Expert Report

Web Filtering Technology Assessment

5 December 2003

Table of Contents

List of Tables

1 INTRODUCTION

This document offers opinions regarding the technologies currently available to Internet Service Providers (ISPs) that could enable them to filter web traffic destined to a specific website resource, as identified by its Uniform Resource Locator. The following technologies and techniques are capable of filtering traffic to a website:

- DNS filtering – Blocks all network access to web servers identified with a specific name, by intentionally responding incorrectly to specific Domain Name System requests.
- URL filtering – Blocks traffic by the Uniform Resource Locator and the host field located inside the packet's payload (the actual content of the packet).
- IP filtering – Filters some or all network access to specific servers, as identified by their Internet Protocol addresses. This includes purposely rerouting traffic to an alternate host (interception) and sinking traffic into a "black hole" (null-routing), as well as limiting access by using Access Control Lists (ACLs).

DNS and URL filtering are the focus of this report. IP filtering requires blocking access to either an IP address or IP address and port combination. As a direct consequence, every web page located on this website, as well as any other websites hosted on the same IP address[1], will also be filtered. This situation is frequently encountered on websites made up of user communities and web hosting sites; IP filtering would unduly affect the entire World-Wide Web experience.

It is my opinion that DNS and URL filtering are reasonably effective methods that run little risk of filtering websites other than those intended to be filtered. These two methods involve varying degrees of cost to the ISPs. Basic DNS filtering is simple and inexpensive. URL filtering can be a more complex and costly process, but may be available to some ISPs.

No technical solution will be 100% effective, and the user will always be able to evade filtering mechanisms. Nonetheless, the user's ability to avoid an obstacle, including filtering access to a website, does not wholly invalidate the filtering technique. Each deployment of either of these techniques will have greater or lesser degrees of effectiveness, and similarly, they will also vary in feasibility of implementation. Realizing both of these limitations, this document attempts to provide an explanation and support for the opinions stated.

During the writing of this document, I have drawn extensively upon my years of experience working at an ISP.[2] I have also drawn upon other existing sources, all of which are cited.

2 BACKGROUND

This section provides background information prior to presenting the basis for opinions documented in §3.

2.1 Definitions

- World-Wide Web – The complete collection of web pages available from all web servers Internet-wide, which are available via the *HyperText Transfer Protocol*. There are many more services and hosts on the Internet than just those which make up the World-Wide Web by hosting the content of websites, but many people have the impression that the World-Wide Web is the only means of making data available to the Internet community. This term is often abbreviated to "WWW."
- FQDN – An acronym which stands for "Fully Qualified Domain Name." The *FQDN* is the full name of a system, consisting of both its *hostname* and its entire *domain name*. For example, the *FQDN*

[1]This practice of hosting multiple websites on a single IP address is known as "virtual hosting."
[2]My *Curriculum Vitae* is attached.

`venera.isi.edu` is composed of a *hostname*, `venera`, and a *domain name*, `isi.edu`.[3] Most computers on the Internet have only one *FQDN* and only one *IP address*, but servers may have many *FQDNs* and many *IP addresses*.

- Hostname – An easily remembered name for a computer, which can be used instead of the *IP address* in high-level communications. *Hostnames* are considered local to a single site, are just a single word, and are prepended to the *domain name* to form an *FQDN*. The word "hostname" is often used in place of the wordier "FQDN," and throughout this document, the two terms are used more or less interchangeably. (Strictly speaking, all *FQDNs* are *hostnames*, but not all *hostnames* are *FQDNs*.)

- Domain name – A generic name, published in the *DNS*, for the owner of one or many hosts on the Internet. Examples of *domain names* include `example.com` and `state.pa.us`. When the *domain name* is appended to the *hostname*, the *FQDN* is formed.

- DNS – An acronym for the "Domain Name System." This is a protocol for translating *FQDNs* into their corresponding IP addresses. By extension, the term also refers to the collected information published with this protocol. Computers can only use the IP address to indicate the destination of traffic, although people almost always use *hostnames* or *FQDNs*.

- Name Resolution – The process of converting the *FQDN* of a host on the Internet to the associated IP address for that host, using the information in the *DNS*.

- Access Control Lists (ACLs) – A means by which access to, and denial of, services can be controlled. It is often found in *router* configurations to selectively allow certain types of traffic to pass through that *router*.

- IP Address – A numeric identifier for computers that are connected to the Internet. The *IP address* of a computer is analogous to the street address of a building.

- TCP Port – A numeric sub-identifier, analogous to the apartment number of a suite in an apartment building.

- HTTP – An acronym for the "HyperText Transfer Protocol," which is the high-level protocol used between web browsers and web servers to request web pages and to answer such requests. Web browsers indicate to web servers the latest revision that they can employ by sending as a part of the request for data the latest version that they can understand. Currently, all popular browsers use version 1.1.

- Firewall – A set of related programs, often running on a dedicated device and located at a network gateway server, that protects the resources of a private network from users from other networks. *Firewalls* generally have a more sophisticated set of access control features than simple *ACLs*. By extension, this term also refers to the security policy that is enforced.[4]

- Router – A device or, in some cases, software in a computer, that determines the next network point to which a packet should be forwarded toward its destination. *Routers* are connected to at least two networks, and decide which way to send each packet based upon their current understanding of the state of the networks to which it is connected. *Routers* are located at every meeting of networks, including at Internet Points of Presence.

- Application Switch – A device in a telecommunications network which channels incoming data from any of multiple input ports to the specific output port that will take the data toward its intended destination. On a local area network, a switch determines the destination of each incoming message (by inspecting the packet header) and thereby decides which output port to forward it to. An

[3] This example is from RFC 1392, "Internet Users' Glossary."
[4] This definition and subsequent definitions have all been derived from those found at `http://searchsecurity.techtarget.com/gDefinition/`.

application switch uses not only the packet header, but also higher-level information, such as parts of the IP information included in the packet, or some external traffic-rate metrics.

- Cache Server – A device, typically within a business enterprise, that saves, or caches, web pages, and possibly other files that users have requested, so that successive requests for these pages or files can be satisfied by the cache server. This is intended to speed up response times and reduce bandwidth utilization by eliminating the need for the same data to be downloaded multiple times. By extension, data being accessed can be limited or altered by the *cache server*.

- Proxy Server – A device which receives requests for an Internet service, such as web service, from users. If the request passes filtering requirements, the *proxy server* will usually forward the request on to the Internet and supply the result back to the original requester. A common case occurs when the *proxy server* is also a *cache server*. In these circumstances, once the request is approved, it will go through the usual *cache server* mechanisms for retrieval.

- Anonymizer – A privacy service that allows users to visit websites without allowing anyone to gather uniquely identifiable information about the sites that they visit (including obscuring this information from the visited websites). They function as public *proxy servers*, for all intents and purposes. A*nonymizers* are used for two reasons: to protect the privacy of the users, or to bypass blocking applications that would prevent access to websites that they wish to visit.

2.2 URL Overview

A URL designates a resource available from an Internet website, and can have the following form:

`http://<host>:<port>/<path>?<searchpart>`

The "http://" indicates that the resource specified should be retrieved using HTTP. <*Host*> is either a web server's FQDN or its numeric IP address. <*Port*> designates the TCP port number that the web server is using to accept requests. Both <*path*> and <*searchpart*> specify to the web server how to internally locate the desired resource. Everything other than the "http://" and the <*host*> portion are optional.

2.3 HTTP Request/Response Overview

Throughout the following discussion, the term "user" shall refer to a user of the World-Wide Web. A high level overview of an HTTP request/response exchange can be summarized as follows:

1. The user initiates a request for a web page. (For this example, we shall use the resource defined by the URL `http://www.example.com/index.html`). The user's workstation issues a DNS request to the user's preconfigured nameserver[5] to convert the hostname `www.example.com` to an IP address (which is required to send the request to the correct system).[6]

2. The request is sent to the IP address found in step 1. The following table contains a portion of the request that is sent:[7]

```
GET /index.html HTTP/1.1
Host: www.example.com
... <Additional HTTP data> ...
```

Table 1 Sample HTTP Request

[5] All computers capable of effectively using the Internet have been given the IP addresses of selected nameservers. Although not essential, this is *a priori* required for all practical uses of the Internet.
[6] This step is skipped when an IP address is used instead of a hostname, or if the user has recently retrieved web pages from this host.
[7] Notice the "Host" field. This field is used by the web server to identify the destination of the request in a virtual hosting scenario. This is explained in greater detail in latter portions of this document.

The path the request typically takes starts at the user's workstation, goes via the user's ISP to any upstream ISP(s), and finally to the web server that hosts the website.

3. The web server receives the request and sends a response, completing the transaction.

2.4 Web Hosting Overview

"Web hosting" can be defined as providing the storage, connectivity and services necessary to operate a website. A company that provides these services is a "web host." There are many web hosting services, ranging from those designed for individuals to those intended for use by corporations. This overview does not cover all types of hosting options.

Some web hosts provide their customers the ability to register and own their domain names. For example, a customer, "Customer A," could make its website accessible via the `http://www.CustomerA.dom/` URL while Customer B's website could be accessible via the `http://www.CustomerB.dom/` URL.

Web hosts can also provide their customers with control of a sub-page from their website. This type of offering is often referred to as a "user community." For example, a web host renting out control underneath the `http://www.example.com/` URL could provide Customer A with sub-pages starting with the `http://www.example.com/CustomerA/` URL.

Occasionally, web hosts will provide control over a subdomain of their own domain to customers, such as leasing control to Customer A of the URL `http://CustomerA.example.com`, or possibly `http://www.CustomerA.example.com`.

In all of these cases it is possible, and even common, for two URLs (even with different domain names) to be hosted on a single IP address, and possibly stored on the same server or servers. For example, `http://www.CustomerA.dom/` and `http://www.CustomerB.dom/` could be hosted on the same host and both `www.CustomerA.dom` and `www.CustomerB.dom` could resolve to the same IP address. This practice is known as "virtual hosting."

3 BASIS FOR OPINIONS

This section of the document briefly discusses each of the aforementioned options that ISPs may employ for filtering web traffic. This section of the document provides the basis for the opinions expressed in the introduction.

3.1 Domain Name System (DNS) Request Filtering

This section describes DNS filtering and documents its benefits and limitations. Technical details, including an example configuration, can be found in Appendix A.

3.1.1 Overview of DNS Filtering

The basic concept behind DNS filtering is to use a fake DNS entry for the hostnames (or more precisely, the FQDNs) of restricted sites. Suppose that `http://www.example.com/home.html` is a restricted URL. An ISP could place an artificial entry in its local customer-facing nameservers for `www.example.com`.[8] When its users attempt to resolve the hostname into an IP address, the resolution process will return the data that the ISP has chosen, or, if the ISP has so configured their nameservers, fail completely. If the ISP provides an alternate IP address, such as to a "notification site" (a server that

[8] I have read Christopher Bubb's deposition, in which he states that his employer, America Online Incorporated, "doesn't control domain servers in the traditional sense," implying that adding such entries might not be technically possible in AOL's existing deployment. Nonetheless, it remains my opinion that every DNS deployment which includes customer-facing nameservers for name resolution is wholly capable of supporting the addition of the entries I describe.

responds to all HTTP requests with an error page) then the content on the offending site is not accessible. Likewise, if the user cannot resolve the hostname into an IP address at all, they will not be able to surf to that site.

3.1.2 Results of DNS Filtering

Filtered websites will be totally unreachable to all customers configured to use filtering nameservers for name resolution. Since all of the filtering nameservers will be returning invalid information when asked to resolve a filtered website's hostname, the server actually hosting the filtered material will not be accessible to anyone using the filters. However, unfiltered websites will remain reachable as long as they have a different hostname from the filtered websites. Since their name has not been associated with invalid information by the filtering nameservers, the correct data will be returned to anyone who requests name resolution, and the site will therefore remain accessible.

For example, before DNS filtering of `www.badsite.dom`, let us assume that the following items are true:

- `www.goodsite.dom` resolves to 10.1.2.3
- `www.anothersite.dom` resolves to 10.5.6.7
- `www.badsite.dom` resolves to 10.1.2.3

If DNS filtering of `www.badsite.dom` is performed, the following items would be true:

- `www.goodsite.dom` still resolves to 10.1.2.3
- `www.anothersite.dom` still resolves to 10.5.6.7
- `www.badsite.dom` either does not resolve at all or resolves to a notification site.

3.1.3 Benefits of DNS Filtering

3.1.3.1 Multi-Protocol Use

This technique can be used to remove access via HTTP, FTP (File Transfer Protocol), Gopher, and any other protocol that relies on system names.

3.1.3.2 Can Block Non-Standard Ports

Websites on non-standard ports are blocked; the blocked website does not have to be on TCP port 80 (the default) for filtering to be effective. No substantial overhead is imposed by this mechanism, since the traffic cannot flow to the filtered site in the first place, and there is no need to inspect traffic packet by packet.

3.1.3.3 Unaffected by IP Address Changes

Changing a restricted website's IP address will not affect this filtering method. This method is wholly independent of IP addresses, and as a result, the blocked site can change its IP address as frequently as the website owner wants without bypassing the filters.

3.1.3.4 Low Overhead

This technique adds a minimal amount of processing to operational nameservers, and has the potential to reduce network traffic rather than increase it. In my experience, the speed and memory costs of adding zones to a nameserver are negligible until one reaches many hundreds of zones. Further, if the

nameserver responds in a manner that prevents the HTTP request from ever being issued, overall network traffic will be slightly reduced.

3.1.3.5 Scalability

Since this technique uses preexisting infrastructure, and imposes minimal operational impact, it will scale almost identically to the installed base for nameservice. In terms of maintenance, any staff qualified to maintain the existing nameservers can keep this mechanism running. It is my opinion that if an ISP already operates a large number of nameservers, they are more likely to have automated tools to simplify the administration of those servers, and adding DNS filtering to these tools should not present a serious impediment. ISPs without such tools will need to manually integrate such a solution, but I expect that such ISPs will not have a large set of nameservers, and therefore the integration should be relatively easy.

3.1.4 DNS Filtering Limitations

3.1.4.1 Not Effective for URLs Containing an IP Address

Most URLs contain the web server's DNS hostname. For example, in the URL `http://www.example.com/home.html`, `www.example.com` is the DNS hostname. However, it is possible for a URL to contain the IP address instead of the DNS hostname, such as the URL `http://10.1.2.3/home.html` (or even `http://167838211/home.html`). Specifically in this scenario, IP blocking techniques may also be a viable means of filtering without significant collateral damage, as sites that are accessed by IP address are typically not virtually hosted. In very rare circumstances, the IP of a filtered site may be known to someone wishing to access it, and the webserver may be configured to allow a specific website to be transmitted when it is contacted by IP address instead of by hostname, but these two circumstances are extremely rare, and are generally only the case when the person wishing to access the website is in fact the owner of the site.

Most commercially-hosted websites cannot be accessed by IP address, since, as mentioned in §2.4, commercial providers of websites often use a practice known as as "virtual hosting." When a web server that is providing virtual hosting receives a request for a website, it indirectly depends upon the hostname typed by the end-user to determine which website is requested. It is extremely uncommon for web servers with virtual websites to treat a request destined for an IP rather than for an FQDN as anything but an error.

3.1.4.2 Entire Web Server is Blocked

This technique does not allow for selective blocking of individual pages on a web server. Therefore, if the restricted page was `http://www.example.com/badpage.html`, this technique would block all access to `www.example.com`, not just the offending sub-pages.

By extension, this limitation also applies to some community pages. For example, it would not be possible to block just one community sub-page if that community sub-page is in the path of the URL, such as `http://www.example.com/~baduser/`. However, it is possible to selectively filter a community website if it has a unique hostname, such as in the URL `http://baduser.example.com/`.

Note that blocking an entire website is not a significant concern. Responsible community websites will remove prohibited material upon notification; thereby preventing the need for their entire site to be blocked in the first place. Other sites, which knowingly allow users to post illegal materials, are likely to be wholly filled with such content.

Overall, this issue can be at least partially mitigated with techniques described in §3.1.5.1.

3.1.4.3 Will Affect Subdomains and Additional Types of Traffic

Technically, a domain name alone, such as the `example.com` in the URL `http://example.com/`, might itself be used to access a web server (rather than using the hostname `http://www.example.com/`). At the same time, this domain name could serve as a parent domain to other entries, such as `host2.example.com`. In this case, placing a fake DNS entry within a nameserver for the parent domain (`example.com`) would successfully block the restricted site (e.g., attempts to access `http://example.com/` would fail), but it would also cause DNS resolution to fail for subdomains, such as `http://host2.example.com/`. Additionally, all other network services, such as email, to both that domain and all subdomains of that domain, will fail for filtered users. (Note that filtering a subdomain only affects that subdomain, and as mentioned in §3.1.4.2, it will affect neither the parent domain nor siblings of the filtered domain name.)

3.1.4.4 Scope is Limited to the DNS Server Users

This technique could be bypassed by users who manually change their computer's DNS settings to point to a non-filtering DNS server. It is my opinion that most residential customers will not change their nameservice configuration without having both explicit directions and deriving an obvious benefit from making the change. Additionally, large corporate customers of ISPs often manage their own nameservers, which completely bypass the ISP-controlled nameservers, and are therefore not subject to external filtering, but most corporate customers are discouraged from viewing content at work that is likely to be filtered.

Another method of evading filtering is to contact an anonymizer that is not subject to filtering, but this generally requires explicit directions or a relatively high amount of technical expertise. Nonetheless, using anonymizers will successfully bypass the nameservice filters.

3.1.4.5 Multi-Jurisdiction Nameservers

This technique would become more complex if the filtering nameserver was being used by users both inside and outside Pennsylvania. If filtering were applied to nameservers used by residents of Pennsylvania, non-Pennsylvanian users of the nameserver would have their traffic restricted identically to Pennsylvanians. While a technical solution is possible, limiting access to nameservers based upon political jurisdiction requires extensive planning, network engineering, and increased server deployments for those ISPs with any customers outside Pennsylvania.[9]

3.1.5 Optional Issue: Using a Notification Website

One question when using DNS filtering is that of which IP address to return when a restricted site is requested. There are three possibilities:

- Return an invalid, non-routable IP address, such as 10.0.0.1 or 127.0.0.1.
- Return a valid IP address that points to a government or ISP owned web server, which will be referred to as a notification website.
- Intentionally fail, and return an answer which stops further processing.

When a user attempts to surf to a restricted website, such as `http://www.example.com/`, the nameserver could return the IP address for the notification web server. The user would automatically surf to that website instead, and the notification website could serve a web page that explains why the real site is not reachable. This page would serve to educate the public about the Pennsylvania law(s) in effect. Without such a notification page, it is not clear how the public will be informed as to what is being

[9] If multiple jurisdictions mandate filtering (without combining their lists), maintaining appropriate filtering on a region-by-region basis will require additional logistical planning.

restricted. Additionally, without notification, the filtering becomes indistinguishable from a network failure, at least to the majority of most ISPs' customers.

3.1.5.1 Optional Issue: Additional Software

Should a notification server be deployed, it is possible to include the functionality of either a web cache or (preferably) a proxy server on the notification server, and indeed, other application proxies. By including such software, DNS filtering can be made granular (specific) enough to allow those portions of a website not in need of filtering to reach the end user, and further reduce the impact of DNS filtering on other protocols, such as email. Although it would require investigation and analysis to ensure that the notification server(s) deployed could handle the traffic that would be both received and generated, it is within the realm of technical possibility.

In brief, this would combine DNS filtering and URL filtering to minimize filtering impact while maximizing filtering capability. As mentioned in §3.1.5, nameservers will redirect those end-users seeking filtered content to the notification server, by supplying the notification server's IP address instead of the correct one. Each request that was thusly redirected can now be analyzed with URL filtering at a single point, the notification server. If the request is determined to be for a legitimate resource, the notification server can either serve the material from cache or proxy the request onwards. If the request is in fact determined to be in need of filtering, an error message can be displayed instead.

If this option is deployed, it is critical that the notification server not act as a critical point of failure and also that the overall system be resilient enough to continue correct operations under abnormal or unexpected load. It is also essential that the notification server not be filtered by DNS filters to prevent a loop from occurring. (If the notification server is DNS filtered, whenever it attempts to fulfill a request for legitimate data, it will be redirected to itself again rather than to the correct source for the information.)

Such additional software can be purchased commercially, created in-house, or deployed using Open Source software. All of these solutions will require support at some level, be it purchasing annual support from the vendor, training the available staff, or contacting a third party to support the installation. For more detail, see §3.2.2.

3.2 URL Filtering

This section describes URL filtering and documents its benefits and limitations. Technical details, including an example configuration, can be found in Appendix B.

3.2.1 URL Filtering Technology Overview

URL filtering monitors web (HTTP) traffic by looking at the URL and the “Host” field within the HTTP request to determine the destination of the request. The host field is specifically used by web hosting servers (when multiple sites are virtually hosted on the same server) to determine which resources to return. For example, a web server reachable via the Internet on a given IP address, hosting both `http://www.companya.dom/` and `http://www.companyb.dom/`, would be able to determine which resources to return based upon the host field provided in the requests. (There are two versions of the HTTP protocol in use, and only the newer version, HTTP/1.1, uses the host field. The older version, HTTP/1.0, is not required to include it, and may not send it, at the expense of not being able to access virtually hosted websites.)

URL filtering often falls under the broader topic of “Content Management.” URL filtering technologies come in two varieties: “pass-by” and “pass-through” filtering.

3.2.1.1 Pass-by Filtering

A pass-by filtering product (either software or bundled software and hardware) operates on network traffic without being directly (i.e., serially) in the path between the user and the Internet. The original request is

transmitted to the end-point web server, but if the request has been deemed inappropriate (that is, identified for filtering), the filtering product will prevent the original page from ever reaching the requester. This technique allows the filtering device to be uninvolved with routing the request. If the filtering device fails, network traffic will continue to flow normally. In other words, a pass-by URL filtering device will not introduce a point of failure on the network on which it resides. Also, since pass-by filtering is largely uninvolved with routing, it is relatively non-intrusive and is a minimal drain on network resources.[10] This technique is similar to that of commonly used Intrusion Detection Systems that inspect (or "sniff") network traffic without being serially part of the communication path.

3.2.1.2 Pass-through Filtering

Pass-through filtering involves using a device that is situated directly in the path of all user requests. As traffic passes through the filtering product, it is filtered. Examples of pass-though filtering devices may include some models of firewalls, routers, application switches, proxy servers, and cache servers. Alternatively, some devices situated serially in a data stream can "hand off" traffic to a third party filtering product for inspection, which will determine whether or not to filter the traffic, and return it to the normal data flow if it is determined to be acceptable.

3.2.2 URL Filtering Options

A number of different products and product types are capable of performing URL filtering. Some of the products are specifically designed with the sole purpose of performing Content Management, of which filtering URLs is a component. Such software often comes bundled with a service that provides a listing of websites that have been determined by the manufacturer to be either inappropriate for many environments, or in some way disruptive. Licensing for these products is often on a per-user basis, and includes in the cost some sort of subscription to the manufacturer's "Bad Site List." Some of the products that fall into this specially designed category include:

Product	Company
Sentian	N2H2
IM Web Inspector	Zixcorp
Smartfilter	Secure Computing
Web Filter	SurfControl
Web Security	Symantec
EIM	WebSense
iPrism	StBernard Software
ProxySG	Blue Coat
PureSight	iCognito Technologies Ltd.
bt-WebFilter	Burst Technology
Intelligent Content Management	FilterLogix
R3000	8e6 Technologies
OrangeBox Web	Cobion AG
DynaComm i:filter	FutureSoft, Inc.
CyBlock Web Filter	Wavecrest Computing

Table 2 URL Filtering Products

(The presentation of these products is intended to demonstrate the widespread availability of Content Management Systems, but does not represent an endorsement of these products, nor does it imply that these products are necessarily suitable for any given filtering environment.)

[10] To function correctly, pass-by filters must be connected to a network port that duplicates all traffic flowing through that segment of the network. This connection does impose more load upon the infrastructure.

To the best of my knowledge, most, if not all, of these products allow user-specified URLs to be added or removed from the "Bad Site List," although the original websites on the list may not always be removable.

Other products capable of performing URL filtering are available, often as an additional feature embedded within a product, above and beyond the primary function(s). As the demand for Content Management continues to increase, both hardware and software vendors have been adding pass-through URL filtering functionality into devices that may already be on many networks, including those at the ISP level. As touched upon above, URL filtering may be supported on currently deployed routers, firewalls, application switches, proxy servers, and caching servers. Note that not all brands and models of these devices support URL filtering, but many do. Another important fact to keep in mind is that different brands may implement their solutions differently, and with varying levels of granularity; therefore, when considering a URL filtering solution, a stricter requirements-based approach defined by the needs of the specific installation base should be undertaken as part of the evaluation and decision-making process.

In addition to the hardware products listed above, software solutions, including Open Source solutions, are available. One such program is "Squid," software capable of acting as either a proxy server or a cache server or both. Squid includes URL-filtering functionality as a part of its operation as a proxy server. Additionally, when installed upon a pass-through device on the network, Squid can be configured to receive and arbitrate requests as a transparent proxy.[11] FAQs and other how-to documentation can easily be found on the Internet which will provide step-by-step instructions on setting up Squid to filter URLs.

3.2.3 URL Filtering Benefits

3.2.3.1 Equipment May Already be Installed

Depending on the equipment installed, this technique might not require additional hardware. An ISP may already have appropriate hardware capable of URL filtering. For example, many models of Cisco Systems routers (a common brand in most ISPs' infrastructure) may be able to run software that supports filtering URLs.[12]

3.2.3.2 Virtual Websites are Unaffected

This technique does not affect virtually hosted websites that share the same IP address as the restricted website. A blocked website and a non-blocked website can share the same IP address. Since the HTTP/1.1 host field is read to determine the ultimate destination of the request, non-blocked sites will remain reachable by users.

3.2.3.3 Unaffected by IP Address Changes

In most cases, changing the restricted website's IP address will not affect this method. Since filtering is not related to IP address, the owners of a blocked site can change the IP address as much as they want, but the site will still be unreachable to users behind the filters. (This technique of frequently changing IPs to avoid filtering is in common use in the field.)

3.2.3.4 Specific Pages May be Blocked

In general, this technique does allow the selective blocking of individual pages on a web server. However, this feature is dependent on the selected filtering product's capabilities, and not all products listed support this level of granularity. Therefore, in some cases, it may be possible to restrict the URL

[11] Transparent proxies operate differently than normal proxies, in that the users are generally unaware of their existence. Also, with transparent proxies, no configuration of the user's browser is required.
[12] Of course, before altering the configuration of a critical piece of infrastructure such as a router, care must be taken not to overload the device and cause severe network issues. More detail is given in §3.2.4.5.

`http://www.example.com/badpages/` without blocking all access to the site `http://www.example.com/` or to any other sites underneath that URL. This is most applicable to filtering content present upon some web servers that host community pages. However, if the owner of the filtered content can find or create a different sub-page, the filters must be updated to include the new location as well. (This technique of frequently moving prohibited content around on one or many servers is also in common use in the field.)

3.2.3.5 Effective for URLs Containing an IP Address

Most URLs contain the web server's DNS hostname. For example, in the URL `http://www.example.com/home.html`, `www.example.com` is the DNS hostname. However, it is possible for a URL to contain the IP address instead of the DNS hostname, such as the URL `http://10.1.2.3/home.html`. URL filtering can still restrict access to those websites. An HTTP/1.1 conversation will place "10.1.2.3" (or alternatively, "167838211") into the Host field of the request. This entry can be filtered upon as if it were a traditional hostname. Again, an HTTP/1.0 request may try to directly access the machine with the IP address 10.1.2.3, and therefore such requests will be subject to normal HTTP/1.0 filtering rather than the more flexible HTTP/1.1 filtering. Specifically in this scenario, IP blocking techniques may also be a viable means of filtering without excessive collateral damage, as sites that are accessed by IP address are typically not virtually hosted.

3.2.3.6 Not Limited to the Users of the DNS Servers[13]

Unlike DNS filtering, this technique cannot be bypassed by users that manually change their computer's DNS settings to point to a non-filtering DNS server. This method will work as long as their connection to the Internet is visible to the filtering system.

3.2.3.7 Multi-Protocol Use

Content Management products can be used to filter access via HTTP, FTP (File Transfer Protocol), Gopher, or any other protocol. The ability to support non-HTTP protocols is product and configuration dependent.

3.2.4 URL Filtering Limitations

3.2.4.1 Usually Cannot Block Non-Standard Ports

A web server does not have to provide data using the default TCP port number. Websites on non-standard ports are generally difficult to block, since they require an extra level of attention to filter properly. A URL filtering solution could be technically capable of filtering HTTP connections on non-standard ports, but between the extra administration required and the possible overhead of filtering a much larger volume of data, it is often impractical.

3.2.4.2 Does not Work with Encrypted Traffic

Because HTTPS requests using SSL/TLS[14] are encrypted, URL filtering software cannot read the Host field. Therefore, filters cannot effectively determine which resource on an IP address the request is actually intended for. However, the vast majority of encrypted web servers have a configuration limitation of a single host per IP address, and therefore the filters can often (although not always) fall back to some form of IP-based blocking.

---

[13] See §3.1.4.4 for the relevance of this analysis.

[14] SSL/TLS is a means of encrypting HTTP traffic. The majority of public websites do not use encryption to transmit their material, but the method exists and is well codified in RFCs 2616, 2817, and 2818. Encrypting all traffic is somewhat expensive and impractical for commercial sites, since it requires a significant amount of processing by the web server.

3.2.4.3 Potentially Very Expensive

Some vendor products designed specifically for Content Management and URL filtering may be prohibitively expensive, especially to ISPs, because many vendors tie number of filtered users directly into their pricing models. Since ISPs have a large number of users, the costs for the software and any mandatory recurring licensing become high. Depending on implementation specifics and products under analysis, this problem might not be applicable. For example, if existing infrastructure can be used in conjunction with an externally provided "Bad URL List," while consuming minimal operational overhead, then deploying a URL filtering solution will have little to no additional financial cost.

3.2.4.4 Jurisdiction Across the Network

Regardless of the technique used, it may be quite difficult to isolate a certain region, such as Pennsylvanian users. This limitation depends solely upon on the ISPs network infrastructure and the components available to them.

3.2.4.5 High Overhead

Depending upon the product chosen, these techniques may require substantial processing on their host. If they are running in pass-through mode, this will also translate to increased network latency and therefore slower traffic overall. If they are running in pass-by mode, their connection to the network will need to duplicate all traffic passing through that segment of the network, which has the potential to slow it down. Finally, dedicated servers might need to be provided for the filtering, which therefore cannot perform other tasks. If the filtering can be done on a device that is performing other duties at the same time, filtering might impose a slowdown upon the completion of those duties as well.

Depending on traffic volume and distribution, this technique has a potential of overwhelming the URL filtering product, especially during abnormal traffic situations. In the case of pass-through filtering, this degradation could result in anything from slower response times for users to complete failure of a network link. In the case of pass-by filtering, this could result in a failure to filter, or other transient network problems which would cause degradation that would be difficult to prevent. It may be possible to minimize the degradation in either case to acceptable levels by carefully studying, choosing, and planning the filtering options prior to implementation.

3.2.4.6 Scalability

In general, the more customers an ISP has, the more difficult it will be to perform URL filtering, unless the ISP has already deployed web caches or filters independently. Very small ISPs may be affected the least.

3.2.4.7 Planning Expense

Do to the possible performance degradation and large number of available alternatives, an ISP would have to study, plan and carefully monitor the performance impact of any URL filtering implementation.

3.3 Other Potential Filtering Techniques

3.3.1 IP Filtering

IP filtering techniques directly block traffic destined for the IP address of a website. This technique may be practical in cases where the entire website is commonly accessed via IP address only or is accessible by IP address rather than name, such as with a URL like `http://10.0.0.1/`. In most cases, however, this filtering technique is not recommended due to three primary deficiencies that cannot be overcome:

- Blocking access to an IP address will also block traffic to other sites virtually hosted on the same IP address, regardless of whether or not they are related to each other. This will negatively impact a significant percentage of websites on the Internet.[15]

- Blocking access to an IP address will block traffic to every member of a community hosted by that IP address. It will block an entire website, not just a portion or set of sub-pages. This is similar to the consequences described in §3.1.4.2, but is even more destructive, since all IP traffic will be blocked, not just all name-based traffic.

- It is a frequent practice to change the IP address of filtered websites as soon as the owner of the website discovers the filtering. This practice relies upon DNS to allow visitors to still reach the site, and renders IP-based filters useless.

4 CONCLUSIONS

All techniques will have limited effectiveness, with both those who intentionally provide prohibited content and those users who will deliberately try to access such content. Technically proficient members of both categories will be able to bypass obstacles with varying degrees of difficulty. Ideally, ISPs will attempt a reasonable best effort to make this evasion as challenging as possible.

The technique(s) used should be scalable, cost-effective, and impose minimal performance delays. ISPs will vary by the number of users, connection speeds, traffic volumes, architecture, products, and technologies employed. Therefore, it is highly unlikely that one solution will work for all ISPs, regardless of size and customer base. For example, URL filtering using the Cisco IOS "Firewall" feature set may be practical for one ISP but not for another. Therefore, several approaches have been discussed in this document. The following table compares some of the considerations for implementing the two solutions that were the focus of this document:

Filtering Technique	Implementation Difficulty	Financial Cost	Performance Impact
DNS based	Low	Low to Medium	Low
URL based, purchasing content management solution	High	High	Pass-by: Low Pass-through: High
URL based, on a preinstalled or inline network component	Medium	Medium	Medium to High*
IP based, using ACLs alone	Low	Low	Low (on edge devices)
IP Based, using Policy Maps (see Appendix B)	High	Low	Low (on edge devices)
*Performance impact in this case will depend upon traffic volume, hardware capacity, etc.			

Table 3 Web Traffic Filtering Solution Comparison

[15] Although it is indeed common for unrelated websites to share an IP address, Michael Clark's report has flaws in its methodology. I believe that Mr. Clark's report is probably accurate in its figures indicating the number of different domain names referencing shared IP addresses, but it fails to account for the cases when the websites are related. For example, at the time of writing, the websites available from `www.disney.net`, `www.disneyworld.com`, and `www.disney.org` all share identical IP addresses, but additionally, they are related to each other (and in fact, all provide content from a website at `disney.go.com`). There are many other cases where this is true, such as when a registrar holds multiple domain names for sale on a "parked" website with static content. Mr. Clarke's report references work by "Benjamin Edelman of the Berkman Center for Internet & Society of Harvard Law School," which acknowledges these limitations, but neither report actually accounts for these cases in their concluding statistics.

4.1 Summary of Proposed Solutions

DNS filtering is not very difficult for technically adept end users to avoid, but provides a significant impediment to accessing filtered material in most cases. In general, as DNS filtering is adopted on a greater scale, it becomes more effective at filtering content. Although it requires planning and effort to maximize results, even minimal results are effective, and impose almost no additional burden upon existing infrastructure. With careful design and appropriate integration with other filtering techniques, DNS filtering can provide an inexpensive yet effective element in overall content filtering. It is my expectation that DNS filtering will properly filter the vast majority of requests for content that requires blocking. Of course, each ISP should examine the effectiveness of its deployment to ensure that it is performing properly and is worth the added complexity.

URL filtering is difficult to evade, assuming that the network of the ISP using it is capable of supporting an effective deployment. If this is the case, all HTTP traffic will be inspected as it travels from the user's workstation to the destination website, making the filtering extremely effective. However, depending on specific implementations, it may be bypassed by using HTTP over SSL/TLS, either for encrypted communication directly with the website, or for traffic through anonymizers[16]. If encryption is used, it is not currently feasible for URL filtering software to decode the request and determine the ultimate destination at any finer detail than that of the IP address of the recipient. This is often insufficient data to allow responsible filtering. All existing filtering software has the ability to decode cleartext HTTP requests, but not all software on the market can decode requests routed through an anonymizer, as opposed to those sent directly to a web server. As a result, care must be exerted to ensure that the URL filtering solution that is chosen can correctly filter the types of traffic that will be seen upon the network. Finally, connections to web servers utilizing non-standard HTTP ports may be able to bypass the filters.

Again, it is my opinion that DNS and URL filtering are both reasonably effective methods that both run little risk of filtering any websites other than those intended to be filtered. These two methods involve varying degrees of cost to the ISPs. Basic DNS filtering is simple and inexpensive. URL filtering can be a more complex and costly process, but may be available to some ISPs. Using either or both of these methods in combination with other filtering solutions may lead to even stronger filtering capabilities, but this depends upon the specific nature of each ISP's network topology. Neither of these solutions in any way prevents ISPs from using wholly different techniques or combinations of techniques, given sufficient analysis to ensure minimal undesired interaction between the various filtering platforms in use. Finally, new solutions for performing "Content Management" are continuously being developed. Future solutions may become less expensive, faster, and possibly a more practical solution for ISPs.

Benjamin A. Stern

[16] Anonymizers are basically proxy servers available to the public. When using an anonymizer, both the DNS requests for the destination website and the actual web page requests are performed by the anonymizer, not the user's workstation. Generally, encryption is not used when communicating with anonymizers, but using it is not impossible.

5 APPENDIX A: TECHNICAL DETAILS OF DNS FILTERING

This section explains one of the methods available for performing DNS filtering using BIND (the Berkeley Internet Nameservice Daemon), which is the software most commonly used by ISPs for nameservice.

5.1 Configuration and Zone Files

Suppose that the restricted URL is `http://www.example.com/naughty.html`, and the ISP is using BIND on their nameservers.

An entry similar to the one below would be added to the configuration file:

```
zone "www.example.com" {
    type master;
    file "restricted.db";
};
```

Table 4 Sample "named.conf" File

A corresponding zone file called “restricted.db” could contain the following data:

```
$TTL 81600
@ IN SOA ns.state.pa.us. hostmaster.state.pa.us. (
    2003110900 ; serial
    10800      ; refresh
    3600       ; retry
    604800     ; expire
    81600)     ;
  IN A   <fake IP address>
```

Table 5 Sample "restricted.db" File

(To add more restricted sites, additional entries would be added to the configuration file.) Upon issuing the command “ndc[17] reconfig”, BIND will rescan the configuration file and start to answer with the supplied data for the filtered zones. The exact same zone file could potentially be reused for all blocked websites, reducing the number of independent files required to maintain DNS filtering.

Note that the fake IP address in the zone file could be a non-routable address, such as 10.0.0.1, or preferably a notification web server's IP address[18]. If a notification server cannot be used, invalid data can be intentionally included in the zone file to force the nameserver to reliably fail when resolving filtered domains.

5.2 Advantages of This Configuration

It is not required to perform a cold restart of the server when adding or removing filtered sites. Performing an “ndc reconfig” is anticipated to take approximately one second per one thousand domains on a properly configured nameserver. In other words, making changes will not substantially impact the operation of properly maintained nameservers.

Only one zone file is strictly required, although certainly more could be used. The sample zone file listed above can be referenced by all restricted sites entered into the nameserver’s configuration file. This greatly facilitates maintenance of this method over time, and makes changes such as altering the notification server's IP address simpler.

[17] ndc is a program that interacts with BIND 8 to control the daemon's operation. Under BIND 9, the program is called “rndc”. Most other nameservice programs have similar utilities.
[18] See §3.1.5 for details on “notification servers.”

Normal DNS maintenance consists of periodically updating the server's configuration or individual zones, and more commonly consists of both. Therefore, deploying this technique will immediately integrate into existing ISP procedures. Large ISPs that are maintaining hundreds or thousands of domains are likely to use proprietary tools for nameservice data distribution. These tools can be leveraged to push out this sort of information as well. ISPs capable of effectively maintaining their nameservers without such tools are unlikely to be unduly inconvenienced by including filtering data in routine maintenance.

6 APPENDIX B: TECHNICAL DETAILS OF URL FILTERING

This section explains one of the methods available for performing URL filtering with Cisco Systems routers, which are a popular brand of router in the ISP industry.

Sometimes, it may be possible to perform pass-through URL filtering on existing infrastructure, such as on routers. For example, if Cisco routers are deployed, and they are capable of running a recent version of the IOS software which supports the "Firewall" feature set, either of the following configurations could be used:[19]

```
! If using an external URL filtering product:
ip urlfilter server vendor websense 10.9.8.7

! When using Cisco IOS internal URL filtering:
ip inspect name child-porn-filter http urlfilter
ip urlfilter exclusive-domain deny www.badsite1.dom
ip urlfilter exclusive-domain deny www.badsite2.dom
interface FastEthernet0/0
ip inspect child-porn-filter in
```

Table 6 Sample Cisco "urlfilter" Configurations

Both of the above configurations only supports filtering based upon the domain name, such as `www.badsite1.dom`. Filtering sub-pages using the "urlfilter" facility on Cisco Systems routers is not possible. However, another facility on Cisco routers can be used to filter sub-pages on websites with known IP addresses:

```
class-map match-any bad-urls
match protocol http url "/badsubpage"
policy-map mark-bad-urls
class bad-urls
set ip dscp 1
! 10.3.4.5 is the IP of the site with the bad sub-page
access-list 105 deny ip any host 10.3.4.5 dscp 1
interface Ethernet0/0
service-policy input mark-bad-urls
ip access-group 105 out
```

Table 7 Sample Cisco "Map"-Based Configuration

The above example will block any HTTP traffic from customers terminated off of Ethernet0/0 that is both destined for 10.3.4.5 (an example IP address) that also contains "/badsubpage" as the path in the URL. Note that to effectively block sub-pages with this technique, multiple ACLs and policy maps will be required.

Both of these examples are intended to demonstrate how existing infrastructure might be used to provide filtering capabilities. However, it is entirely possible that not all ISPs will be able to effectively adapt their existing equipment to provide filtering in addition to its normal duties, or that their existing equipment will not have suitable filtering abilities.

[19] Not all Cisco products can support versions of IOS that include the "Firewall" feature set while still operating within existing constraints that the ISP may have. Additionally, the "urlfilter" component of the "Firewall" feature set is only available in recent IOS images. For details on "urlfilter," visit `http://www.cisco.com/`.