0	100.00%	0	100.00%	0	100.00%	0	100.00%
Eset NOD32	0	100.00%	0	100.00%	0	100.00%	1	99.91%	0	100.00%
FRISK F-Prot Antivirus	0	100.00%	0	100.00%	2	99.91%	4	99.69%	1	93.33%
F-Secure Anti-Virus	0	100.00%	0	100.00%	0	100.00%	0	100.00%	0	100.00%
Grisoft AVG	0	100.00%	12	99.71%	425	83.72%	46	97.11%	16	48.33%
H+BEDV AntiVir	0	100.00%	56	99.26%	522	87.18%	34	98.42%	4	85.33%
Kaspersky Virus Scanner	0	100.00%	0	100.00%	0	100.00%	0	100.00%	0	100.00%
NAI VirusScan	N/A	-	N/A	-	N/A	-	N/A	-	N/A	-
SOFTWIN BitDefender	4	99.12%	21	99.49%	11	97.46%	69	97.49%	6	83.33%
Sophos SWEEP	N/A	-	N/A	-	N/A	-	N/A	-	N/A	-
Trend ServerProtect	0	100.00%	0	100.00%	215	95.77%	11	99.56%	4	93.33%
VirusBuster VirusBuster	0	100.00%	0	100.00%	102	91.45%	11	99.60%	39	13.33%

On-access the scanning performed well, though there was some noticeable slowing especially on some of the polymorphic test sets. On demand there was no such slow down, leaving one to conclude that the issue lies with the on-access component. Despite a slow performance in places, the product's detection rates were certainly sufficient to qualify for a VB 100% award.

F-Secure Anti-Virus 4.60 3100

ItW File	100.00%	**Macro**	100.00%
ItW File (o/a)	100.00%	**Standard**	100.00%
Linux	100.00%	**Polymorphic**	100.00%

Being related to *F-Prot Antivirus* by way of the *FRISK* engine within, the performance of *F-Secure Anti-Virus* was expected to be similar. The installation method marked the first difference between the products, in *F-Secure*'s case consisting of a shell script which leaves little configuration to the administrator. Similarities in scanning performance existed to a certain degree, in that the on-access engine showed distinct slowness on certain polymorphic files. However, the slow speed of scanning did not affect the product's thoroughness and *F-Secure Anti-Virus* earned a VB 100% easily.

Grisoft AVG 7.03 262

ItW File	100.00%	**Macro**	99.63%
ItW File (o/a)	100.00%	**Standard**	97.36%
Linux	81.67%	**Polymorphic**	83.72%

Returning to *Dazuko*-powered scanners, *AVG* was of mixed pleasure to install. Installation is via an RPM file which distributes files to various directories scattered across the file system. Upon detecting these it was necessary to activate an update application and to install a licence key, again through an application. This over-complicated matters to an irritating degree. After installation, however, the *AVG* scanners performed well, and there were no further problems of any sort. No false positives were registered, and In the Wild detection was exemplary. *Grisoft* thus receives a VB 100% award.

H+BEDV AntiVir 2.1.0-9 6-24-0-39

ItW File	100.00%	**Macro**	99.55%
ItW File (o/a)	100.00%	**Standard**	98.45%
Linux	57.00%	**Polymorphic**	87.18%

As might be expected from the company's support of the *Dazuko* development process, *AntiVir* uses the *Dazuko* engine as its method of scanning. The installation procedure consists of a fairly lengthy shell script, similar to that found in other products. The installation and operation of *AntiVir* was without any noticeable problems as far as functionality or stability were concerned. It is thus of little surprise that *H+BEDV* is in receipt of a VB 100% award.

Kaspersky Virus Scanner 5.0.1.0/#1

ItW File	100.00%	**Macro**	100.00%
ItW File (o/a)	100.00%	**Standard**	100.00%
Linux	100.00%	**Polymorphic**	100.00%

Kaspersky's product was another of those whose documentation caused problems. Initial installation is via RPM file, after which two perl scripts must be run – these were discovered more by luck than judgement. The vfs object was duly added to the smb.conf file, though at this point the *Samba* share simply ceased functioning. Activation of the daemon scanner solved this, though it was difficult to find mention of this workaround in the documentation. The product's detection rates were faultless, however, and *Kaspersky* earns a VB 100% award.

NAI VirusScan 4.32.0 4333

ItW File	100.00%	**Macro**	100.00%
ItW File (o/a)	N/A	**Standard**	99.79%
Linux	80.00%	**Polymorphic**	100.00%

The first product in this review not to offer an on-access component, *VirusScan* arrived as Tar.Z files which were inaccessible to a standard *Red Hat* installation. This did not bode well, though the installation, through a shell script, continued smoothly after this point. With no on-access functionality, a VB 100% award for *VirusScan* was always an impossibility, though on all other fronts the performance was close to admirable. Misses which did occur were limited to archived or packaged objects, since *VirusScan* does not handle archives in its default installation state.

SOFTWIN BitDefender Console 7.0(2489)

ItW File	99.12%	**Macro**	99.49%
ItW File (o/a)	99.12%	**Standard**	97.55%
Linux	60.00%	**Polymorphic**	97.46%

BitDefender opts for an RPM format for installation, though this is packaged within a RUN file so as to offer both licence and configuration functionality. This seemed to be a good compromise between convenience and information. Despite this configuration functionality, however, paths must be inserted manually. The scanning functionality is provided by a vfs object reference in the SMB.conf file. One peculiarity was noted, in that the extension listing for scanned files appeared to be set so that only files with lower-case extensions were scanned. By default, the entire *VB* test set is fully upper case. This obstacle was overcome quickly, but certainly warrants a mention.

Scanning on access proved more of a problem. Files were missed both on access and on demand, and the connection to the *Samba* share had a tendency to break after 10,000 files passed through the scanner. These problems have been acknowledged by the developers and should be rectified in the future.

Sophos SWEEP 3.79

ItW File	100.00%	**Macro**	99.80%
ItW File (o/a)	N/A	**Standard**	99.60%
Linux	60.00%	**Polymorphic**	100.00%

The second product to be tested with no on-access component, *SWEEP* is designed without any such functionality. *SWEEP* is installed though a shell script and is accompanied by documentation in the form of a helpful readme. With no on-access component, *SWEEP* is not eligible for a VB 100% award, though detection rates for on-demand scanning were of the same high standard as seen from *Sophos* in recent *Windows* comparative tests.

Trend ServerProtect 0403Nov03D021004

ItW File	100.00%	**Macro**	100.00%
ItW File (o/a)	100.00%	**Standard**	99.72%
Linux	93.33%	**Polymorphic**	95.77%

On-demand tests	ItW file		Macro		Polymorphic		Standard		Linux	
	Number missed	%	Number missed	%	Number missed	%	Number missed	%	Number missed	%
Alwil Avast!	0	100.00%	18	99.56%	112	93.58%	18	99.36%	11	70.00%
CAT QuickHeal	0	100.00%	102	97.51%	1277	91.84%	313	83.42%	26	40.00%
DialogueScience Dr.Web	0	100.00%	0	100.00%	0	100.00%	0	100.00%	0	100.00%
Eset NOD32	0	100.00%	0	100.00%	0	100.00%	1	99.91%	0	100.00%
FRISK F-Prot Antivirus	0	100.00%	0	100.00%	2	99.91%	2	99.82%	6	66.67%
F-Secure Anti-Virus	0	100.00%	0	100.00%	0	100.00%	0	100.00%	0	100.00%
Grisoft AVG	0	100.00%	15	99.63%	425	83.72%	42	97.36%	10	81.67%
H+BEDV AntiVir	0	100.00%	28	99.55%	522	87.18%	34	98.45%	9	57.00%
Kaspersky Virus Scanner	0	100.00%	0	100.00%	0	100.00%	0	100.00%	0	100.00%
NAI VirusScan	0	100.00%	0	100.00%	0	100.00%	3	99.79%	3	80.00%
SOFTWIN BitDefender	4	99.12%	21	99.49%	11	97.46%	70	97.55%	10	60.00%
Sophos SWEEP	0	100.00%	8	99.80%	0	100.00%	5	99.60%	7	60.00%
Trend ServerProtect	0	100.00%	0	100.00%	215	95.77%	9	99.72%	4	93.33%
VirusBuster VirusBuster	0	100.00%	0	100.00%	102	91.45%	11	99.66%	39	13.33%

ServerProtect is the only product to offer a GUI in the *Linux* comparatives. It is supplied as a BIN file, which acts as a wrapper for an RPM, giving licensing details. The GUI aspect of the software is reached by use of an http connection, performed from a local or remote browser. There was a slight problem in that a stray " ' " was added to one URL when triggering the GUI, though once this had been removed (manually) there were no further problems in program operation.

In terms of detection, *ServerProtect* performed very well, gaining a VB 100% award. Considering the addition of a GUI to the program the throughput rates on the clean sets were not noticeably slower than the bulk of other products.

VirusBuster VirusBuster 1.12.019

ItW File	100.00%	**Macro**	100.00%
ItW File (o/a)	100.00%	**Standard**	99.66%
Linux	13.33%	**Polymorphic**	91.45%

VirusBuster is one of those products where the exact nature of the scanning is not mentioned in the process of installation – this being through the faceless method of an RPM package. The developers contacted me after submission, having discovered that there were issues with the *Samba* scanning functionality, which had a tendency to break after 10,000 files. In fact, problems were encountered sooner than this and the testing of on-access file interception was performed in small batches through the use of blocked copy operations. With these limitations in mind, the product's detection rate was very good and a VB 100% award is duly gained for detection. According to the developers the problems noted in scanning are no longer present in shipping products.

CONCLUSION

By the nature of the complex interactions required, on-access problems are at least understandable, if not forgivable. Testing puts unique strains on a scanning engine,

Hard Disk Scan Rate	Executables			OLE Files			Zipped Executables		Zipped OLE Files		Linux Files	
	Time (s)	Throughput (MB/s)	FPs [susp]	Time (s)	Throughput (MB/s)	FPs [susp]	Time (s)	Throughput (MB/s)	Time (s)	Throughput (MB/s)	Time (s)	Throughput (MB/s)
Alwil Avast!	31	17643.0		8	10438.7		21	7591.3	13	5739.0	3	10007.6
CAT QuickHeal	50	10938.6		13	6102.6		34	4688.7	13	5739.0	2	12866.9
DialogueScience Dr.Web	179	3055.5	[12]	11	7212.2		83	1920.7	14	5329.1	4	6755.1
Eset NOD32	33	16573.7		4	22666.8		21	7591.3	4	18651.9	2	16887.9
FRISK F-Prot Antivirus	81	6752.2		3	23333.5		43	3707.4	5	15226.0	2	13510.3
F-Secure Anti-Virus	119	4596.1		13	6102.6		89	1791.2	31	2406.7	6	4503.4
Grisoft AVG	77	7103.0		9	8718.0		52	3065.7	12	6217.3	10	2702.1
H+BEDV AntiVir	108	5064.2		5	16527.9		33	4830.8	6	12863.4	6	4740.4
Kaspersky Virus Scanner	148	3695.5		13	6102.6		67	2379.4	19	3926.7	7	4032.9
NAI VirusScan	77	7103.0		9	9015.2		60	2656.9	13	5739.0	5	5749.1
SOFTWIN BitDefender	586	933.3	[1]	6	12395.9		26	6131.4	7	10812.7	6	4289.0
Sophos SWEEP	56	9766.6		11	7212.2		35	4554.8	12	6217.3	4	6433.5
Trend ServerProtect	77	7103.0		6	13222.3		34	4688.7	7	10658.2	5	5404.1
VirusBuster VirusBuster	200	2734.7		7	11173.8		119	1339.6	13	5739.0	6	4579.8

which do not relate to any real-world situation likely to be encountered by users. Take, for example, the case where a *Samba* share automatically disconnects when many viruses have been detected. As part of a test scenario this may cause upset, but as part of a network where viruses must be contained, this behaviour may perform a useful function.

Informational problems, on the other hand, cause me far more grief and are multi-part in nature. Many products for this test were packaged using the RPM format which, due to its monolithic nature, does not allow for very much in the way of obvious documentation. Some developers packaged the RPM within a tarball, with a readme file as the sole other object present – this was very welcome.

It would be useful to know the location of the files which are installed. With there being no consensus as to the correct place for anti-virus software to be installed, the impression arrived at after this test was that all products wish to be unique in this respect. Installing to root, /local, /opt, /etc, /usr/lib, /usr/local/lib and many other locations gives a first-time user very little idea of where to locate their new scanner. Particularly irksome were those products which scattered components over four or more directories.

In addition, the basic command line to activate the scanner is rather a handy piece of information, especially if paths are not set up. On several occasions I searched for appropriate-sounding filenames in desperation, having no other clue as to how to initiate the scanner. This is particularly frustrating where daemons are required to be activated manually and are not mentioned at any stage in the documentation (if useful documentation exists at all).

On a more positive front, the overall standard of products has improved since last year, which is reflected in the number of VB 100% awards gained. As product lines become more stable it is hoped that the level of documentation will also show improvement.

Technical details

Test environment: Identical 1.6 GHz Intel Pentium machines with 512 MB RAM, 20 GB dual hard disks, DVD/CD-ROM and 3.5-inch floppy drive running *Red Hat Linux 9*, kernel build 2.4.20-8 and *Samba* version 2.2.7a. An additional machine running *Windows NT 4 SP 6* was used to perform read operations on the Samba shared files during on-access testing.

Virus test sets: Complete listings of the test sets used can be found at http://www.virusbtn.com/Comparatives/Linux/2004/test_sets.html.

A complete description of the results calculation protocol can be found at http://www.virusbtn.com/Comparatives/Win95/199801/protocol.html.