Content Filtering: Sifting Through the Mess

NASA SEWP Security Center
Aaron Powell
Christopher Vincent

July 20, 2006

DISCLAIMER: This document is intended for informational purposes only and is no substitute for performing one's own analysis of the products and solutions discussed herein. It represents the NASA SEWP Security Center's analysis and opinions. There are no express or implied warranties regarding the veracity of the information provided. When implementing any content filtering solution, it would be wise for one to seek legal counsel beforehand.

Introduction

As business, government, and non-profit organizations continue to combat information insecurity, they have made some headway against common problems that once plagued network administrators. Regular anti-virus scans, locked-down firewall rule-sets, persistent log monitoring and strong password policies have reduced some of the more egregious offenses against information technology infrastructures. As a result, browser-based vulnerabilities have received increased scrutiny as a possible avenue for improving security.

In response to these threats and others, vendors are marketing numerous suites of products claiming to protect end-hosts from attacks via the web and web-related services. Their products, both hardware and software, are usually referred to as "content-filtering" products. Web filters have existed for some time, primarily as a means to prevent children from viewing material deemed inappropriate and to prevent users on public machines from accessing illegal material, but this new generation of products hopes to address newly identified problems as well.

Currently, organizations face several different problems that they hope to address via "content-filtering." First, organizations hope to prevent authorized users from transporting and storing illegal materials via their networks and machines, possibly creating liabilities on the part of the company. Second, organizations hope to prevent authorized users from wasting company resources by using the internet for legal non-work related purposes. Finally, organizations hope to protect their equipment from vulnerabilities introduced by spyware, adware, viruses, and other malware, including malicious web content that exploits vulnerabilities in the web-browser, itself.

These problems are distinct but related because the same sites that host illegal or productivity-reducing content often include spyware in their products and use browser-based vulnerabilities to attack the hosts that visit their sites.

The network administrator has several options as to how to implement a content-filtering solution. A) The filtering can be done inline, by dropping unauthorized packets, or by eavesdropping and killing unauthorized connections. B) The filtering can be performed via a devoted hardware device or via software installed on a device. C) The filtering policy can be applied uniquely to individual workstations or to the organization as a whole. D) The filter can look only at the means of communication (the URL and the

port), or it can look at the contents of the communication by parsing and processing the payload of the packets in transmission.

In all cases, filtering relies on large databases of current information about possible threats, either in the form of a list of unauthorized URLs, a list of unauthorized ports and IP ranges, or a set of signatures of known or potentially unauthorized content. The databases require constant updating to remain current and effective and some proportion of the systems resources will be devoted to contacting a centralized server to obtain new URL assessments and new signatures.

Technology behind Content Filtering

Inline Filtering vs. Eavesdropping

Filtering products achieve the desired effect of interrupting unwanted flows of information in multiple ways. If the filtering product sits on the gateway route (or on the actual gateway), the product can filter by dropping any packets that have an undesirable characteristic. For example, if a content filter receives a packet originating at “www.apple.com” and the filter is configured to block “www.apple.com,” the filtering application can erase the packet instead of forwarding it to the host. This is how a typical firewall operates. In addition, firewall rule-sets typically perform some basic sanity checks to prevent local packets from misidentifying their source IP address in some cases (sometimes referred to as ingress/egress filtering). Packets with inaccurate information or packets coming from blocked IP ranges or on blocked ports are simply dropped.

A filtering product can also interrupt the flow of data if the filter has access to all the traffic on a network (i.e. eavesdropping) and can insert its own traffic onto the network. For example, if a content filter is attached to a hub (rather than a switch), the content filter can "read" all the traffic traveling to and from the various hosts connected to the hub. Whenever it sees an undesirable TCP connection involving a local host and a remote machine, the filter can fire off a "HTTP redirect" to the local host (assuming the hosts are communicating via HTTP), and then send TCP RST or TCP FIN messages to both the local and remote machines, closing the connection.

There are advantages and disadvantages to both these approaches. One advantage of inline filtering is that the inline product can prevent any and all communication to a "blacklisted" remote machine, as identified by an IP address. The inline product will simply drop the attempted connection, log it, and possibly serve a message to the offending local host indicating that the connection was not established. On the other hand, if the filtering process is slow, the filtering gateway might introduce transmission delays if it is forced to analyze large amounts of content on the fly. Inline filters are prone to become detrimental bottlenecks adversely effecting both good and bad data in transmission. Additionally, should an inline filter “fail closed,” then all access outside of the gateway would halt; this may or may not be desirable and should be considered in light of an organization's availability, security, and/or legal requirements.

Eavesdropping filters do not suffer from the single point of failure / bottleneck problems that plague filtering gateways, but they come with their own limitations. First, eavesdropping filters require a network configuration such that all traffic can be viewed simultaneously from one point. A simple hub under the original conception of Ethernet traffic automatically provides the necessary functionality for complete eavesdropping. Unfortunately, most contemporary Ethernet networks are configured such that individual end hosts cannot eavesdrop on traffic that traverses a switched network. In some networks, it might pose a significant engineering challenge to configure the network so that one connection can eavesdrop on the entirety of the network traffic.

Further, the eavesdropping filter's ability to terminate an unwanted connection relies on the correct behavior of the local and remote machines. If the local and remote machines ignore TCP resets and HTTP redirects, the malicious traffic can still enter the network and potentially the host. Even if the local and remote machines respond appropriately, an overloaded eavesdropping filter may react too slowly to prevent the transmission of a malicious payload.

Finally, some protocols, such as UDP, cannot be filtered by an eavesdropping device. An eavesdropping filter has no way to halt the flow of connectionless traffic, like UDP because there is no connection to reset. Thus, the only way to filter UDP-type protocols is via an inline device.

Hardware vs. Software

Typically, people believe that hardware solutions are faster than their software counterparts. Often, hardware vendors rely on this belief to market their products and make claims about the limitations of software products. However, it is not necessarily the case that a hardware solution is faster than a comparative software solution. First, any software-based content-filtering solution is going to have to be installed on some hardware device. That device could be faster, or slow, or the same as any hardware solution. Also, software solutions are not tied to proprietary hardware which can result in lower up-front and long-term maintenance costs. Additionally, some of the software solutions allow spanning across hardware thus giving two advantages: better continuity of operations and capacity scalability. Should a hardware device fail or additional capacity be necessary, a cheap COTS PC can be purchased and put in place.

Certainly, if a hardware device is specially manufactured to optimize the processing of high volumes of network traffic it is likely that such a hardware device would produce better results than a software solution installed on a generic machine. Unfortunately, hardware vendors may falsely imply that their hardware products are customized solutions designed specifically to handle the task of content filtering while they are, in reality, generic rack-mount PCs load with proprietary software that cannot be purchased separately. This misleads the purchaser into thinking that the hardware device differs from a standard PC when it does not.

URL Categorization and Port blocking

Many so-called "content-filtering" products are not actually "content-filters" in any strict sense of the phrase. They do not look at the contents of the payloads being transmitted. Rather, they perform "URL categorization-based filtering" to prevent machines from accessing specific web resources and "port blocking" to prevent machines from using certain services or protocols. URL categorization-based filtering relies on a database of URLs which associate every conceivable location on the internet with one or more subject-matter categories (as well as categories for sites that are known to serve malware). The filter checks the source or destination of the traffic to see if that source or destination was "blacklisted" or "white listed" and allows or blocks the traffic as necessary.

For example, “www.ebay.com” might be categorized under “online auction sites” which might be a subcategory of “shopping sites.” If an administrator blocked “online auction sites” or “shopping sites” traffic to and from any site in the category, including “www.ebay.com,” would be filtered.

As with URL categorization, port blocking relies on a small database of known protocols defined by the TCP/UDP ports over which they operate and, if applicable, the IP addresses they require to communicate. The filter merely blocks traffic to ports over which the banned services normally operate.

There are several limitations to URL categorization and port blocking as a means to control content. First, a given URL must be in the database and correctly categorized for content filtering to function appropriately. If your network’s users are the first users to visit a site with problematic content, your network becomes the guinea pig for the rest of the vendor's clients. Further, naive port blocking only looks at the standard ports over which a protocol communicates. Traffic can easily circumvent a port-blocking filter by running a protocol over an open non-standard port.

On the other hand, port blocking on an eavesdropping filter provides a network-protecting mechanism that is unavailable on an inline filter because it can terminate unauthorized connections between machines on the same network. As a result eavesdropping filters provide some benefit that exceeds the typical functionality of the gateway firewall. This same effect can be achieved, though, if all machines internal to the network are running locked-down local firewalls, themselves.

One oft-touted property of URL-categorizing filters is their ability to block spyware, adware, and other malware. It is important for a network administrator to understand how this filtering is accomplished by a pure URL categorizing filter. The filter, itself, is completely unaware of the contents of any payloads traversing its networks (except the limited file-type identification capabilities included in some filters.) The filter only looks to the URL source and destination of monitored traffic and compares it with a list of sites that are known to serve malware. If malware comes from an unknown, or that is labeled innocuous, the filter will not detect it, even if it is well-known malware.

Signature-based Content Filtering

One way to reduce the propagation of malware (including spyware and adware) is through the use of signature-based scanning techniques. In signature-based scanning, the scanning device relies on a vigilantly updated database of "signatures" of known malware. The database stores snippets of binary code sampled from known malware. The filter searches the payloads of all the packets it encounters to see if they match known malware patterns from the database. If the packet matches a known malware pattern, it is dropped. In some cases the database contains the "hash function output" of the known malware snippet rather than the malware code, itself, because hash function outputs are smaller and cannot be reversed to recover the original malware snippet. This reduces the size of the database and speeds the pattern-matching process.

Unfortunately, when signature-based scanning is performed at the gateway, it reduces the gateway's throughput by increasing the latency of all the data scanned therein. (Any average computer user is aware of the time-intensiveness of traditional anti-virus scanning.)

Signature-based scanning has the capacity to discover known malware, but may be unable to block novel code if the code is substantially different from previously encountered malware code. Whether a system administrator chooses to implement intensive signature-based scanning on her network will depend on the service needs of the network's users. In some cases, such as Aladdin's Esafe suite, network administrators can select which traffic to scan and invoke quality of service policies to ensure that timely traffic delivery preempts malware scanning for critical content.

File Type Identification

Some measure of security can be achieved by limiting the transmission of data into the network based on the type of file that is being transmitted. For example, many organizations do not need to allow video files, audio files, or executables to traverse their gateway during normal operations. (Obviously, exceptions exist.) A content filter can prevent such files from entering the network, thus preventing the unintentional installation of malware or the appropriation of illegal content.

File-type identification is not an easy task. Naive file-type identification systems look only to the filename extension attached to most filenames. For example, ".doc" usually indicates that a file is a MS Word Document. Most computer systems have no means to prevent filenames from being changed arbitrarily (beyond a few limitations that apply to all filenames) so the mere presence of a ".doc" extension does not guarantee that the file is actually a MS Word Document. Merely relying on filename extension is easily circumvented.

Better file type identification can be determined by analyzing the contents of the file itself. All Microsoft ".exe" files (executable programs) begin with the ASCII string "MZ". Similarly all compiled java programs begin with the hexadecimal string

"CAFEBABE". Thus file type identification that looks beyond filename extensions alone can prevent simple circumvention by name change. The task of actually categorizing the characteristics of every conceivable file type is daunting, though, because identifying information within a file can be non-unique, it can change between different versions of a program, and is not standardized.

Deployment Considerations

Effectiveness

The major motivations for installing content filters are 1) to prevent illegal material from entering the network; 2) to reduce access to legal non-work related materials; and 3) to prevent malware (including browser-based vulnerabilities) from adversely affecting machines on the network. The technologies' ability to solve the problems outlined varies.

Content filtering via URL categorization (and to some extent, port blocking) stands a good chance of preventing known illegal material from entering the network, especially if the source of the illegal material is sufficiently well-publicized and has already been added to the URL database. Unfortunately, hosts serving illicit materials move frequently to avoid legal recourse and network blockage. Signature-based content filtering also has the capacity to block illegal material, but would require a company to amass huge amounts of illegal material and create signatures of it--an unlikely circumstance. Further, other kinds of filtering, including the use of "keywords" to search ASCII-encoded text, might help but can produce large numbers of false positives while failing to block some illegal content.

Another motivation for introducing content-filtering products is to increase productivity by preventing users from accessing legal, non-work related web content. Some statistics suggest that more than 30% of internet usage in the workplace is not work-related. Certainly, blocking web sites prone to egregious offense (such as online gambling, online video games, etc.) has the capacity to prevent workplace problems and increase workplace productivity. Fundamentally, though, worker productivity is a management and personnel issue rather than a technology issue. Managers should be aware of what their subordinates are doing, what tasks their subordinates are responsible for, and how much time those tasks should take, so they are aware of whether their subordinates are working or not. Blocking internet traffic to expansive arrays of web sites, especially in an enterprise environment is more likely to produce a large volume of help-desk calls and complicate the content filter's rule set unnecessarily when increasing numbers of exceptions need to be carved out. Even if a content filter blocks all unnecessary internet content, the unproductive worker can always play solitaire or read the newspaper he/she brought to work.

Malware, like illegal content, can be significantly reduced through the use of a content filter. As mentioned, URL categorization requires a database of known malware sites, so networks are vulnerable to any sites that are not yet in the database. (Also, network administrators are at the mercy of the update schedule of the product vendor. This is not

a catastrophic issue, but it does reduce the capacity of the content filter to address new vulnerabilities.) Further, port blocking can be tricked by crafty malware that communicates via non-standard ports.

Browser-based vulnerabilities present some unique challenges not found in other contexts. Whereas many attacks exploit unprotected machines running flawed (or unpatched) services, browser-based vulnerabilities exploit the internet-browsing software itself. For most networks, the port over which typical internet services run is always open. Thus, a firewall is a poor defense against browser-based vulnerabilities. URL categorization can prevent known attacks, and inline signature-based content filters can provide some protection from known exploits coming from novel sources. However, this protection against new sources comes at a significant performance cost to the network because the filter needs to scan the content before transmitting it to the final destination, potentially introducing latency into the network. In addition to signature-based filtering, one possible aid to the problem of browser-based vulnerabilities is better browsers.

Legal Issues

Emphasis on URL categorization could create some legal issues if a product or its implementation is deemed discriminatory in its filtering practices. For example, Websense allows an administrator to block sites based on various topics, including religion. If one chooses, one can delve further into the category hierarchy to block either "Traditional Religion," or "Non-Traditional Religions and Occult and Folklore." Unfortunately, the application of these categories appears inconsistent and arbitrary. While numbers of adherents do not strictly define "traditional" versus "non-traditional" religions, one major category of religious sites was categorized as “non-traditional” even though there are hundreds of thousands of self-described adherents in the United States alone. In addition, Websense allowed a connection to the web site of one religious organization while blocking a web site of former adherents who criticized that organization. It is easy to imagine an organization finding itself liable for discrimination if it enforces restrictions against one group of religions and not another.

In addition, content filters allow administrators to apply content filtering rule-sets to particular machines, groups of machines, or the entire network. This could cause problems if the blockages are enforced unequally within the organization. While categorization of popular online gambling sites and other clear "time-wasters" can increase productivity and blocking known sources of malware can protect the network from infection, blocking content in a discriminatory fashion, especially content related to people's beliefs or political opinions, raises serious legal concerns and has the potential to damage workplace functionality.

False Positives / False Negatives

One major danger in using URL categorization is the occurrence of false positives and false negatives. A false negative condition would allow malicious, illegal, or other undesired content to traverse the network while a false positive condition would block

work-related, mission-critical, innocuous, or allowed-but-controversial content. The self-proclaimed 97% accuracy of one product would be sufficient to create numerous help desk calls and retard productivity in a large environment with significant traffic.

In addition, the granularity of the filtering system may be too coarse to safely distinguish between safe web site about a controversial topic and unsafe web sites involved in the controversy. Websites that host security related information for security professionals, including snippets of malicious code, links to compromised sites, and descriptions of vulnerabilities might also be blocked (especially when using keyword or relational blocking) reducing the ability of legitimate researchers and system administrators to access necessary resources.

Solutions Considered

CPSecure Content Security Gateway (CSG) Series

CPSecure's CSG Series is a line of hardware-only appliances that offer what the company refers to as “stream”-based filtering. They claim that stream-based filtering introduces less latency into the flow of network traffic than other types of filtering. The hardware device resides inline, at the gateway, behind a firewall but in front of an organization's hosts. Unfortunately, we were unable to procure a test unit and cannot verify the company's claims. We could not find any literature describing specifically how the "stream" scanning works from a technical standpoint, so its effectiveness is not known. The device appears to be signature based and receives daily updates like most other products in its class. Its distinguishing feature seems to be its reduced latency compared to similar products. CPSecure does not advertise the cost of its products but its licenses include support for an unlimited number of users.

Blue Coat WebFilter

This hardware product is a URL inspector that watches for known bad sites. Its URL database is grouped into numerous categories and supports the creation of custom categories. Setting this product apart from other URL-filtering products is its "Dynamic Real-Time Rating (DRTR™)" algorithm that analyzes web page content in real time for any site that is not in the database. WebFilter evaluates an uncategorized site based on vocabulary contained within it and the categories of known sites to which it links.

We could not determine the accuracy of WebFilter’s algorithm, its ease-of-use, and its resistance to false positives. Administrators should consider whether simple URL filtering alone is sufficient to protect against malware- based threats against their network. Database updates are loaded "on demand." With malware, spyware, and phishing, WebFilter does offer some smart options: disallowing access to sites with expired or mismatched SSL certificates, warning users about data entry on sites with questionable URLs, and recognizing well-known spyware or botnet-like activity. Blue Coat does not advertise its prices or licensing terms.

WebSense

WebSense is a software solution and requires a Windows machine with considerable computational and disk resources. It performs URL categorization, port blocking, and naïve file-type identification. It offers the ability to add machines for increased scanning and storage capacity which demonstrates some scalability and redundancy. Also, it runs on COTS hardware and operating systems. The current version of WebSense does not perform signature-based scanning or binary file-type identification and cannot block connectionless traffic, like UDP, because it is an eavesdropping solution rather than an inline solution.

SurfControl Web Filter

SurfControl's software solution not only monitors and filters but also offers reporting capabilities so that a high-level diagnosis of recent usage trends and activity can be more readily identified. Additionally, it interfaces with numerous third party products: Windows, Microsoft ISA, Novell BorderManager, Blue Coat ProxySG proxy devices, CheckPoint firewalls, Citrix presentation server, Cisco CE and PIX, and Juniper Networks' firewall and VPN products. Using both host- and network-based monitoring, this distributed (not inline) solution allows for custom policy rule sets to be implemented.

Web Filter has threat databases for instant messaging (IM), peer-to-peer (P2P), spyware, and games respectively. SurfControl does not advertise prices or licensing terms, but the web site notes that a license to use one of SurfControl's products covers the use of any of its other products.

8e6 R3000

This Redhat Linux-based hardware appliance uses an eavesdropping method of content filtering similar to that of WebSense. It can also operate as a router and firewall, and claims to filter based on type of traffic protocol, source, or file type. Licenses are volume priced per user with 1, 2, or 3 year subscriptions available. 8e6 does not advertise prices.

We were unable to procure a device for testing, but a survey of the whitepapers published by 8e6 suggests that the R3000 performs basic URL categorization filtering, port blocking, and naïve file-type identification. One of its white papers claimed that “pass-by” technology (what we refer to as eavesdrop filtering) is only available as a hardware solution, but this claim is patently false. WebSense seems to perform substantially similar functions to the 8e6 products but is software rather than hardware.

Alternate Solutions

Both content filtering solutions and IPS devices watch internet traffic and attempt to block or “filter” unwanted content. However, their goals, targets, and methodologies differ in significant ways. Many content filters attempt to prevent or limit access to

information based on a predetermined categories. Thus, these content filters only address the category under which a given URL has been placed rather than actual content contained within a transmission. For this reason, traditional content filters are poorly suited for detecting and preventing malware from novel sources. Most vendors exaggerate the malware-defense capability of their products which are at best successful at stopping non-novel *sources* of malware. They can not recognize the malware itself nor can they recognize new sources of distribution unless they use a signature-based scanning technique.

IPS systems and some sophisticated content filters address this problem directly, inspecting each packet that comes across the line for malware or other malicious activity. This "deep" packet inspection requires much higher computational resources; thus, these machines are typically much more expensive than simpler content filters. The advantage of IPSs is that they can remain entirely ignorant of the source or destination of the content. Further, many devices in this class incorporate intimate knowledge of protocols and their respective weaknesses, allowing recognition of disguises such as TCP fragmentation attacks and reordering. Devices capable of TCP stream reassembly and sequence reordering are also available.

As a class, IPS devices are more intelligent and capable of effectively targeting and stifling transmission of malware and malicious code than are content filters because they are designed to identify the malware itself, rather than make a determination based on the source of the content. Thus, administrators using IPS devices instead of content filters will also spend less time carving out exceptions to their filtering policies.

Unfortunately, this sophistication comes with a significant financial cost. First, while content filters can be obtained for under $1000, IPS devices often cost tens of thousands of dollars. Second, if an IPS malfunctions, it might be much more difficult to diagnose and remedy the source of the failure due to increased complexity. Finally, administrators must place IPSs inline creating a potential for bottlenecks and introducing design problems for complicated networks with multiple connections to the internet.

In general, a sufficiently large and well-funded organization could benefit from using both categorizing content filters to quickly eliminate connections to sites that are known to be bad while using an IPS to dissect the payloads of unblocked traffic into and out of the network.