FACT, FICTION AND MANAGED ANTI-MALWARE SERVICES
VENDORS, RESELLERS AND CUSTOMERS DIVIDED BY A COMMON LANGUAGE

David Harley
National Health Service Information Authority,
Aqueous II, Aston Cross, Rocky Lane, Birmingham
B6 5RQ, UK

Tel +44 1743 357978
Email david.harley@nhsia.nhs.uk
david.harley@nhs.net • macvirus@dircon.co.uk

ABSTRACT

Not all of the assumptions upon which the malware management ethos is founded have changed since the 1980s. The anti-virus research community is aware of changes in malware technology, and in malware-management technology and methodology, as well as changing patterns of deployment and end-user attitudes to the problem.

However, security software is not always sold or administered by experts. The end-user community (system administrators included) varies widely in expertise and perceptual accuracy, of course.

Many organisations delegate their malware management deployment and maintenance to providers of managed services. However, experience suggests that a wide gap can exist between the expectations of the customer, and the range and quality of actual services provided, especially as project scale and the complexity of the protected environment increase.

Do researchers, customers, and product resellers offering one-fits-all management services share the same perception of what a 'complete' management solution is? Is the provider necessarily the best judge of best practice?

In this paper, we examine the full range of malware management functionality, and highlight some of the areas where dissonance arises between the customer's expectations and those of the supplier.

WHAT *IS* MALWARE MANAGEMENT?

Formerly, malware management was seen primarily as a desktop issue. Currently, however, it is often regarded as a systems management issue, focused primarily on mail gateways, reflecting the predominance of mass-mailers.

However, while multi-layering still offers maximum protection, managed services often focus entirely on gateway protection, and this is the area addressed most closely here. First, though, it is appropriate to examine the range of areas that malware management (in-house *or* managed) can be expected to cover [1].

Management of malware can be divided into two main areas: proactive and reactive, reactive measures being focused on incident management.

PROACTIVE MALWARE MANAGEMENT

This can be defined as including policy, standards and guidelines; education, training and information dissemination; systems and network administration; and development.

Policy, standards and guidelines

Conformance with external standards and certification: ISO17799 etc.
Conformance with legislation: data protection, computer misuse etc.
Formulation/implementation of internal standards
Internal policy development

It would be a brave salesman who declared that his product or service does not conform to all applicable legislation. However, organisations committed to ISO17799 compliance/certification are sometimes surprised at how difficult it can be to get a vendor to declare the same commitment in a formal contract. The canny contract negotiator must ensure that buying in a security solution of any sort does not compromise such conformance.

Similarly, the contract should not normally allow the supplier to evade customer-internal policy. This compromises not only security, but morale too.

Providers of managed anti-virus services favour (unsurprisingly) a minimal, automated approach to incident management and reporting, which may be at odds with the requirements imposed on internal units to return detailed reports of security incidents – such as those recommended by the Information Security Forum [2], for instance – with a view to management of those incidents and improvement of security posture.

These problems are less likely to occur if policy is left to a security vendor to whom all such functionality is outsourced. However, other problems are likely to take their place, and there must be doubt as to whether a one-fits-all set of policies can be more appropriate than a tailored policy set.

Education, training and information dissemination

Education and Training • Self-Education • Helpdesk/IS/Management/Staff Training
Setting up information channels. /dissemination of information 1. On/offline publications 2. Online services (Intranet web pages, mailing lists)

Here too, the trend, where responsibility is offloaded to a third party, is away from full-blown provision of a full educational service, towards the presumed provision of a complete and transparent management service requiring no significant action on the part of the serviced population. Service calls are diverted to the provider's service desk, resulting where necessary in an engineer callout within agreed service levels. This is reasonable, as long as the service provider has the competent personnel to handle such episodes, even when under pressure to deal with similar concurrent problems at many customer sites.

Furthermore, services supplied may range far beyond malware/security management, and expertise in these areas may not be evenly distributed throughout the support team. Reports abound of mishandled incidents to rival any badly trained, badly resourced in-house operation:

- Anti-malware software disabled because of conflicts with other applications also supported by the service provider.
- Server and client anti-virus software running obsolete engine versions or outdated definitions.
- Quarantined virus-infected systems reconnected to the local network/Internet to allow an engineer to download the latest definitions (rather than burning a CD), and consequently broadcasting malware.

MULTI-LAYERED SYSTEMS/NETWORK ADMINISTRATION

In virus management, this is essentially implementation and maintenance of multiple protective systems (virus-specific *and* generic).

The system levels under consideration include:

- Level 1: desktop protection, including remote users and home users whose home systems may interface at some point with their work systems.
- Level 2: workgroup/LAN server protection, including file and print servers, application servers, database, email and Intranet servers.
- Level 3: Internet/Extranet Protection, including perimeter devices, mail gateways, proxy servers, VPN hosts and other choke points.

(Adapted from [3])

There is still an argument for fully protecting the desktop, though this by no means protects the entire enterprise from all threats. However, the potential difficulties of distribution, installation, updating, and maintenance in any but the smallest organisations suggest gateway protection as a highly effective supplement.

Currently, the trend is away from multi-layering, and towards protection at a single choke point. This is only a viable complete defence where levels 1 and 2 are totally locked down and hermetically sealed by stringent firewall and VPN implementation from insecure protocols and backdoor/concurrent connections.

In practice, this degree of security is easier to accomplish with level 1 devices using secure operating systems than with home-user oriented operating systems. However, level 2 devices, though likelier to be furnished with relatively secure operating environments, require considerable maintenance and adherence to strict upgrade/patching regimes, while remaining sufficiently client-friendly to enable critical business processes.

Development

Many companies, if they think about these issues at all, hope that farming out the function to a third party will free up in-house personnel to concentrate on other areas of IT support. But will these functions be carried out to the same standard by third party implementers as an in-house team?

Comparison of installation and update roll-outs as practised or advised by managed service providers, anti-virus vendors, and in-house, sometimes indicates quite a different sense of priorities. Security vendors are sensitive to the difficulties of an accelerated development cycle, since they may have to produce stable definitions files, patches, or recompiled executables with a frequency and regularity inconceivable in many other areas of software development. However, surveying vulnerabilities-oriented mailing lists suggests that no application (certainly no network-aware application) is free from scrutiny or potential compromise. *Microsoft*'s transformation to a provider of security information and functionality is the prominent tip of a very large iceberg.

Customers who do their own roll-outs may have more faith in the stability of their chosen product, and be happier to take shortcuts, but tend to have mechanisms in place for tracking possible conflicts and implementing workarounds and rollbacks where the need arises. Customers with significant in-house expertise may be better able to test updates, patches and upgrades in the context of their own

site-specific systems and configurations than third parties. Providers, however, sometimes seem unaware that management of updates can be a significant problem. Such fixes are applied immediately and automatically, without on-site testing.

Consequently, it is not unknown for problems that are well-documented on mailing lists such as AVIEN/EWS and on vendor and other security websites and lists, to go unnoticed by service providers until drawn to their attention by their customers.

Often, the implementation of the product is shared between an anti-malware vendor, the developers of a wrapper application, and a partner systems integrator or solutions architect. If the implementer is not fully aware of the functionality and limitations of the product (not an uncommon scenario!), the result can be miscommunication at the specification and design stage where the customer's expectations are seriously divergent from the intentions of the developer.

Product evaluation
Configuration and functional testing
Performance and compatibility testing
Installation/rollout/update testing and execution
Incident management testing
Meeting threats the market doesn't yet address

Development areas

Many organizations avoid hands-on evaluation of anti-malware software, relying on third-party reports (consultants, comparative magazine reviews etc.) and selecting a product on these criteria and others, such as market share and cost.

Selecting a service provider and leaving the choice of software to him eliminates the need for hands-on evaluation altogether, but deprives the provider of the need to consider any range of solutions but those he already supports. Sometimes, support of a limited range of products and options renders the provider reluctant to undertake any improvement or customisation of the service, especially where he has little or no direct control over the development of the product.

On occasion, the augmentation of existing capability is discouraged by the service supplier, on grounds such as:

- Not within the current core capability of the product (and may be promised two or three major releases down the line!).
- Not immediately practicable because of cost and timescale of development.
- Cost implications of services not explicitly contracted.

The restrictive implications of such issues may be exaggerated. This author has been assured, mistakenly, by third parties that their AV engines lack relevant functionality. This may be due to lack of knowledge, reluctance to bother, or confusion between the capabilities of the 'wrapper' software and those of the underlying AV engine, exacerbated when developers adopt confusing terminology such as referring to the wrapper as the AV program, and the AV engine as 'the server'.

As a result, the customer with a cost-conscious Management Board and locked into a long-term contract may feel obliged to put up with a substandard solution.

REACTIVE MALWARE MANAGEMENT

Reactive management of malicious software is primarily incident management: firefighting, in a word, from the logging of a problem with the Helpdesk, through identification of the nature of the problem as malware-related, to taking appropriate remedial action.

The precise meaning of 'appropriate' depends on whether the incident is a hoax alert, a known threat identified at the point of entry and before the malicious code can be executed, a known threat *not* identified before code was executed, or a completely new threat. However, shades of grey may disappear entirely in a managed service. The hoax management function shrinks to handling helpdesk queries initiated by the end-user community, or referral to a web resource such as vmyths.com. No distinction is made between a virus caught at entry and a substantial on-site infection. Where the service is restricted to gateway scanning, notification to the apparent sender of an infective email is seen as sufficient, spoofing viruses notwithstanding. Dealing with infection on site may be seen as an exercise in public relations and referral to in-house crisis management: the service supplier may accept responsibility in terms of contractual penalty, but not for direct remedial action.

Compare this limited response capability to the range of incident management functions defined in a previous paper [1].

Incident logging/Reporting
Confirming existence of malware, if necessary by submission of samples
Disinfection/disinfestation
Dealing with direct damage
Advising sources of infection
Secondary Infection and damage
Secondary damage (including psychosocial issues)
Post-infection/disinfection/remediation/uprating
Dealing with false alarms
Hoax management
Regular Reports/Analysis
Re-evaluation

ANTI-MALWARE MEASURES

Anti-malware measures fall roughly into two areas: pre-emptive (proactive) and reactive. Reactive measures can be divided further into KVS (Known Virus Scanning), generic (which doesn't depend on the threat being precisely identified), and hybrid (incorporating elements of both approaches).

Proactive measures modify the environment pre-emptively so that it doesn't support a given type of threat, denying it an entry point.

Some proactive measures (CMOS configuration, user account privilege management) may, if implemented, be rolled out as part of the in-house IT management function, unless this has been completely outsourced (a scenario out of scope for this paper). A managed service may add value by recommending such measures, but is likely to be directly responsible only for measures that involve the direct use of recognised security software and, on occasion, the provision of alerts and advisories regarding new threats and vulnerabilities.

The proactive measure most likely to be implemented by a managed solution focusing on gateway protection is generic attachment blocking by file-type (or, less satisfactorily, filename extension). Essentially, this consists of flagging, quarantining or discarding file types perceived as 'dangerous'. In fact, most services (managed or in-house) nowadays tend to use a list like the following:

.bas	.bat	.chm	.cmd	.com	.cpl	.crt
.eml	.exe	.hlp	.hta	.inf	.ins	.isp
.js	.jse	.lnk	.mdb	.mde	.msc	.msi
.msp	.mst	.pcd	.pif	.reg	.scr	.sct
.shs	.url	.vbs	.vbe	.wsf	.wsh	.wsc

(Some also cover unlikely threats such as .CSV and .MPG.) In fact, lists like this are at the same time inadequate in terms of covering all potential threats and over-zealous in terms of the risk of blocking legitimate content. Robert Vibert includes a list of file-types indicating executable content that runs close to three figures, and is by no means fully-inclusive, and few organisations are likely to go that far [4]. At the same time, such lists don't distinguish in terms of risk threshold between the following groups:

1. Extensions hardly ever seen legitimately as attachments.
2. Extensions that might be legitimate, but the business case for allowing them is less persuasive than that for blocking them as a precautionary measure.
3. Extensions that might denote executive content, but blocking them would be more likely to cause major harm to the organization than the comparatively low risk of major virus damage.

Clearly, there is wide scope for disagreement on exactly which attachment filename extensions belong in which group.

While file-types such as .LNK, .PIF, and .SHS are never normally legitimate attachments, and therefore belong in group 1, policy-driven bans on exchange of screensavers may carry equal force. Including .EXE files in group 2 means that legitimate programs such as self-extracting or self-decrypting archives are hampered, sometimes to the serious detriment of business processes. Group 3 will include macro-bearing documents rather than raw executables, but there is still scope for disagreement. Some organisations regularly transfer *Access* databases, others refuse to deal with *Word* or *Excel* documents. The point here is not to argue for or against particular file types, but to illustrate the need for flexibility in such blacklisting, perhaps by implementing effective whitelists.

It is, of course, possible to circumvent these measures to some degree by strategies such as renaming files to resemble non-executable file types, archiving them as .ZIP files and so on. These carry their own penalties. They encourage measures that are open to abuse by malware authors and others using social engineering techniques to bypass security measures, and they may fall foul of technically more adept solutions that scan inside archives for executable content, or check actual file type rather than crude filename checking. Encrypting executables or archives to slip them past security software is also a measure available to the legitimate user and the social engineering virus writer alike.

INCIDENT MANAGEMENT

Unfortunately, use of terminology in security is notable for its inconsistency. (One man's worm is another man's virus or Trojan.) Presumably, the primary role of an anti-virus solution is to manage incidents, but what do we mean by an 'incident'?

The terms *incident* and *attack* are often used interchangeably in the security field, resulting in a certain amount of confusion. In computing, the term 'industry standard' is usually self-contradictory, but outside security, the following definition is probably close enough for most service desks:

> '…an 'Incident' is defined as
>
> any event which is not part of the standard operation of a service and which causes, or may cause, an interruption to, or a reduction in, the quality of that service.' [5]

Clearly, many of the incidents that are reported to service desks do not result from malicious action. Is the opposite necessarily true in the security arena?

'Security incident' is a term of which the meaning is apparently so obvious, it is hardly ever defined, or may be restricted to malicious action.

BS7799 does not include a definition at all, but refers somewhat similarly to 'security incidents and malfunctions' as if a malfunction were something different to a security incident. This is not an unreasonable position, as long as no one falls into the trap of thinking that software, systems, procedural and other organisational malfunctions are not within the security manager's remit. Even though this discussion is certainly primarily concerned with malicious action (that is, managing malicious code), the same principles apply: misinformation, misdiagnosis and malfunction are as much and sometimes more of a concern than actual malicious software, which is, in principle, relatively easy to control.

The Information Security Forum's 'Fundamental Information Risk Management' methodology very specifically addresses this risk, favouring the term 'Information Incident' over the term 'Security Incident' because 'it avoids preconceptions about the scope of information security.' [2]

The BS7799 standard does indeed include the control objective 'To minimize the damage from security incidents and malfunctions, and to monitor and learn from such incidents", which covers most of the management functionality we are concerned with here. For instance:

- Timely reporting of security incidents.
- Reporting of actual or possible security weaknesses and threats to systems and services.
- Reporting of software malfunctions.
- Quantification and monitoring of type, volume and cost trends of incidents and malfunctions.
- Implementation of a process for dealing with the violation of security policies and procedures [6].

Within the UK's National Health Service, a more rigorous, low-level definition is currently in use. A security incident is defined as, 'any breach or potential breach of information/ data security … Incidents may be internal, external or both …" [7].

The current trend here is towards integration of information security reporting with other types of incident such as clinical incidents and malfunction of medical devices. Information security breaches are classified in terms of confidentiality, integrity, availability, accountability and so on.

WHAT CONSTITUTES A VIRUS/MALWARE INCIDENT?

Harley *et al.* take a sternly utilitarian view of what constitutes a virus incident: 'Any case where a program reports a potential virus or Trojan symptom …' (Perhaps 'potential or actual' would have been better.)

A number of examples of possible indicators of virus action are quoted, but 'In such a case, it's perfectly legitimate to run an up-to-date and reputable anti-virus package under controlled conditions … In fact, it's legitimate to scan for viruses even where there are no perceived virus indicators whatsoever.' [8] This is an area of support where managed services seldom go unless automatic alerts suggest an unspecified problem.

ICSA Labs currently define a 'virus encounter' as an event in which any number of files, PCs or diskettes were infected at the same time. However, they acknowledge that some of the survey respondents may have '…used the number of files, PCs or diskettes infected as an "encounter" rather than our intended meaning…' Does this mean that each infected file, each infected PC, and/or each infected diskette, might be counted by end-user organizations as 'separate encounters'? Certainly, this maps to this author's past experience in the field. Trouble tickets may report virus infection problems such as:

- Several hundred viruses on a single system. Usually, this will turn out to be several hundred infectable objects (normally files!) infected with a single virus. Occasionally, there may be infectable objects infected with more than one virus (a not uncommon event when macro virus infections were at their peak), or individual infectable objects infected with different viruses. Much less commonly, anti-virus software will have reported dozens or hundreds or even thousands of infected/infective objects. This has occurred because of problems with the software resulting in a spectacular crop of false positives, but is more likely to indicate an authorised or unauthorised malware collection.
- An infected system which turns out to be uninfected, but where real-time virus detection software has issued an alert because a malicious program has been introduced to the system. The malware is there on the system, but is inactive (in some sense quarantined) and cannot be active with the anti-virus software in memory.
- A system on which one of the many non-viral objects, such as some joke programs and many Trojans, still flagged by some anti-virus software as a virus. In this scenario, the 'virus' may be active or inactive.
- A genuinely infected system that may require not only simple removal of infective objects, but some degree of rebuilding of the system.

A virus disaster was defined in previous surveys as an incident in which 25 or more machines experienced a single

virus on or about the same time. However, the definition has subsequently been expanded to include 'virus incidents causing their organizations significant damage or monetary loss' [9].

In terms of local management, the individual or team managing the incident can easily accommodate this degree of granularity, if they have a reasonable knowledge of the field. Does the same apply to a managed service? More to the point, is such discrimination seen as part of the contracted service?

A summarized report (tending to the retrospective rather than the timely in any case!) usually gives no information beyond 'infections per identified virus', as if each case of a detected virus were a separate incident. In fact, there may be a single infected system generating thousands of infective emails, especially where the infectee (and therefore the virus) has access to many thousands of addresses on a corporate list, and especially where mailouts are not capped. This is not the problem that central scanning usually addresses, but it may be worse than a limited infection or wide infection by a virus with a low impact virus. The mail storm continues and may even be exacerbated by the scanning service, even though each individual message is disinfected. By the ICSA definition, an incident can involve one infected system but thousands of infected massmailer messages: however, gateway software will report each message as an 'incident'.

WHAT DO MALWARE MANAGEMENT SERVICES OFFER?

Historically, commercial anti-virus software has mostly been focused on virus-specific scanning and the principle of user transparency (saving the customer from himself!). Virus-specific scanning has been more successful commercially than generic solutions that require the customer to make their own decisions about whether an alert (modification to a system file, for instance) indicates malicious code in action, a legitimate process, or happenstance.

Generic discarding or rejection at the gateway of objects that might carry (or entirely constitute) malicious code is more popular than quarantining of such objects pending intervention by an administrator. Managing a virus incident without human intervention is very attractive to cost-conscious management, quite reasonably. Indeed, while AV vendors continue to focus mainly on virus-specific solutions, corporate administrators have largely introduced generic blocking at least by filename extension, using non-AV solutions where their anti-virus solution of choice has not included sufficient functionality.

Vendors of managed services have responded to this culture with enthusiasm: manpower savings maximise profitability and may also be passed on to the customer, making the product more attractive. Few companies still cling to the belief that if they ignore the malware problem, it will go away and not bother them. However, most organisations embrace the hope that management of the problem can safely be passed on to (hopefully) competent third parties: the maintenance of the vehicle is left entirely to the mechanics.

In consequence, the trend has been away from organisationally proactive measures such as end-user training, and even the training of in-house support staff. Provision of information concerning new threats tends to be low priority. Customers are more likely to have to pull information from a website than to have it pushed at them, except in the case of the occasional high-profile threat whose prominence often has more to do with the whims of the media than an unusually low risk threshold. Mailing lists aimed at in-house support and administration staff often offer more of a branding and PR exercise than a serious attempt at assisting with risk assessment of new threats.

The underlying assumption here is that the core anti-virus functionality will deal automatically with all new threats. Indeed, where there is some form of generic blocking and/or advanced heuristics available, this is more or less the case, though the manner of processing may not be optimal, since it can hamper legitimate traffic. But is every incident being 'managed', or is the process simply *avoiding* management, as opposed to rendering it unnecessary?

There is more to dealing with a virus incident than cleaning or discarding an infected object. There is also a need to block (in so far as it is practical) the loophole by which the malicious program broke in, and to limit damage caused by any secondary infection prior to detection. Cleaning only the infection found is purely a matter of treating the symptom rather than the illness. Yet this is what many automatic systems do. The loophole in the case of an email virus, for instance, may be a whole peer organisation, customer organisation, supplier, internal unit, and so on. However, email scanning tends to be on the basis of a handful of scanning scenarios along these lines:

- No infection detected: pass it on.
- No specific virus infected, but contains suspicious object. Discard, quarantine, or pass on with warnings to sender and recipient. (Some services send the warning in a separate message which may or may not arrive before the infected message!)
- Disinfectable virus detected. Clean and pass on, or discard, or pass on with warnings.
- Non-disinfectable virus detected. Delete infectable object and pass on message, or pass on a standard advisory message, or pass on undeleted with warnings.

In each case but the first, it is usual to send warnings to the apparent sender and to the intended recipient.

In fact, a more realistic range of scenarios runs something like the following (very partial) list, raising a whole set of issues.

- No known malware detected in message. No attachment. Mail passed on. Transaction not logged.
- No known malware is detected in message/attachment. Attachment file type/file name extension turns out to be proscribed. Mail discarded, quarantined, or passed on but flagged as suspicious? Sender notified? Recipient notified? Transaction logged automatically? What level of detail? Transaction checked manually by servicing organisation? Customer notified to allow any necessary incident handling?
- Known malware is detected in message. Classified as infected but contains legitimate message content. Mail discarded, quarantined, or passed on but flagged as infected. Sender notified? Recipient notified? Transaction logged automatically? What level of detail? Transaction checked manually by servicing organisation? Customer notified for incident handling?
- Known malware is detected in message. Contains no legitimate content. Mail discarded, quarantined, or passed on but flagged as infected. Sender notified? Recipient notified? Transaction logged automatically? What level of detail? Transaction checked manually by servicing organisation? Customer notified?
- Message content suggests known virus. Attachment is infected. Mail discarded, quarantined, or passed on but flagged as infected? Passed on but attachment deleted? Sender notified? Recipient notified? Transaction logged automatically? What level of detail? Transaction checked manually by servicing organisation? Customer notified?
- Actually or possibly infective. Origin is internal to the organization. Any change in process?
- Actually or possibly infective. Origin is external to the organization. Any change in process?
- File encrypted: no meaningful scan possible. What action? Notification?
- Message digitally signed. Notification? Will it unhinge the signature?
- Virus is identified as spoofing virus. Any change in process? (Try to identify real sender? Notify apparent sender with caveat?)

How many managed services allow configuration according to the customer's response to all these questions? How many even expect the customer to think about them?

INCIDENT REPORTS

Good business security practice is to use data gathered routinely from security-related helpdesk logs to show trends and changing threats. Incident-led information gathering is the basis for future action. However, a typical managed service approach seems to make the following assumptions:

- The approach already implemented is optimal.
- Incidents are fully managed automatically.
- The point of information gathering is, therefore, to prove to the customer that everything works as it should. Reports of incidents managed need only be retrospective, aggregated data to take to the Board.
- There is no need to identify sources of infection. Notifying the sender is enough, and the sender is always the person in the From: or Reply-To field.

CONCLUSION

This paper is not about the 'best' way to run a managed service: it's for the customer to define what that service ought to be, and the vendor to ascertain those needs and implement them, or offer realistic alternatives. However, there are certainly some questions worth asking about a service before a customer buys in.

- How does it handle infection?
- Does it detect malware and take no action (alert only), delete everything, or clean what can be cleaned?
- If a virus is processed, there's an infected machine somewhere. Is the apparent owner of the machine notified? What if the virus spoofs? Does the service distinguish between spoofing and non-spoofing viruses and notify accordingly? If not, does it at least modify its standard notification message to take spoofing into account? Can notification text be configured or specified by the customer? A common measure is to include text to the effect of 'you may not be the actual sender of the virus at all'. Apart from making the service look very amateur, this may confuse the sender profoundly. Yet this measure seems to be adopted even by vendors who have the technology to discriminate. Whether this is to allow for errors in discrimination or an option that can be turned off to maximize throughput at the expense of 'correct' handling and meet speed-of-throughput SLAs is a question this author cannot answer.
- Does the recipient need to know about an infected message? Does the need to know vary according to

the virus (spoofing/non-spoofing)? Is it possible to turn off notification to the recipient if the infected message is discarded anyway? Does the product offer a choice? If the service provider also provides messaging services and charges on a message volume basis, do they allow for exemption from charges arising from the generation of multiple alerts? Can they notify the customer as well or instead by diverting recipient notifications to an IT administrative account to allow the customer to monitor virus activity and intervene if appropriate?

- What does the client organisation regard as a responsible *modus operandi* for dealing with an incident, and does it match the view of the service management supplier? Is it enough to block incoming malware, or only to inform the *apparent* sender of a spoofing virus? Does the software discriminate between spoofing and non-spoofing viruses? Does the wrapper/rules engine have the same level of discrimination as the AV engine?
- What logs are kept? Does the *customer* have access to full logs, or partial logs, or none? Managed email scanning services are frequently based on the assumption that every incident is fully managed, and that all the customer needs is a periodic summary with which to reassure the Board that the service is justifying its existence by blocking viruses. This is a fallacy: detection and disinfection are not the same as incident management. An organisation may decide that is sufficient, but the vendor should be aware that they are offering a severely limited service, and should ensure that the customer can make an informed choice about whether or not to accept that limitation.
- How regular are reports? How detailed? Characteristically, a managed service summarizes incidents over a week or month by number of viruses, and probably specifies the malware ID. Does it ID the infected system/user/mail source? A typical report shows number of 'incidents' per virus, but not the origin, which is useless for virus management purposes. If there is no correlation between source and virus, there is no way to tell if the apparent source was really infected or simply fingered by a spoofing virus. If there is no named source, no follow-up action is possible.
- Do they differentiate between senders within the organisation and external senders? Can notifications be regulated by domain?
- Do they scan at the email gateway inbound and outbound? (Do they assume that all malware is inbound?)
- Do they scan archives? Do they scan nested archives? What archive formats do they scan? How do they treat archives they can't scan?
- Do they block generically? What do they block, and how (file type or filename extension)?
- How do they handle encrypted archive files (.ZIP etc.) and other encrypted files? (Scan and clear, scan and flag, scan and discard or quarantine?) How do they handle encrypted messages?
- How do they handle signed messages?
- Do they recognise organisation-specific EDI transaction sets?
- What degree of configurability does the service have? Does it match the configurability of the AV engine?
- What handling choices are you offered? Block/discard? Quarantine? Pass through flagged?
- What do they do manually to maintain the efficiency of the service? Do they routinely follow up alerts, or only when a problem is flagged? Do they boost the volume of alerts with aggressive EICAR testing?
- Do they include any form of traffic analysis to flag massmailer activity and other potential abuse? (Hoaxes, chain mail, spam, accidental and amateur spam and so on.) For the customer, it no longer makes sense to differentiate rigidly between viruses and email abuse (Cf. Friendgreet), if it ever did. Nowadays, spam specialists offer bolt-on AV or vice versa, but this may be a Swiss army knife rather than an effective toolkit.

REFERENCES

[1] David Harley, 'Managing Malware – Mapping Technology to Function', *EICAR conference proceedings*, 1999.

[2] Information Security Forum, *Fundamental Information Risk Management*.

[3] Ken Bechtel, 'Anti-Virus Defence In Depth', *SecurityFocus*, 2003.

[4] Robert Vibert, *Enterprise Anti-Virus Book,* 1st Edition, Segura Solutions Inc., 2000.

[5] *Best Practice for Service Support*, CCTA IT Infrastructure Library: TSO, 2000.

[6] BRITISH STANDARD BS 7799-2:2002 Information security management systems – Specification with guidance for use: A.6.3.

[7] *NHS*net, Incident Reporting and Investigation Support Security Operating Procedure: internal documentation 2002–2003.

[8] David Harley, Robert Slade and Urs Gattiker, *Viruses Revealed*, Chapter 10, Osborne/McGraw-Hill, 2001.

[9] *ICSA Labs Virus Prevalence Survey 2002*, TruSecure.