

Nitke v. Ashcroft  
Geolocation of Web Users

Ben Laurie  
([ben@algroup.co.uk](mailto:ben@algroup.co.uk))

June 4, 2005

UNITED STATES DISTRICT COURT  
SOUTHERN DISTRICT OF NEW YORK

-----X

BARBARA NITKE, THE NATIONAL  
COALITION FOR SEXUAL FREEDOM, and  
THE NATIONAL COALITION FOR  
SEXUAL FREEDOM FOUNDATION,  
Plaintiffs,

DECLARATION OF  
BENNET LAURIE  
IN LIEU OF DIRECT  
ORAL TESTIMONY

-against-

01 Civ. 11476 (RMB)

JOHN ASHCROFT,  
ATTORNEY GENERAL OF THE  
UNITED STATES OF AMERICA, and  
THE UNITED STATES OF AMERICA;

Defendants.

-----X

BENNET LAURIE, pursuant to 17 U.S.C. 1746, certifies that the following statements are true and correct and understands that these statements are made under penalty of perjury:

1. I am a Director of The Bunker Secure Hosting, a company providing Internet consultancy, specifically in the area of privacy and security, and also providing hosting for Internet servers. I am a founding director of the Apache Software Foundation, a not-for-profit corporation registered in Delaware, which is responsible for the world's most widely used webserver, Apache. I am also one of the developers of Apache. I am one of the founders and developers of OpenSSL, a widely used free cryptographic toolkit. I am the author of numerous publications concerning Internet applications, computer technology and other scientific matters, which are set forth in an appendix hereto.
2. I was qualified as an expert on the subject of the feasibility of accurate geolocation (that is, the tracing of an internet user's physical location) by the the Tribune de Grande Instance de Paris in Licra et autres c/Yahoo France & Yahoo!, Inc, in March, 2000.
3. I make this declaration to provide testimony addressing the feasibility of determining the physical location of an Internet user to the degree

of accuracy required to successfully limit access to content on a geographical basis down to the town, county or neighborhood level. Also, I address the economics of implementing such systems, and of implementing tailored content such that users from each geographical location see whatever is appropriate to them and nothing else.

## 1 Overview

This report seeks to address the feasibility of determining the physical location of an Internet user to the degree of accuracy required to successfully implement the requirements of CDA.

Also, I report on the economics of implementing such systems, and of implementing tailored content such that users from each geographical location see whatever is appropriate to them and nothing else.

## 2 How The Internet Works

In order to understand the feasibility of locating users it's very important to understand to some degree the inner workings of the Internet. Note that this explanation is simplified, but does not conceal any important aspects for the purposes of the report. For full details of the relevant technical workings of the Internet, I would recommend "TCP/IP Illustrated, Volume 1" by W. Richard Stevens.

First of all, every machine directly connected to the Internet, server (i.e. websites) or client (i.e. users at home or elsewhere surfing the web), is identified by a number. No two machines have the same number at the same time, though it is possible for any particular number to be allocated to different machines at different times. This number is known as the "IP address", and is usually written as four numbers separated by full stops, each number being in the range 0 to 255. For example, 1.2.3.4 is an IP address, as is 213.129.65.100.

Generally speaking, when one machine wishes to talk to another (for sending email, or fetching a webpage, for example), it makes a "connection" between the two machines using a protocol called TCP (Transmission Control Protocol). This connection is defined by just four pieces of information - the client port number and IP address and the server port number and IP address.

As previously discussed, the IP addresses are unique to each machine. The port numbers are numbers in the range 1 to 65,535, and serve slightly different purposes at each end of the connection. At the server end, by convention, each port corresponds to a particular service - for example, email is normally transmitted using a protocol called SMTP (Simple Mail Transfer Protocol) and that is normally connected to port 25. Webpages use HTTP (HyperText Transfer Protocol) and that is usually found on port 80. Note that this is merely a convention; there is no reason not to run services on any port you feel like - however, if a non-standard port is used, there does have to be some mechanism to communicate that fact to the client (who establishes the connection to the server, and therefore needs to know the port number in advance).

The client port number is less interesting - it is merely used to distinguish between connections originating from the same client, so that if the client makes two connections to the same port on the same server (to fetch two different webpages at the same time, for example), they can be distinguished.

These four numbers, the IP addresses and ports of client and server are the totality of the information available to the server with respect to any particular connection (apart from any information transmitted over the connection itself, of course). And it is from these numbers that any information about the geographical location of the client must be derived. In fact, the only piece of information of any practical use in this regard is, of course, the client IP address.

Incidentally, note that both SMTP and HTTP (the protocol of interest in this case) are both layered on top of TCP - that is, they use TCP to transfer information between the two machines.

### 3 IP Routing

IP addresses identify the machine on the network but that is not all that is required to get data from one machine to another on the Internet.

Each machine is connected, somehow, to a router<sup>1</sup>. Each machine's local router is connected to other routers, and so on, forming an enormous and complex mesh of routers that ultimately covers the entire planet. Each router communicates with other routers to inform it which IP addresses it can reach, and how far away (in terms of "hops") they are; this information is known as a "routing table".

---

<sup>1</sup>A router is a piece of equipment whose job is simply to route data.

When machine A wishes to communicate with machine B it sends packets containing both A and B's addresses to its nearest router. The router then looks at its internal routing table and works out where the next "hop" for the packet should be. It sends it there, and that router then repeats the process. Eventually the packet reaches the router B is directly connected to, and the packet is delivered.

Of course, the routing table is constantly changing – machines connect and disconnect, ISPs add and remove connections between them, faults occur on telephone and leased lines, and so on. This, and the fact that every router must know how to reach every machine currently connected to the Internet given only its IP address is a major constraint on the way IP addresses are allocated.

Since there are over 4 thousand million IP addresses available, it is clearly not possible to have each individual IP address in the routing table of every router on the planet. So, IP addresses are grouped together into chunks that are all connected to the same router. This means that ISPs are allocated blocks of IP addresses, then they then allocate individual addresses to machines they connect to the Internet. Often these are allocated "dynamically" – that is, the IP address of a machine using dial-up is allocated when it connects, and then re-used later when it has disconnected. An ISP also will reallocate blocks according to its own internal routing and usage requirements, so even blocks of IP addresses do not tend to stay with any particular router over time.

## 4 Firewalls, Proxies, NAT and IP Shortages

Although there are, in theory, over 4 thousand million IP addresses available, because of the requirement that they are used in blocks, and other technical constraints, there is actually a shortage of IP addresses. It has, therefore, become difficult for many users to get as many IP addresses as they need.

This has led to the widespread use of a technique known as NAT (Network Address Translation). A client on a NATted network connects to a NAT gateway which has a "real" IP address which is known to the Internet, whilst the client has a "throwaway" IP address which is **not** known to the Internet (by "known" I mean "in the routing tables of the routers"). The NAT gateway takes the packets from from the client and changes the IP address of the client to its own IP address, possibly modifying the client's port number, too (because the port may have already been used by some other client) – the

gateway remembers this translation, and when packets come back from the server addressed to the translated IP address and port, the gateway applies the reverse translation and sends the packet on to the client.

As a result, from the client's point of view the connection looks just as if it directly connected to the server, but from the server's point of view it appears that the NAT gateway is connected instead of the client.

Because NAT gateways have this "one-way mirror" effect, they are also widely used as part of a corporate firewall. In fact, it is very rare for computers in corporations to ever connect directly to the Internet, both for security and IP allocation reasons.

In addition, many ISPs have noted that a large amount of the traffic that they have to pay other ISPs to carry is in the form of duplicate requests for identical webpages from their clients. So, they operate devices known as caching proxies. A caching proxy intercepts the Web requests and checks to see whether it has recently seen the same request, and, if so, returns its own copy of the page. If it has not, then it makes the Web request on behalf of the client, and caches the response for future use. Again, this has the effect that from the server's point of view it looks as if the proxy has connected rather than the client.

## 5 Information Available in HTTP

Before I move on to discuss what can be done using IP addresses, I should briefly address the information transmitted from the client to the server in HTTP. Here is what Microsoft Internet Explorer sends when asked to fetch `http://www.somewhere.com/xx.html`.

```
GET /xx.html HTTP/1.1
Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg,
application/vnd.ms-powerpoint, application/vnd.ms-excel,
application/msword, */*
Accept-Language: en-gb
Accept-Encoding: gzip, deflate
User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.0)
Host: www.somewhere.com
Connection: Keep-Alive
```

As you can see, this tells us nothing of interest about the client.

## 6 Location Using the IP Address

Since all we can know about the client is their IP address, that's all we can use to figure out where they are. There are two scenarios to consider, the first being where we use existing resources, and the second being where we require changes to the infrastructure of the Internet to facilitate geolocation.

### 6.1 Location Using Existing Resources

There are essentially two things we can find out about an IP address. The first is the registered owner of the address – for most people that will be their ISP. In some cases, that will allow us to locate them to the state level, as many small ISPs operate in single states. However, in the vast majority of cases, their ISPs will be companies like AOL or Earthlink, who operate global networks. Knowing the IP belongs to one of those companies doesn't even necessarily tell us which country they are in, let alone which state or city.

The second thing is that it is possible to trace the route to the client's IP address – of course, this is just a list of other IP addresses, but often they give clues about where they are, particularly if we have some knowledge of the various ISPs infrastructure. But again, this is a very coarse measure. The best we could expect from this is to locate people to within some distance of the nearest major city, and often not even the state would be available.

For example, if we look at the route from a machine in the US to Ms. Nitke's webserver, using standard tools, we see:

```
$ traceroute -n www.barbaranitke.com2
traceroute to www.barbaranitke.com (69.57.136.135), 30 hops max,
40 byte packets
 1 12.21.210.233 0.420 ms 0.356 ms 0.342 ms
 2 12.17.141.1 17.928 ms 14.258 ms 14.134 ms
 3 12.21.213.53 15.296 ms 14.327 ms 14.119 ms
```

---

<sup>2</sup>Running the traceroute with the “n” option set (as designated by the “-n” in the traceroute command) does not provide the names each ISP assigns its routers. Some of these designations may be keyed to geographically revealing information, such as airport codes where the router is located. However, not all such designations reveal information, and no authoritative key to designations exists. Moreover, such information can be misleading, and in any event requires a human being to examine the traceroute to endeavor to interpret the designations in light of the other data, which adds dramatically to the time and expense as projected here, which presupposes automation. Also, the process of obtaining the name from the IP address is time consuming, which would delay the client.

```

4 12.21.213.65 17.840 ms 17.511 ms 16.010 ms
5 12.124.174.57 17.147 ms 18.106 ms 17.960 ms
6 12.123.203.2 72.111 ms 52.330 ms 51.125 ms
7 12.123.44.57 51.468 ms 48.579 ms 48.736 ms
8 12.122.12.157 51.670 ms 50.580 ms 56.207 ms
9 12.123.44.129 54.372 ms 44.772 ms 48.522 ms
10 192.205.32.234 62.213 ms 50.504 ms 44.543 ms
11 208.185.175.178 51.505 ms 50.480 ms 50.361 ms
12 216.200.127.117 63.701 ms 63.784 ms 95.425 ms
13 64.125.30.2 68.911 ms 71.399 ms 67.517 ms
14 208.184.232.81 110.555 ms 108.608 ms 146.910 ms
15 216.200.127.213 104.789 ms 108.790 ms 103.712 ms
16 64.125.31.37 132.113 ms 111.415 ms 109.402 ms
17 216.200.251.29 109.661 ms 110.794 ms 111.941 ms
18 207.218.245.45 111.812 ms 111.352 ms 108.935 ms
19 69.57.136.135 111.879 ms 134.056 ms 110.702 ms

```

In this output, the first column is the hop number, the second the IP address of the computer or router at that location, and the next three the time taken in milliseconds for three packets to get to that hop and back.

Note that from Ms. Nitke's webserver, this route would be in the opposite order, starting at 69.57.136.135 (the webserver) and ending at 12.21.210.233. What can we discover about the user's location? Again, using standard tools, we can ask to whom the address 12.21.210.233 is registered:

```

$ whois 12.21.210.234
AT&T WorldNet Services ATT (NET-12-0-0-0-1)
12.0.0.0 - 12.255.255.255
MTA Solutions MTAONLINE-208 (NET-12-21-208-0-1)
12.21.208.0 - 12.21.215.255

# ARIN WHOIS database, last updated 2003-10-16 19:15
# Enter ? for additional hints on searching ARIN's WHOIS database.
$ whois -a MTAONLINE-208

```

```

OrgName: MTA Solutions
OrgID: MTAS
Address: 619 East Shipcreek Avenue, Suite 241
City: Anchorage
StateProv: AK
PostalCode: 99501
Country: US

```

NetRange: 12.21.208.0 - 12.21.215.255  
CIDR: 12.21.208.0/21  
NetName: MTAONLINE-208  
NetHandle: NET-12-21-208-0-1  
Parent: NET-12-0-0-0-1  
NetType: Reallocated  
Comment:  
RegDate: 2000-02-04  
Updated: 2001-06-01

TechHandle: JM1702-ARIN  
TechName: Solutions, MTA  
TechPhone: +1-907-793-4100  
TechEmail: dnstech@mtasolutions.com

# ARIN WHOIS database, last updated 2003-10-16 19:15  
# Enter ? for additional hints on searching ARIN's WHOIS database.

Unfortunately, this is the address of the ISP, not the machine itself, which is in Wasilla, AK, about forty miles away on the other side of a lake, and a rather more rural community than Anchorage. The owner of the machine is not MTA Solutions, but rather a resident of Wasilla.

The next hop out yields nearly exactly the same information:

```
$ whois 12.17.141.1
AT&T WorldNet Services ATT (NET-12-0-0-0-1)
12.0.0.0 - 12.255.255.255
MTA Solutions MTAONLINE-140 (NET-12-17-140-0-1)
12.17.140.0 - 12.17.143.255
```

# ARIN WHOIS database, last updated 2003-10-16 19:15  
# Enter ? for additional hints on searching ARIN's WHOIS database.  
\$ whois -a MTAONLINE-140

OrgName: MTA Solutions  
OrgID: MTAS  
Address: 619 East Shipcreek Avenue, Suite 241  
City: Anchorage  
StateProv: AK  
PostalCode: 99501  
Country: US

NetRange: 12.17.140.0 - 12.17.143.255  
CIDR: 12.17.140.0/22  
NetName: MTAONLINE-140  
NetHandle: NET-12-17-140-0-1  
Parent: NET-12-0-0-0-1  
NetType: Reallocated  
Comment:  
RegDate: 1999-02-01  
Updated: 2001-06-01

TechHandle: JM1702-ARIN  
TechName: Solutions, MTA  
TechPhone: +1-907-793-4100  
TechEmail: dnstech@mtasolutions.com

# ARIN WHOIS database, last updated 2003-10-16 19:15  
# Enter ? for additional hints on searching ARIN's WHOIS database.

This stays the same until we get to hop 5:

\$ whois 12.124.174.57

OrgName: AT&T WorldNet Services  
OrgID: ATTW  
Address: 400 Interpace Parkway  
City: Parsippany  
StateProv: NJ  
PostalCode: 07054  
Country: US

NetRange: 12.0.0.0 - 12.255.255.255  
CIDR: 12.0.0.0/8  
NetName: ATT  
NetHandle: NET-12-0-0-0-1  
Parent:  
NetType: Direct Allocation  
NameServer: DBRU.BR.NS.ELS-GMS.ATT.NET  
NameServer: DMTU.MT.NS.ELS-GMS.ATT.NET  
NameServer: CBRU.BR.NS.ELS-GMS.ATT.NET  
NameServer: CMTU.MT.NS.ELS-GMS.ATT.NET  
Comment: For abuse issues contact abuse@att.net

RegDate: 1983-08-23  
Updated: 2002-08-23

TechHandle: DK71-ARIN  
TechName: Kostick, Deirdre  
TechPhone: +1-919-319-8249  
TechEmail: help@ip.att.net

OrgAbuseHandle: ATTAB-ARIN  
OrgAbuseName: ATT Abuse  
OrgAbusePhone: +1-919-319-8130  
OrgAbuseEmail: abuse@att.net

OrgTechHandle: ICC-ARIN  
OrgTechName: IP Customer Care  
OrgTechPhone: +1-888-613-6330  
OrgTechEmail: qhoang@att.com

OrgTechHandle: IPSWI-ARIN  
OrgTechName: IP SWIP  
OrgTechPhone: +1-888-613-6330  
OrgTechEmail: swipid@nipaweb.vip.att.net

# ARIN WHOIS database, last updated 2003-10-16 19:15  
# Enter ? for additional hints on searching ARIN's WHOIS database.

Which places us in New Jersey, even further afield.

Even commercial providers of location information don't dare claim accuracy above 70% at the region level, and note that a vast subset of Internet users cannot be located at all, for example:

“ow many countries are included in the database? What is the accuracy?”

The database has over 95% of accuracy in country and ISP level, 70% in region level and 65% in city level, which is higher than any of our competitors. The country-level inaccuracy is due to dynamic IP address allocation by large ISPs such as AOL and MSN TV. Because AOL uses a network that routes all of the company's Internet traffic through Reston, Virginia. All IP-based geo-location services, including IP2Location, are unable to determine the state and city for people who dial into the AOL

network. The region-level and country-level inaccuracy is due to the flexibility given to each ISP to re-assign dynamic IP address within their service area.”[9]

At the time of writing AOL had in the vicinity of 20 million subscribers.

## 6.2 Possible Modifications to the Infrastructure

Most people use one of two methods to connect to the Internet from their home. Either they use dialup, or they use broadband. In either case, their ISP could, in theory, tie the phone number (used for dialup) or cable or ADSL district (used in broadband) to the IP address and make that available for query. This has interesting economic and privacy implications, of course. Firstly, the change required to every ISP’s infrastructure would be enormous - many ISPs do not have appropriate equipment to report this information at all, let alone make it available to others. The privacy implications must be obvious: if anyone could get the phone number or street address of any other user of the Internet, this would be a serious problem.

Ignoring those issues, there remains the question of accuracy. Although it is theoretically possible to nail a phone number down to the exact house it is in, this is, for obvious reasons, not information that is normally made available. The publicly available information is rather coarser than that, and generally resolves only to the city level. For example, querying it for Mr. Wirenius’ number (212-533) resolves only to “New York City Zone 1”[1].

A lower impact change would be to make available the location of the dial-up number the user dialed into, or the local router for cable users – neither of these give a particularly great idea of neighbourhood – the typical ISP has only one or two dial-up numbers per major city, and cable routers can service large areas.

In order to make these changes, every ISP would have to at least add extra services to their existing equipment, or potentially replace it with new equipment. Ordinary users of the Internet, both end users and operators of web servers, could not implement these changes themselves.

## 7 Modifications to Protocols

Another possibility is to modify Internet protocols such that the client reports its location to the server. This sounds attractive, but actually has little merit,

for several reasons. First and most obvious is that the information would have to be configured by the user, so they could trivially forge it. Secondly, the protocols exist and do not carry the required information, so any such scheme would have to be opt-in, and, again, opting-out would be a trivial way to avoid being located. Finally, the cost of replacing every browser on every machine in the world, even if it could be mandated, would be extraordinary.

The appropriate body to make such changes would be the Internet Engineering Task Force, who own all Internet protocols. The process for changing protocols is lengthy, and in general must be justified on technical grounds – which would seem troublesome in this case.

## 8 Deliberate Evasion

So far we have only discussed what is possible if everyone is willing to play along, and not try to find ways around the system. Unfortunately, it is fantastically easy to deliberately evade geolocation even if you are someone who has not accidentally evaded the system by using AOL's proxies, or because you are hidden behind a corporate firewall using NAT.

As discussed above, a proxy server makes it appear that all traffic originates from the proxy rather than the real client, and hence the location (if available at all) would be appear to be the proxy's location. Many such proxies are freely available on the Internet – and they're easy to find. All a user has to do is point his browser at such a proxy and his location is completely invisible to the server.

Another method of evasion for dial-up users, if only the location they've dialed into is advertised, is to choose a dial-up that is in a different neighbourhood from their own.

## 9 Peer-to-peer Protocols

It should be noted that the Web is not the only way to publish content. Peer-to-peer networks are an increasingly popular mechanism, for example the well-known commercial network, KaZaA, and various free networks, such as FreeNet or BitTorrent. In these networks the originator of the content has no control over the final recipients, and cannot even know who they are, since the content is eventually distributed over many of the machines connected

to the network, each of whom can potentially deliver it to anyone who wants it.

Indeed, many of these networks have the specific design aim that the delivery of content should be impossible to control once it has been injected into the network (of course, this is not intended to negate the recipient's obligation to obey copyright and other applicable law). Controlling content delivered according to location is still technically feasible in such networks, but would require software modifications to all participants in the network – including, of course, those not based in the US, who would have no incentive to use the modified software. Furthermore, it is difficult to imagine how content would be tagged as appropriate for each location other than by listing all the locations in which it is acceptable – a list which, it seems to me, would be practically impossible to create and maintain as well as being very large – I am informed that another expert (Jeffrey J. Douglas, Esq.) will address this issue, and that the current guess is around 1,500 to 2,000.

## 10 Cost of Implementation

The cost of implementing can be broken down into two parts. The first component is the cost, given the location of the client, of making the site choose and deliver appropriate content for that content. The second is the cost of implementing systems to determine the location of the client. This cost varies according to the various methods of implementation.

### 10.1 Delivering Appropriate Content

Traditionally, many websites, including Ms. Nitke's, are constructed as “flat” sites – that is, their content is entirely contained in static text and graphic files, edited either by hand or in one of the many cheaply available HTML editors. Such websites are very easy to create and maintain, and also easy to publish: it is merely a matter of copying all the files to a webserver.

In order to serve tailored content according to location, this must change. Each page must, at the very least, have alternatives for each possible location (or group of locations). Ms. Nitke estimates that for the purposes of her work there are 1,500 to 2,000 different “types” of community. Even allowing for the fact that for each individual page there must be some overlap between communities, it would seem to be a conservative estimate that 100 different

versions of each page would need to be maintained – and, of course, each of the 2,000 different communities assigned to each of those pages.

In addition to this cost, it is also necessary to convert the website to “dynamic” – that is, the webserver must decide, based on the client’s location, which version of the page to serve. This adds cost and complexity to the website. Firstly, the code that makes that decision must be written and tested (or purchased). Secondly, the webserver must be prepared to run code, which is not a function usually offered on the cheapest webserver, so the cost of renting the website is increased. Thirdly, the webserver must be configured and tested with the software chosen.

Also, every time the site is updated it must be checked that it still conforms with the content policy. This causes two further costs: the cost of a second web site to test the new content on<sup>3</sup>, and the cost of confirming policy is still adhered to, which would require visiting every page on the site pretending to be in every group of locations, and checking the content by hand<sup>4</sup>

It could be argued that automation could reduce the costs by cunningly fabricating appropriate pages from snippets, given the permissible content for the individual users - however, the difficulty of maintaining good design and comprehensible text (bearing in mind that it is not just the visual content but supporting verbiage that must be tailored) when such an approach is used means that even the largest corporations have failed to use this technique except in the crudest ways.

There is also a hidden cost: security. Static websites are inherently less susceptible to attacks on their content; they have no code to attack. Switching to a dynamic website introduces a potential for security holes in the software used to determine location and serve appropriate content. Since, because of cost issues, this code is likely to be home-grown, and bitter experience shows that most users are incapable of writing secure code, this risk can be quite high. The cost of a security breach can vary from embarrassment (the attacker modifies the website causing you to look foolish), to major financial damage (the attacker gets access to your bank account, or clients’ credit card details), to privacy breaches (the attacker gets access to clients’ personal information).

Quantifying these costs, Ms. Nitke estimates that she has spent around 75 hours and \$2,000 over 2 years on the maintenance of the site, which is an

---

<sup>3</sup>Because of the specialised nature of webserver it is usually very difficult to test on the user’s own PC; instead a server with the same setup as the “live” server must be used.

<sup>4</sup>Note that for most static websites such a check is not necessary at all, since all content is visible to all comers.

average of 3 hours and \$80 a month. Multiplying that by 100 gives 300 hours or 37.5 days a month – in other words, a little less than two full-time jobs – and \$8,000 per month in costs.

## 10.2 Adapting Infrastructure

Since the required information cannot be made available without modification to the Internet infrastructure, the cost of making those modifications must also be factored in. The simplest scheme I can imagine in terms of delivery is to be able to query, for any particular connection<sup>5</sup>, the location of the eventual user of that connection. This would require a number of linked changes.

First of all, the information would have to be made available for every endpoint of every user of the Internet (i.e. every dial-up connection, broadband connection or machine in an office). Estimating the cost of doing this is a little difficult, but perhaps a starting point is to estimate how many hosts would have to be catered for. According to the Internet Software Consortium[2], there were, in January 2003, 171 million hosts advertised in the DNS. Of these, 40 million were in the .com domain. Unfortunately, for historical reasons, although .com is the domain of choice for US-based machines, it is also widely used in other parts of the world. The largest single-country top-level domain, .jp – Japan, has about 10 million machines. Assuming that the US is somewhere between the two, that would give us 20 million machines. However, we should note that the vast majority of machines are connected by ISPs who do not put them into the DNS.

Another approach is to work backwards from the number of users. According to Harris Interactive[3] in February 2003 140 million Americans had Internet access. Given that of these, nearly 30% had access from work and over 55% from home, an estimate of one machine per user would not seem unreasonable.

So, we can take 20 million machines to be an absolute lower limit, and 140 million to be a reasonable estimate of the number of machines affected.

We now should consider that for many of these machines, their connection will be proxied via some kind of firewall or gateway, and each of these will have to be modified to query the geographical data of the user and make it

---

<sup>5</sup>Note that being able to query by IP address is insufficient because of proxies, NAT and so forth. For this reason, the existing DNS LOC records, defined in RFC 1876[8], are inadequate for the purpose.

available for their proxied connection (in the case of caching proxies these would also have to be modified to cache according to community [which would have to be derived from the location, of course]). Naturally, there will be far fewer of these gateways, but it will be much more costly to modify each one. It does not seem unreasonable to me to estimate that for each 100 users there will be one gateway (if this estimate is too low, then that will be offset by the increased cost of modification of larger gateways). This gives us around 14 million gateways.

Assuming the cost per end-user machine of making these modifications is \$10, and per gateway is \$1,000, that gives us a total cost of approximately \$2.8 billion to modify clients.

The cost of modifying and maintaining servers is a slightly greyer area. Assuming that in general, servers classified as hosting pornographic material would be required to make these kinds of modifications, as well as others such as Ms. Nitke's, that gives us a lower bound for the number of affected servers. CyberAtlas[4] estimate that 12% of world's web servers, about 4.2 million, host pornography. Netcraft[5] estimate that over half of these are in .com and .net, and we can guess that around half of those are based in the US, giving us around 1 million affected servers.

Assuming that each of these would, like gateways, require \$1,000 of software changes, and also, unlike client and gateway machines, regular maintenance of content as described above, because of regular changes, taking one full-time person paid \$30,000 per year per website, this gives us an initial cost of \$1 billion, and a annual cost of \$30 billion (it is worth noting that the entire Web market in all categories is estimated to be worth \$57 billion worldwide).

### 10.3 Traffic Impact

A final effect to consider is the amount of extra traffic that location determination will cause. Again, this is problematic to determine, but a basis for estimation is to consider that for each HTTP connection (i.e. every time a user connects to a server), it will be necessary to interrogate the server holding the location information for that user. Assuming that this is a relatively small exchange, say of the order of DNS interrogations, this amounts to around a kilobyte of data in total for each query.

Unfortunately, because of the way HTTP works, most requests are relatively short and connections do not have a long lifetime (most busy servers are configured to drop connections within 1 second of inactivity, or even as soon as the request is fulfilled). Because of this, studies[6] have shown that over

75% of web traffic is actually under 1 kilobyte in size. Therefore adding geolocation would increase the traffic for HTTP by somewhere in the region of 100% or more. Studies[7] of traffic patterns have shown that HTTP traffic is over 50% of all the traffic on the Internet. So, this would suggest that geolocation would increase the total traffic on the Internet by something in the region of 50%. Since the cost of maintaining the Internet's infrastructure is tightly coupled to the amount of traffic, this would suggest a commensurate increase in that cost.

It is worth noting that the usual method for mitigating this kind of traffic, local caching, doesn't work because of NAT and proxies – since each network connection uses a different source port address, and since NAT and proxies would cause connections with the same IP but different ports to correspond to different locations, caching does not seem to me to be possible.

## **11 Conclusion**

Although it is technically possible to modify the Internet such that geolocation of users is possible, it is a fantastically expensive change, both for the infrastructure of the Internet and for operators of websites.

Without these modifications, it appears to be impossible for Ms. Nitke (or anyone else) to make the required changes to her website.

Most importantly, even if such a level of expense were acceptable, it would still be trivially easy for users to deliberately evade the controls (by using proxy servers, for example), and, unless great care was taken, many users would accidentally evade them. In the case of peer-to-peer technologies, originators of material have no control over who sees it at all.

Also, it is worth noting that if only the United States were to implement these changes – or, indeed, if any other country were to not implement them – then the proxies (or other means of evasion) could not be forced to comply, nor could peer-to-peer software or other methods of distribution.

## **12 Opinion of Witness with Basis and Reasons Therefore**

For the above reasons, it is my opinion that it is technically and economically infeasible for Ms. Nitke to comply with the technical requirements of the CDA.

## 13 Data or Other information Supporting the Opinion

See the bibliography below, also my previous research for the Yahoo! case (see below), my general experience of the Internet and its infrastructure and conversations with other experts on this and related subjects.

## 14 Qualifications and Resume

### 14.1 Resume

I am Technical Director of A.L. Digital Ltd., a company providing Internet consultancy, specifically in the area of privacy and security, and also providing hosting for Internet servers.

I am a founding director of the Apache Software Foundation, a not-for-profit corporation registered in Delaware, which is responsible for the world's most widely used webserver, Apache. I am also one of the developers of Apache.

I am one of the founders and developers of OpenSSL, a widely used free cryptographic toolkit.

### 14.2 Publications

“Apache: The Definitive Guide”, three editions, published by O'Reilly, latest edition in 2003, with Peter Laurie.

“Forward Secrecy Extensions for OpenPGP”, an Internet Draft, available at <http://www.apache-ssl.org/openpgp-pfs.txt>, with Ian Brown and Adam Back.

“Geometry and Physics of Catenanes Applied to the Study of DNA Replication”, B. Laurie et al., *Biophysical Journal* (June 1998), 74, 2815-2822.

“Ideal Knots (chapter)”, Ed. A. Stasiak, V. Katritch and L.H. Kauffman, World Scientific (1998), ISBN 981-02-3530-5.

“Security Against Compelled Disclosure”, <http://www.apache-ssl.org/disclosure.pdf>, with Ian Brown.

“Seven and a half things about PKI”, <http://www.apache-ssl.org/7.5things.txt>.

“European Working Group on Libre Software”, <http://eu.conecta.it/>.

“Sedimentation and Electrophoretic Migration of DNA Knots and Catenanes”, A. Volgodskii et al., J. Mol. Biol. (1998), 278, 1-3.

“Signatures: an Interface between Law and Technology”, <http://www.apache-ssl.org/tech-legal.pdf>, with Nicholas Bohm.

### 14.3 Previous Testimony as an Expert

I acted as an expert in *Le Ligue Contre Racisme et L’Antisemitisme v. Yahoo, Inc.*, No. RG 00/05308 (Jean-Jacques Gomez, J.).

## References

- [1] “NPA NXX Lookup” – <http://puck.nether.net/npa-nxx/>.
- [2] “Internet Domain Survey” – Internet Software Consortium – <http://www.isc.org/ds>.
- [3] “Those With Internet Access Continue To Grow But At A Slower Rate” – Harris Interactive – [http://www.harrisinteractive.com/harris\\_poll/index.asp?PID=356](http://www.harrisinteractive.com/harris_poll/index.asp?PID=356).
- [4] “Porn Pages Reach 260 Million” – CyberAtlas – [http://cyberatlas.internet.com/big\\_picture/traffic\\_patterns/article/0,,5931\\_3083001,00.html](http://cyberatlas.internet.com/big_picture/traffic_patterns/article/0,,5931_3083001,00.html).
- [5] “Netcraft Web Server Survey, October 2003” – Netcraft – <http://www.netcraft.com/Survey/Reports/0310/>.
- [6] “NeTraMet Web Session Performance” – CAIDA – <http://www.caida.org/analysis/workload/netramet/web/>.
- [7] “oc48 statistics” – CAIDA – [http://www.caida.org/analysis/workload/byapplication/oc48/20020305/time\\_perc\\_-20020305/20020305\\_b.xml](http://www.caida.org/analysis/workload/byapplication/oc48/20020305/time_perc_-20020305/20020305_b.xml)
- [8] “A Means for Expressing Location Information in the Domain Name System” – C. Davis, P. Vixie, T. Goodwin and I. Dickinson – <http://www.ietf.org/rfc/rfc1876.txt>.
- [9] “IP2Location<sup>TM</sup> IP-Country-Region-City-ISP Database FAQ” – IP2Location – <http://www.location.com.my/README-IP-COUNTRY-REGION-CITY-ISP.htm#17>.

I declare under penalty of perjury that the foregoing statements are true and correct.

Dated: London, England October 5, 2004

----- Bennet Laurie