COMPARATIVE REVIEW

NOVELL SUSE LINUX ENTERPRISE SERVER 10

John Hawes

I approached my first attempt at VB100 testing under *Linux* with a little more than the usual trepidation. Despite some experience of running the open-source UNIX clone, my knowledge of anti-virus products for the platform was limited mainly to the exasperated rantings I had read in previous *VB* comparative reviews. The simple command-line scanners of old, I assumed, were fast becoming a thing of the past, with ever more sophisticated systems now providing the on-access detection that forms a central part of the VB100 testing methodology.

Alongside these advances I expected updating systems to keep products in touch with the latest discoveries in their base labs in the blink of an eye, and for the more corporate-oriented products I expected complex network administration and reporting systems to provide admins with control over the security of their many systems. As I embarked on the testing I could only hope that the baffling installation processes, obscure, incomplete or misleading documentation and generally bizarre behaviour reported by my predecessor would have long since been eradicated.

PLATFORM

The *Linux* operating system continues its steady movement into the mainstream, pushing its unruly way into the media and public consciousness with ever-growing compatibility, efficiency and usability. Novice-friendly distributions provide simple, out-of-the-box installation and a colourful and streamlined user experience, while server platforms – which have long been the most common implementation – have acquired the support and backing of major global concerns, even those that own and promote their own competitive UNIX flavours. Of course, for the legions of admins who prefer to get their hands dirty tinkering merrily under the bonnet, more serious distributions and bespoke versions are as popular as ever.

At the desktop level penetration remains fairly low, with most estimates placing *Linux* on less than 5% of systems. Servers, on the other hand – particularly web and mail servers – are much more likely to be running some kind of *Linux* implementation, with probably more than a quarter based on the open source alternative to UNIX, *NetWare* or *Microsoft*'s offerings.

SUSE (formerly *SuSE*) emerged in Germany in the 1990s and soon rose to prominence as one of the most widespread commercial distributions, particularly in Europe where it has long been the main rival to *Red Hat*'s global domination. While the fedora-themed distributor has blossomed in its own right as provider of a solid and supported platform, *SUSE* was acquired in early 2004 by *Novell*, and has since been marketed heavily and positioned as a solid base for a corporate network, with many of the tools and services provided by *Novell*'s other server platform, *NetWare*, ported across to *SUSE*-based systems.

The positioning of *SUSE* in the heart of the enterprise brings us to the basis of this comparative review. The old chestnut about *Linux* users not needing protection from malware doesn't cut it at the enterprise level, where file servers store and share data across networks dominated by more vulnerable *Windows* systems at the desktop level, serve up websites and process email for users around the world. More powerful server systems are a crucial layer of defence against infection, with on-access protection preventing uploading of dangerous files and scheduled scanning out of hours allowing more in-depth checking of shared data.

Installing *SUSE* has, for some time, been a simple and painless process, with a clearly designed GUI leading the user gently by the hand through the partitioning of drives and selection of software. The YAST management system, and its offspring, the YOU updater, provide similarly easy-on-the-brain methods of organising things, while the more technically minded can always get their hands dirty tinkering with config files and the like.

With the base systems set up and sharing some drive space via *Samba*, the other piece of software I expected to make extensive use of was the open-source *Dazuko* driver, developed by *Avira* and adopted by many other products for their file-hooking needs. With the kernel sources and other requirements in place, *Dazuko* proved easy and speedy to put in place, and the systems were imaged with a version precompiled as a kernel module and ready to insert at will.

TEST SETS

The latest WildList available at the deadline set for this test (1 March) was the December list, released in mid-February. Of the fair number of new samples added since the previous test, there were few surprises.

Another large swathe of W32/Stration variants joined their relatives in the set, along with a few more W32/Sdbot, W32/Bagle, W32/Feebs and W32/Areses. There were fewer than the usual number of W32/Mytob and W32/Rbot variants, a single nasty W32/Rontokbro, and a couple of new names offering pretty similar functionality. Beyond the worms, there were also a handful of W32/Looked variants, which vary between voracious infectors of just about anything they can find and more choosy types.

On-access tests	ItW		File infector		Macro		Polymorphic		Worms & bots		DOS		Linux		Clean set	
	No. missed	%	No. missed	%	No. missed	%	No. missed	%	No. missed	%	No. missed	%	No. missed	%	False positives	Susp.
Alwil avast!	0	100.00%	13	98.18%	18	99.56%	301	79.98%	1	99.57%	246	99.32%	8	83.33%		4
Avira Antivir	0	100.00%	0	100.00%	0	100.00%	3	98.72%	0	100.00%	32	99.78%	3	86.67%		
CA eTrust	0	100.00%	3	99.33%	12	99.82%	20	92.15%	0	100.00%	367	99.57%	12	53.33%	3	
CAT Quick Heal	0	100.00%	22	96.28%	73	98.23%	370	76.21%	0	100.00%	1120	90.75%	7	60.00%		
Doctor Web Dr.Web	3	99.64%	3	98.72%	19	99.61%	9	96.15%	6	98.70%	0	100.00%	4	76.67%		2
ESET NOD32 for Linux Server	0	100.00%	0	100.00%	0	100.00%	0	100.00%	0	100.00%	32	99.78%	0	100.00%	1	
Frisk F-Prot Anti-Virus	15	99.88%	0	100.00%	0	100.00%	0	100.00%	1	99.57%	0	100.00%	0	100.00%		
F-Secure Linux Server Security	0	100.00%	3	98.72%	0	100.00%	0	100.00%	0	100.00%	0	100.00%	5	73.33%		2
Grisoft AVG	0	100.00%	17	96.13%	0	100.00%	190	75.64%	2	99.42%	663	97.33%	7	65.00%		1
Kaspersky Anti-Virus for Linux	0	100.00%	0	100.00%	0	100.00%	0	100.00%	0	100.00%	0	100.00%	0	100.00%		
McAfee LinuxShield	0	100.00%	0	100.00%	0	100.00%	0	100.00%	0	100.00%	64	99.85%	0	100.00%		3
Microworld eScan AntiVirus for Linux File Servers	2	99.76%	0	100.00%	0	100.00%	0	100.00%	0	100.00%	0	100.00%	0	100.00%	2	
Norman Virus Control	0	100.00%	11	98.46%	0	100.00%	217	85.53%	0	100.00%	118	99.74%	6	66.67%		
Sophos Anti-Virus for Linux	0	100.00%	12	97.95%	0	100.00%	0	100.00%	0	100.00%	2	99.78%	7	71.67%		
Symantec AntiVirus	0	100.00%	0	100.00%	0	100.00%	0	100.00%	0	100.00%	5	99.97%	0	100.00%		
VirusBuster VirusBuster for Linux	0	100.00%	11	97.95%	3	99.93%	98	87.64%	0	100.00%	142	99.32%	9	66.67%		

In the other sets, the gradual revamping of the layout has seen a fairly major step this month. To link up with last time's rebranding of the 'standard' set as 'file infectors' to better reflect its contents, a new set of worms and bots has been added, populated so far with a selection of nasties removed from the old set and a few more recent additions. It is expected that this set will see a steady enlargement as samples of these common threats are acquired and added.

A second set also makes its first appearance this time, although with less up-to-the minute contents. Responding to recommendations that the many DOS samples in our test sets be removed (being of only minor significance these days), a sizeable chunk of these older threats have been plucked from their long-term positions. Abandoning them completely seemed a little extreme however, and would surely deny avid readers the valuable reflection of the in-depth strength of products – so they have been placed in a new set of their own, with a handful of additions thrown in to make good use of some stock waiting to be introduced. The decision to keep the DOS threats was justified by the reappearance, for the first time in many months, of DOS malware in last month's *VB* prevalence table – in very small numbers but from two separate data providers, indicating that some people at least are still exposing themselves and their precious data to these aged dangers.

As usual, the main bulk of the tests were carried out using the products' default settings. However, since some products ignore certain file types in their default settings, where possible, the archive speed tests were performed with archive scanning switched on (although, regrettably such an option was not always available, or at least not easily found). The aim was to compare like with like, and since the concept of 'default settings' is less clear with these predominantly command-line driven products, which expect plenty of qualifiers to tell them what to do, it seemed fair to tweak the settings upward rather than down, for those that needed it.

As a reflection of the increasing speed and capacity of modern hardware and scanning software, on-demand test results are presented this month in megabytes per second. In a further tweak to the presentation of figures, the on-access 'slowdown' figures are now calculated as the lag time added when accessing files. As the measurement is that of the time taken simply to open a file, and does not pretend to represent the overall system-wide effect of on-access protection, it is hoped that presenting the results in this way will provide a more useful indication of a product's overhead. Of course, any criticisms or suggestions regarding the data gathered and presented in these reviews is welcome (email john.hawes@virusbtn.com).

As a final nod to this month's specialist platform, *VB*'s set of *Linux* malware was revived, and alongside a few additions to the false positive set sits a batch of *Linux* files to add to the speed figures, the contents of the /bin, /sbin, /opt and a few other pivotal locations having been copied onto the scanning share. The on-access speed results for this special test set are a little problematic, as such files are

unlikely to be accessed from *Windows* clients in this way, and the large number of very small files results in a far greater scanning overhead relative to the size of the set. To allow the other data to be presented more clearly, the graph for this speed test is presented separately.

With all preparations completed, it was time for testing to commence.

Alwil avast! 4 v. 3.0.1

ItW	100.00%	**Macro**	99.56%
ItW (o/a)	100.00%	**Polymorphic**	79.98%
File infector	98.95%	**Worms & bots**	99.57%
DOS	99.32%	**Linux**	83.33%

Alwil provides its product in the form of rpm installers or more simple gzipped sets of files. I used the rpm method without problem, although this left me somewhat at a loss as to where the files had installed themselves. A brief search located them, with some simple documentation describing the use of the command-line scanner and the implementation of the *Dazuko*-based on-access component.

After a quick look through the usage guide, scanning was straightforward to implement, and detections zipped up the terminal window at an impressive pace.

Implementation of the on-access scanner failed silently at first, with no warning given that *Dazuko* needed to be inserted manually, but once up it seemed reliable and set a pleasing pace. Detection in both modes was reasonably thorough, with none of the new additions appearing among the smattering of misses.

The default setting for processing archive files, however, seemed to balk at anything too large or too deeply nested – to the extent of suggesting that several corrupted files could be 'decompression bombs'. Apart from this, a 'joke' program also found in the clean set was the only other issue, and with better results in the WildList set than in some of the more obscure collections, the product earns a VB100 award.

Avira AntiVir 2.1.9-37

ItW	100.00%	**Macro**	100.00%
ItW (o/a)	100.00%	**Polymorphic**	98.72%
File infector	100.00%	**Worms & bots**	100.00%
DOS	99.78%	**Linux**	100.00%

Not surprisingly, as its original developer, *Avira* also makes use of *Dazuko* in on-access mode. *Antivir* comes as an rpm, with a post-install configuration script to guide the user through the basic selection of settings. These include a request for the location of a ready-compiled *Dazuko* module, which is made use of when required.

Like most of the *Dazuko*-based products, on-access scanning is set on selected directories, rather than provided on the machine as a whole with exclusions needed for secure or sensitive locations, and these settings are adjusted in a configuration file.

Antivir also includes a graphical interface, which required a Java environment. Once this hurdle was overcome, the GUI proved pretty sophisticated, providing a thorough range of configuration options and scanning power, although on-access scanning could not be activated or switched off from here. There is also a rather clever graphical display of how many files have been 'guarded'.

The testing was carried out from the command line, a utility provided with a broad range of options; I was a little thrown at first until I realised that scanning did not recurse into subdirectories by default, but everything else was clear, and logging was laid out very simply and logically.

Scanning speeds were excellent, and only one particularly tricky member of the new set of DOS samples brought detection figures below 100%. With a full house of WildList detections, and no false positives, *Antivir* earns itself another VB100 award.

CA eTrust r.8.1.5310

ItW	100.00%	**Macro**	99.82%
ItW (o/a)	100.00%	**Polymorphic**	92.15%
File infector	99.85%	**Worms & bots**	100.00%
DOS	99.57%	**Linux**	80.00%

CA's product was the first to stray from the path of *Dazuko* and break its own ground for on-access scanning. It was also the first with rather grander pretensions, eschewing the simplicity of the command line and the config file for a system integrating with its cross-platform, centrally managed ITM system. Anticipating the benefits of familiarity, I was somewhat disappointed to find myself struggling with a rather tricky system.

The submission came in the form of the full contents of the distribution CD for *Linux*, UNIX and *NetWare* products. Browsing to the *Linux* section, I found an install script which, after making it executable and running it, took me through a sizeable installation process (including several EULAs which required scrolling all the way through before they could be accepted). Options for install locations etc. were run through, and the installation took place – a fairly

On-demand tests	ItW		File infector		Macro		Polymorphic		Worms & bots		DOS		Linux		Clean set	
	No. missed	%	No. missed	%	No. missed	%	No. missed	%	No. missed	%	No. missed	%	No. missed	%	False positives	Susp.
Alwil avast!	0	100.00%	12	98.95%	18	99.56%	301	79.98%	1	99.57%	246	99.32%	8	83.33%		4
Avira Antivir	0	100.00%	0	100.00%	0	100.00%	3	98.72%	0	100.00%	32	99.78%	0	100.00%		
CA eTrust	0	100.00%	1	99.85%	12	99.82%	20	92.15%	0	100.00%	367	99.57%	8	80.00%	3	
CAT Quick Heal	0	100.00%	20	96.79%	73	98.23%	370	76.21%	0	100.00%	1120	90.75%	7	60.00%		
Doctor Web Dr.Web	3	99.64%	3	98.72%	19	99.61%	9	96.15%	6	98.70%	0	100.00%	4	76.67%		3
ESET NOD32 for Linux Server	0	100.00%	0	100.00%	0	100.00%	0	100.00%	0	100.00%	32	99.78%	0	100.00%	1	
Frisk F-Prot Anti-Virus	15	99.88%	0	100.00%	0	100.00%	0	100.00%	0	100.00%	0	100.00%	0	100.00%		
F-Secure Linux Server Security	0	100.00%	3	98.72%	0	100.00%	0	100.00%	0	100.00%	0	100.00%	5	73.33%		2
Grisoft AVG	0	100.00%	0	100.00%	0	100.00%	190	75.64%	2	99.42%	663	97.33%	7	65.00%		1
Kaspersky Anti-Virus for Linux	0	100.00%	0	100.00%	0	100.00%	0	100.00%	0	100.00%	0	100.00%	0	100.00%		
McAfee LinuxShield	0	100.00%	0	100.00%	0	100.00%	0	100.00%	0	100.00%	64	99.85%	0	100.00%		3
Microworld eScan AntiVirus for Linux File Servers	2	99.76%	0	100.00%	0	100.00%	0	100.00%	0	100.00%	0	100.00%	0	100.00%	2	
Norman Virus Control	0	100.00%	9	98.97%	0	100.00%	217	85.53%	0	100.00%	0	100.00%	0	100.00%		
Sophos Anti-Virus for Linux	0	100.00%	12	97.95%	0	100.00%	0	100.00%	0	100.00%	2	99.78%	7	71.67%		
Symantec AntiVirus	0	100.00%	0	100.00%	0	100.00%	0	100.00%	0	100.00%	5	99.97%	0	100.00%		
VirusBuster VirusBuster for Linux	0	100.00%	8	99.23%	0	100.00%	98	87.64%	0	100.00%	142	99.32%	5	86.67%		

lengthy process. Browsing through the area mentioned in the installer, I found a command-line interface, which seemed inoperative, complaining about missing libraries. Checking the manual, I found much information on the centralised management utility, how to control vast networks of systems, but little on accessing any kind of client-end tools (although the broken command-line scanner was mentioned in passing).

I managed to find the ITM manager, a complex and bewildering thing, by checking some config files for the right port to point my browser at. I was able to sort out my on-access needs from there, but on-demand scanning seemed only to be available as scheduled scans – unsuitable for my speed test needs.

On contacting the product's creators, I was given the secret access point for the client end, which was familiar from previous tests on other platforms, and I managed to perform some more tests from here. Also familiar was the progress bar which dragged along each time a button was clicked, and I soon grew tired of it. I eventually found the missing libraries, enabling the command-line scanner and running the speed tests much more efficiently (and fairly) from there.

All tests recorded a good solid level of detection, and highly impressive speeds. In the clean set, however, something of an upset occurred, with no less than three files alerted on, all apparently infected with 'Antipas.653'. This was enough to deny *CA* a VB100 this time around, rendering all my struggles somehow all the more futile.

CAT Quick Heal 2007 v.9

ItW	100.00%	**Macro**	98.23%
ItW (o/a)	100.00%	**Polymorphic**	76.21%
File infector	96.79%	**Worms & bots**	100.00%
DOS	90.75%	**Linux**	60.00%

Quick Heal offered a pleasant return to the more simple side of things, and to *Dazuko*. Installation took the form of a simple zip file, with an install script within. This shepherded me through the setup process comfortably, and left me in no doubt as to how to go about running things. There is even a GUI, this time QT-based and requiring no further software to power it, providing clear and basic access to configuration and scanning.

There seemed to be few options regarding the on-access side of things, however, beyond the most basic on and off settings. As a result, *Quick Heal*'s on-access times are excluded from the archive table, which endeavours to compare like with like by running all products with archive scanning enabled, where possible. Nevertheless, decent speeds and reasonable detection were combined with a lack of false positives and exemplary coverage of the WildList set, thus qualifying *Quick Heal* for a VB100 award.

Doctor Web Dr.Web 4.33

ItW	99.64%	**Macro**	99.61%
ItW (o/a)	99.64%	**Polymorphic**	96.15%
File infector	98.72%	**Worms & bots**	98.70%
DOS	100.00%	**Linux**	76.67%

From a single file to many; *Dr.Web*'s installation was a rather more complex process, with several rpms provided to install the various components. Fortunately, a simple manual, as well as some tips from the developers, led me through the process of setting up the various daemons, scanners, another straightforward GUI, and the *Samba* integration. This was, it emerged, the first of several products to make use of the VFS functionality added to *Samba* in recent years to allow for file hooking, with a simple entry in the *Samba* configuration file directing all requests to the application of one's choice.

At this point the manual became less than helpful, the English version at least not having kept up with the latest increments to *Samba*; a table, matching up the pile of drivers provided by *Dr.Web* with the appropriate *Samba* versions, didn't include the version I had on my bare *SUSE* install. However, a little trial and error and the consultation of some logs soon had things moving.

The GUI was little help here, focusing mainly on the on-demand end, and as little control of logging was provided from here either, I stuck with the more fine-tunable command line for much of the testing.

On demand, speeds were a little less zippy than the previous few, and on access this was exaggerated, with the connection dropping occasionally and my file-opening utility reporting many files not opened. Running several retries and checking through the logs showed that none of these errors had been due to a false positive, although a couple of items were labelled as undesirable and another as adware.

More seriously, however, three separate variants of W32/Sdbot were missed from the WildList set, thus spoiling *Dr.Web*'s chances of a VB100 award.

ESET NOD32 for Linux Server 2.70.4

ItW	100.00%	**Macro**	100.00%
ItW (o/a)	100.00%	**Polymorphic**	100.00%
File infector	100.00%	**Worms & bots**	100.00%
DOS	99.78%	**Linux**	100.00%

Installation of *NOD32* was pleasingly well-designed, with an install script which made sense, and set things up just so. I barely even needed the simple instructions provided along with the product submission.

Once set up, the overall user experience was equally well thought out. While many of the other products in this review dropped their components into obscure locations with convoluted and unpredictable filenames, here I speculatively typed 'nod32' and got a nice polite response, urging me to provide some more specific options, while a standard –h call gave lucid and detailed information on usage.

Similarly, the on-access component was controlled by a proper init script in the standard location, responding to the standard instructions. *NOD32* was another product using *Dazuko* for its file hooking, and like the others in this class the on-access component was simple to set up, fast and efficient. The speeds recorded were even more eyebrow-lifting than usual, with the screen a blur of detections.

Sadly for *ESET*, my usual pleasure in using their product was marred, initially in a very minor way by missing one of the added sets of DOS samples (a strangely appropriate 32 samples, in fact), which spoiled a flawless record held by the product for some time now. More seriously, a false positive was generated in the older part of the clean set, caused by an apparently accidental upward tweak to the heuristics settings for DOS files in this build of the *Linux* product. Although the use of 'probably' in the log alert made the decision less than straightforward, rescanning the clean set with auto-deletion switched on resulted in the loss of the file in question, and combined with the commonness of 'probably' detections in *ESET*'s heuristic-heavy product, this was adjudged too severe to be classed as a mere 'suspicious file', resulting in *NOD32*'s first failure to achieve the VB100 for five years – its last having been the last time *VB* conducted tests on *SuSE Linux* in 2002 (see *VB*, April 2002, p.16).

Frisk F-Prot Anti-Virus 6.2.0

ItW	99.88%	**Macro**	100.00%
ItW (o/a)	99.88%	**Polymorphic**	100.00%
File infector	100.00%	**Worms & bots**	100.00%
DOS	100.00%	**Linux**	100.00%

F-Prot was another nice, simple product, with its files simply unzipped into /opt. *Dazuko* was again required for on-access scanning, although the absence of the module was not alerted on when running the product. Again, everything was simply configured via config files and scanning run from a pared-down command-line interface.

Speeds were fairly reasonable, and detection thorough almost across the board; unfortunately for *FRISK*, that thoroughness did not extend quite far enough, with one of the new variants of W32/Looked missed entirely while scanning the WildList set. The absence of any false positives more significant than the labelling of a *Sysinternals* tool as

On-demand throughput	Executables and system files		Media and documents		Linux		Other file types		Archive files	
	Time (s)	Throughput (MB/s)	Time (s)	Throughput (MB/s)	Time (s)	Throughput (MB/s)	Time (s)	Throughput (MB/s)	Time (s)	Throughput (MB/s)
Alwil avast!	147	11.07	24	51.89	86	9.40	24	0.57	32	22.96
Avira Antivir	83	19.57	21	57.20	61	13.15	13	1.00	87	8.41
CA eTrust	79	20.58	25	49.39	76	10.54	20	0.68	161	4.56
CAT Quick Heal	94	17.26	584	2.09	626	1.29	23	0.59	219	3.35
Doctor Web Dr.Web	259	6.27	267	4.57	308	2.61	68	0.20	342	2.15
ESET NOD32 for Linux Server	124	13.11	23	52.34	69	11.60	17	0.79	63	11.64
Frisk F-Prot Anti-Virus	158	10.25	37	33.29	33	24.30	12	1.09	76	9.60
F-Secure Linux Server Security	278	5.83	223	5.46	251	3.20	40	0.33	487	1.51
Grisoft AVG	163	9.97	35	34.77	144	5.59	17	0.79	159	4.62
Kaspersky Anti-Virus for Linux	229	7.10	154	7.93	243	3.31	25	0.53	454	1.62
McAfee LinuxShield	232	7.00	48	25.59	148	5.44	24	0.55	248	2.96
Microworld eScan AntiVirus for Linux File Servers	206	7.88	176	6.95	234	3.44	26	0.52	464	1.58
Norman Virus Control	909	1.78	24	50.96	218	3.68	41	0.32	135	5.43
Sophos Anti-Virus for Linux	97	16.74	26	47.24	57	14.03	10	1.40	93	7.91
Symantec AntiVirus	113	14.36	22	55.47	106	7.59	16	0.84	33	22.24
VirusBuster VirusBuster Scanner for Linux	166	9.76	94	13.00	139	5.80	23	0.59	70	10.52

undesirable could not redeem *F-prot* sufficiently to achieve a VB100 award.

F-Secure Linux Server Security 5.50

ItW	100.00%	**Macro**	100.00%
ItW (o/a)	100.00%	**Polymorphic**	100.00%
File infector	98.72%	**Worms & bots**	100.00%
DOS	100.00%	**Linux**	73.33%

F-Secure's product had a more professional feel than many, with some serious and thorough documentation. Installation took the form of a zip and an install script, featuring a selection of languages, EULA and licence code acquisition. There is also a web interface, which was typically crisp and austere, although some rather small fonts proved a little painful on the eye at the resolution setting I was using.

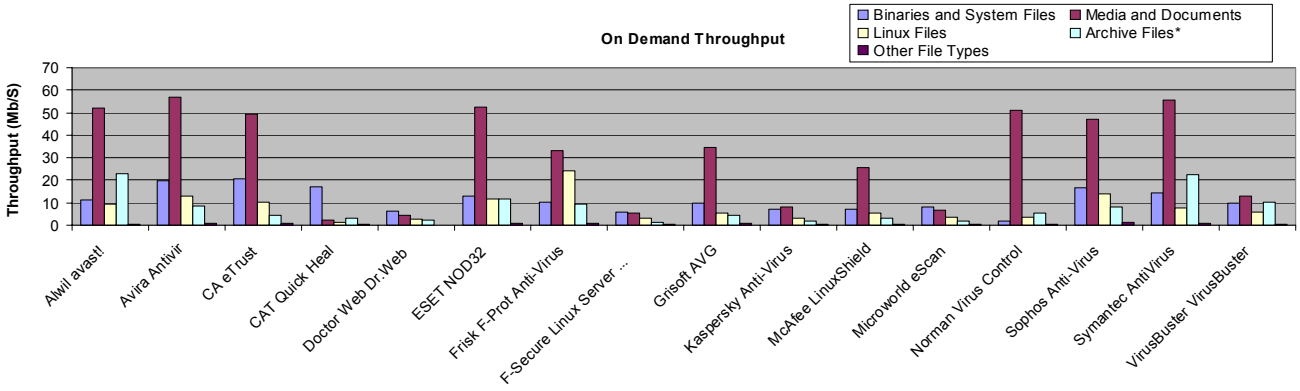
The command line was used for most testing, to ensure fairness in comparison with other products in the speed tests. However, speeds were not impressive, particularly once archive scanning was enabled on-access for the archive speed set. Viewing the logs showed that this could, in part, be due to the double scanning of all files, even once a detection is found, which would also account for the superb detection rates.

The only files the product missed were in archive types ignored by default (with some justification), and the alerts generated on two files in the clean set presented no challenge to *F-Secure*'s entitlement to a VB100 award – while one, the same *Sysinternals* pstools kit alerted on by many products, was described as a 'risktool', the other, an IRC client from *Microsoft*, was labelled, quite accurately, an IRC client.

Grisoft AVG 7.5

ItW	100.00%	**Macro**	100.00%
ItW (o/a)	100.00%	**Polymorphic**	75.64%
File infector	100.00%	**Worms & bots**	99.42%
DOS	97.33%	**Linux**	65.00%

AVG was another rpm-based installer, following which came a registration step with a special tool provided for applying a licence. A GUI is apparently available, although it was not included with the submission for testing. The command line proved more than adequate for my testing however, offering a nice, straightforward set of options, and the various scans were carried out



without difficulty. Scanning speeds were good, and detection was fairly decent too, with a few large sets missed in the DOS collection and a few in the polymorphic set.

Nothing was missed in the WildList set, although yet another undesirable item was spotted in the clean set, this time described as a 'Hacktool' (in fact, something designed to block advertising from an instant messaging client which has recently had some problems with serving up malware via its advertising system, which may be a bit of a hack but is also arguably a security benefit). However, this did nothing to spoil *Grisoft*'s chance of gaining another VB100 award.

Kaspersky Anti-Virus for Linux 5.5.9

ItW	100.00%	**Macro**	100.00%
ItW (o/a)	100.00%	**Polymorphic**	100.00%
File infector	100.00%	**Worms & bots**	100.00%
DOS	100.00%	**Linux**	100.00%

With *Kaspersky* we move away from *Dazuko* once more and into the murky world of *Samba* VFS objects, which have so far proved somewhat problematic.

The product was provided as an rpm, with an install script to run afterward for initial setup. In fact, a range of Perl scripts were provided for the configuration, including inserting appropriate entries into the *Samba* configuration file to operate the on-access side of things. Controlling the product from another browser-based GUI was apparently also possible, but as this required some third-party software to support it, it was not examined.

The command line once again proved more than adequate, with some rather off-the-wall syntax quickly mastered. On access, my fears about the use of the VFS functionality proved unfounded, with scanning as thorough and dependable as it was on demand.

Speeds were not electric – perhaps in part due to some vigorous attention to all manner of archive files – but detection was superb, with *Kaspersky* achieving the first unblemished record of the month. Not even a whisper of suspicion was raised in the clean set, with the only problem provided by a particularly large self-extractor, at which the product complained gracefully of an error while scanning. *Kaspersky*'s VB100 award is thus thoroughly deserved.

McAfee LinuxShield 1.4.0

ItW	100.00%	**Macro**	100.00%
ItW (o/a)	100.00%	**Polymorphic**	100.00%
File infector	100.00%	**Worms & bots**	100.00%
DOS	99.85%	**Linux**	100.00%

After my struggles with *CA*'s product, I feared a similar experience with another large, multi-faceted corporate-oriented product. This time around things were a little less troublesome.

A lengthy interrogation following the initial install demanded login details to access the obligatory web interface, and discussed web and mail filtering as well as file-based anti-malware. The web interface itself seemed fairly clear and comprehensive, but the fact that the page did not refresh proved to be confusing occasionally, leaving me clicking back and forth around the thing trying to discover if a task had completed. The updater task, achieved in my offline state by pointing a browse box at the location where the data was placed, seemed unable to spot the dat files, and in the end I resorted to dropping them in manually, which proved much more effective.

Scanning, carried out in part via the command line, involved setting up scanning tasks in the GUI, and then running them from the shell. The resulting speeds were possibly less impressive than a straightforward command-line scan might offer, but detection figures were

File access time lag	Executables and system files		Media and documents		Linux		Other file types		Archive files	
	Time (s)	Lag (s/MB)	Time (s)	Lag (s/MB)	Time (s)	Lag (s/MB)	Time (s)	Lag (s/MB)	Time (s)	Lag (s/MB)
Alwil avast!	529	0.32	164	29.60	264	48.40	403	74.38	85	14.89
Avira Antivir	113	0.06	27	4.09	202	36.83	28	4.20	93	16.33
CA eTrust	248	0.14	37	5.85	280	51.27	24	3.44	90	15.83
CAT Quick Heal	109	0.06	37	5.95	1319	245.57	31	4.75	(NA)	(NA)
Doctor Web Dr.Web	325	0.19	276	50.70	752	139.59	84	14.65	283	51.91
ESET NOD32 for Linux Server	123	0.07	28	4.20	250	45.78	23	3.34	61	10.50
Frisk F-Prot Anti-Virus	257	0.15	59	10.03	177	32.04	22	3.04	137	24.58
F-Secure Linux Server Security	204	0.12	41	6.61	428	79.08	35	5.48	452	83.46
Grisoft AVG	163	0.09	45	7.47	427	78.84	30	4.69	215	39.21
Kaspersky Anti-Virus for Linux	249	0.14	157	28.45	482	89.13	32	4.91	364	67.06
McAfee LinuxShield	284	0.17	68	11.78	347	63.91	39	6.38	143	25.77
Microworld eScan AntiVirus for Linux File Servers	(NA)	(NA)	(NA)	(NA)	(NA)	(NA)	(NA)	(NA)	(NA)	(NA)
Norman Virus Control	303	0.18	96	16.88	1252	233.13	98	17.35	(NA)	(NA)
Sophos Anti-Virus for Linux	160	0.09	44	7.19	294	53.97	30	4.64	144	25.90
Symantec AntiVirus	150	0.08	31	4.85	269	49.38	23	3.28	39	6.33
VirusBuster VirusBuster Scanner for Linux	808	0.49	79	13.71	2620	489.01	220	37.62	28	4.24

very good, with the only misses in the DOS set, mostly in the new batches.

With nothing from the more 20th-century sets missed, and certainly nothing in the WildList, *McAfee*'s handful of messages warning me about items I may not want in my corporate network do nothing to jeopardise its VB100 award.

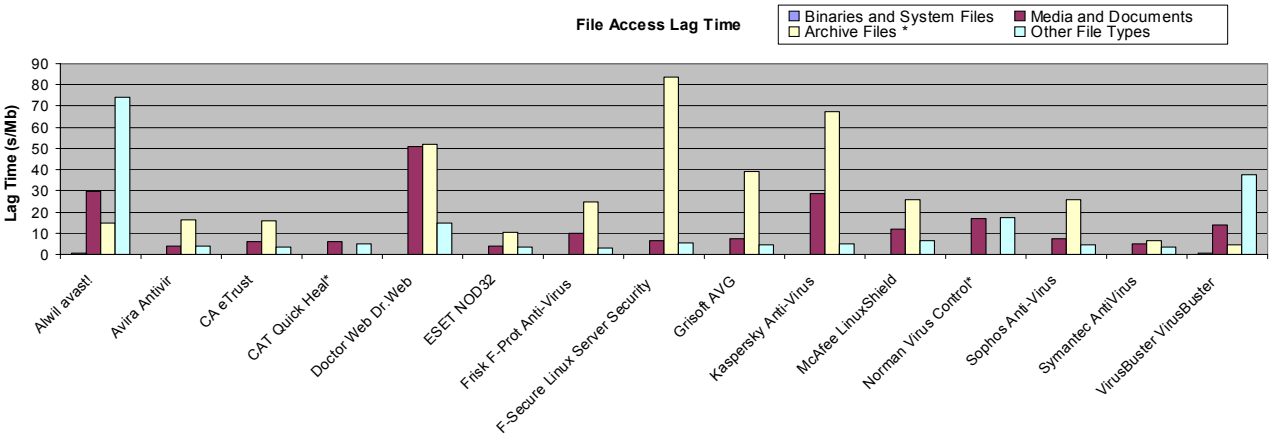
Microworld eScan AntiVirus for Linux File Servers 2.0.11

ItW	99.76%	**Macro**	100.00%
ItW (o/a)	99.76%	**Polymorphic**	100.00%
File infector	100.00%	**Worms & bots**	100.00%
DOS	100.00%	**Linux**	100.00%

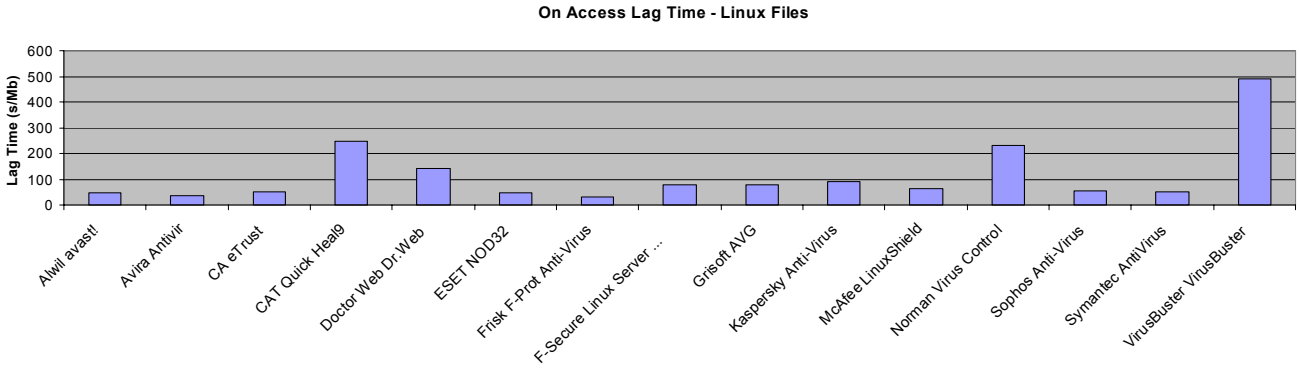
Microworld's product arrived as a swathe of rpms, along with strict instructions as to the order in which they should be installed. While most installed without problems, the web interface section got stuck several times looking for missing files; these I soon diagnosed as pointing to specific versions of items rather than the bare .so filenames, and some symlinking soon got it into a somewhat hacked state of running. An errant line regarding logging in one of the product's own config files also brought things to a halt, but I divined that commenting this out would cause no significant problem.

On-demand scans were carried out without further ado, although defaulting to disinfecting or quarantining infected files caused a moment of teeth-gnashing. Having assumed that my rather inelegant bullying of the web interface into operation would have little effect on my testing, I discovered that I did indeed require the GUI, as the documentation lacked detail on the syntax of the config files for some aspects of the product, notably the on-access scanner. This was once more a *Samba* VFS implementation, requiring several lines to be added to the *Samba* config, and once it was up and running I quickly saw that some scanning of files on access was indeed happening. Satisfied that scanning was in progress, I wandered off for refreshment, leaving it to chug slowly along through the first of the speed file sets.

Returning some time later, I was surprised to see it still going. Watching more closely, I noted frequent long periods of inactivity, with no files accessed at all. Running the scanner over the infected set was even more painful – despite having switched off the 'alert me when something is detected' option, a popup appeared in the *Windows* client for each detection, along with a warning 'ping' noise. Investigating the syslog, I found numerous complaints of a failure to



*May not be default setting.
Note: no archive scannng times available for CAT & Norman products.



quarantine files, with a message suggesting there might be a problem with access rights to the quarantine folder. However, checking the rights and expanding them proved no help here.

Looking further into the beleaguered *Linux* system, I found ever larger numbers of *Samba* daemons were being spawned, along with accompanying copies of the *eScan* daemon, presumably each time the scanning hit a snag.

With careful coaxing and splitting into chunks, I nursed the product through the collection, achieving some decent results over the full range of test sets, but unfortunately I had neither the time nor the patience to sit through the full range of speed tests. Before anyone complains that this gives an unfair advantage in terms of the chances of scoring false positives, I should say that the product had already lost its chance of a VB100 award, as both on access and on demand those pesky pstools and MIRC files were spotted and labelled clearly as viruses, which was enough to deny the product its prize.

But even had these unfortunate misnomers not been applied, the missing of two samples of W32/Bagle, introduced in the November WildList, would have been reason enough to withhold the award.

Norman Virus Control 5.70.01

ItW	100.00%	**Macro**	100.00%
ItW (o/a)	100.00%	**Polymorphic**	85.53%
File infector	98.97%	**Worms & bots**	100.00%
DOS	100.00%	**Linux**	100.00%

Norman's product came as another simple .tgz file, with a post-install script tucked away inside to set things up for me. After some initial tinkering, and the discovery that cleaning of files was the default, I soon had the on-demand detection and speed tests out of the way.

Unfortunately, configuration of the on-access files seemed to be via some config files in an obscure format. To continue, I required another interface, this time back to Java. Once this was in place, I was able to access a fairly simple, minimalist GUI, operating the configuration

controls only with no ability to run scans itself. It provided ample controls to get through the rest of the tests, although there was apparently no option to enable archive scanning on demand, thus upsetting my plan to include only on-access speed data in this mode. Despite this minor setback, *NVC* was generally easy to use and achieved decent levels of detection, with no false positives and spotting everything in the WildList set, thus comfortably winning a VB100 award.

Sophos Anti-Virus for Linux 5.70.1

ItW	100.00%	**Macro**	100.00%
ItW (o/a)	100.00%	**Polymorphic**	100.00%
File infector	97.95%	**Worms & bots**	100.00%
DOS	99.78%	**Linux**	71.67%

Sophos's *Linux* product uses its own alternative file-hooking system, released like *Dazuko* under an open-source licence. The product arrives as a .tgz file, with an installer inside, which checks the kernel version against a list of prepared builds of the driver. Apparently unsupported kernels are provided for by an on-the-fly compilation process built into the installer, but the SLES10 kernel was among those provided for in advance and installation proceeded without difficulty.

The browser-based interface proved pleasantly straightforward, simply laid out and responsive. For on-demand scanning the command line was used. Updating required implanting a large number of small identity files, which are then listed at the start of each command-line scan, and described in more detail when requesting version information, which required a considerable amount of scrolling up the screen to check the numbers, and may have added somewhat to the time taken to get each scan going.

Nevertheless, speeds were excellent, and detection impressive too, with a smattering of misses mostly due to archive scanning not being a default setting. *Sophos* also earns a shiny VB100.

Symantec AntiVirus for Linux 1.0.1.66

ItW	100.00%	**Macro**	100.00%
ItW (o/a)	100.00%	**Polymorphic**	100.00%
File infector	100.00%	**Worms & bots**	100.00%
DOS	99.97%	**Linux**	100.00%

Having expected *Symantec* to sit alongside its global giant rivals with a sprawling corporate-network product, I was surprised to find the product's version number so low, suggesting an immaturity which made me nervous.

Installing the product was no major issue, with a handful of rpms to run. Once this was done, I was at something of a loss as to how to get anything done, even having dug out the associated binaries tucked away under /opt. Some problems with the updater provided – which proved to be the wrong one for the platform under test – were resolved eventually, and in the process of installing and trawling the documentation for advice, I gradually picked up an idea of how things worked.

A central daemon supplies the scanning, with requests for on-demand scans passed into it through a tool which is also used to manage updating and checking up on the on-access part. Once scans are initiated, results are available only in the system log, although if the rather basic GUI (requiring Java) is running, detection reports are flashed on screen too.

The process of changing the configuration of scans, and of the on-access scanner, involves another tool which passes settings into the daemon's config database – not a simple config file but a binary file modelled on, of all things, the *Windows* registry. Indeed, at one point the manual seemed to suggest that the easiest way to set up the desired configuration would be to install a *Symantec* product on a *Windows* system, save the settings from there and export them to the *Linux* setup.

I eventually learned how to deactivate automatic disinfection, a process requiring two separate commands of over 150 characters each just for the on-access scanner, and chugged through the tests relying on the times recorded in the syslog for my on-demand speed results. In the end, very little was missed, and speeds were more than respectable, but would have been much slower had I included the time I spent puzzling over the control system. With no misses in the Wild, and no false positives, *Symantec* also earns a VB100 award.

VirusBuster VirusBuster Scanner 1.3.4/ SambaShield 1.1.3-2 for Linux

ItW	100.00%	**Macro**	100.00%
ItW (o/a)	100.00%	**Polymorphic**	87.64%
File infector	99.23%	**Worms & bots**	100.00%
DOS	99.32%	**Linux**	86.67%

VirusBuster's product comes in two separate modules, one for the on-demand scanner and another to provide on-access protection. The on-demand scanner was pretty basic: a bunch of files in a .tgz file, with updates simply dropped in on top of the existing files. Running from the command line brought up a warning

that the product was unlicensed, so I entered the code provided, assuming that this would be stored somewhere and not needed again. However, it turned out that the code had to be provided for every scan – I assume it could also be entered into a config file providing default scan settings.

Once this was figured out, scanning was no problem, although the logging was a little overzealous, recording everything so much as glanced at in the log file. When it came to the on-access portion, things got a little more fiddly, with several components installed to various places and some rather confusing information provided about how to set up one's *Samba* installation to redirect via another of those tricky VFS objects.

Once this was set up, a visit to my *Samba* share showed two lonely files, in English and Hungarian, informing me rather comically that my scanner was unlicensed and access to my files would be denied until this was rectified. A quick search located a config file where the code info could be entered and stored, and the expected set of folders returned to view after a restart of the *Samba* daemon. Testing proceeded at a somewhat leisurely pace, but detection was thorough and false positives pleasingly absent, allowing *VirusBuster* to add another VB100 to its tally.

CONCLUSIONS

The last time *SUSE Linux* found itself on the VB100 test bench (see *VB*, April 2002, p.16) was memorable for several reasons. It was not merely the last time one of the VB100's most consistent performers failed to make the grade, it was in fact the last *VB* comparative in which not a single award was issued. At the time, on-access scanning for *Linux* was in its infancy. In the intervening years, considerable ground has been made up, with a diverse range of systems – proprietary, open-source and integrated with aspects of the operating system – allowing products to control access to infected files. *Dazuko* in particular has proved a popular and successful option, and the many products that make use of it seem to have done so with considerable success. Other methods are less mature, and seem to have caused difficulties for some, although none so disastrous as to spoil anyone's chances of gaining the coveted award.

On the whole, the products fell into a few broad categories, in terms of both usability and implementation. Those that made use of *Dazuko* tended to be simpler, with more basic installation systems and interfaces, though some did offer full installers. Those attempting to take advantage of *Samba*'s VFS system tended to be meatier products, with more complex configuration required, while the chunky corporate products integrating their own methods of file-hooking were generally the most bewildering to operate, attempting to combine *Linux* products into a cross-platform offering, with varying degrees of success. Almost all offered some degree of automated updating, and most also had a GUI of some sort. *Linux* tends to be the domain of more technically literate administrators, who may prefer the flexibility and simplicity of command-line driven products, but the market for products designed for the less experienced user, more comfortable with an attractive graphical interface, is almost certainly the fastest growing end; it seems a pity that so many of these interfaces add more rather than less complexity to the process of configuring and administering anti-virus.

However, representatives of both the most basic and the most complex types of product managed to pass the tests and to do well in terms of speed, and there were delights and horrors at either end of the scale. It seems in many cases that usability and aptness of design are a reflection of a general company ethos, as many that have caused me trouble in their *Windows* incarnations were equally pesky under *Linux*.

As far as detection goes, after several months in which missing WildList viruses has been quite a common occurrence, it seems it is the turn of the false positive to rear its ugly head once more. Several products failed due to false alarms, while the 'suspicious' label which has long been allowed under the VB100 methodology has become ever more popular.

As more products move beyond adware and spyware into detecting legitimate and often useful software which could be put to malicious ends, a new category of 'toolware' is forming – one which may even be worthy of its own subset in our test collection. This would, of course, be rather difficult to populate and to make any useful judgements about, with such diverse opinions of what should be included. As long as it is made clear that such things are risky rather than innately malevolent, products are free to point them out as they please under the rules of the VB100. One product failed to do so, labelling such items viruses and was penalised accordingly, while several others had false positives in other areas entirely. The false positives will of course, like missed viruses, all be resolved with the vendors, for the benefit of their users, as soon as possible.

Technical details

Test environment: Tests were run on identical machines with *AMD Athlon64* 3800+ dual core processors, 1GB RAM, 40GB and 200 GB dual hard disks, DVD/CD-ROM and 3.5-inch floppy drive, all running *Novell*'s *SUSE Linux Enterprise Server 10*. Clients for the on-access test ran *Microsoft Windows 100 Professional*, Service Pack 4, on 1.6 GHz *Intel Pentium* machines with 512 MB RAM and 20 GB dual hard disks.

Virus test sets: Complete listings of the test sets used can be found at http://www.virusbtn.com/Comparatives/Linux/2007/test_sets.html.

ERRATUM: VB100 LINUX COMPARATIVE

Upon closer analysis of the latest set of VB100 test results (see *VB*, April 2007, p.11) *VB* has regrettably discovered some errors in the detection figures published for the *Dr.Web* product. A re-run of the tests demonstrated that the product was, in fact, capable of detecting all samples in the macro, file infector, Linux, and worms & bots test sets. However, the failure to detect a small number of samples in the polymorphic test set was confirmed, as was the failure to detect three samples from the WildList test set. *VB* extends its apologies to *Doctor Web* for these errors.