



## **Spoofing – arts of attack and defence**

### **Introduction**

Spoofing means pretending to be something you are not. In Internet terms it means pretending to be a different Internet address from the one you really have in order to gain something. That might be information like credit card numbers, passwords, personal information or the ability to carry out actions using someone else's identity.

Some potential spoof attacks are technically difficult. Others are commonplace. There are different ways in which you can defend against actual or attempted spoofing. Some attacks rely upon user error or misunderstanding. These are more difficult to prevent because users do not necessarily have the technical subtlety needed to understand what is happening, whilst the technology they use does not explain itself very clearly in non-technical terms.

In this white paper, several spoofing attacks, and possible defenses against them are considered.

### **DNS server spoofing attack**

The most complex attack is to alter the address the master DNS servers will resolve for a given URL. The URL that an Internet user types in is not the numeric address of the site required, but an alphanumeric address structure. The DNS servers convert, say, [www.articsoft.com](http://www.articsoft.com), into a real Internet address, say 195.217.192.145 (not the correct address, but the point is made). This has to be done because people don't generally remember and associate 12 digit numbers with anything except telephone numbers, and then they generally file them on the telephone with a 'friendly name' that they have some relationship with.

An attack of this type has been successfully mounted that altered the server list, so that, for a period of time, users requesting some sites were directed to the wrong addresses.

This type of attack is a major threat and the Internet naming and addressing authorities have taken it very seriously indeed. DNS servers have incorporated numerous security measures to prevent repetitions of this attack from being successful. These include having the servers mirror and monitor each other as well as controlling very carefully how updates are introduced into the servers. This kind of problem can be resolved by positive site identification, where the end user is able to automatically check the claimed web site URL against the content provided, as provided by the ArticSoft approach.

### **Web site names and addresses**

There are many ways in which a web site may be spoofed. This section covers popular methods.

## Content theft

A copy of a site can be created from the original by copying all the publicly accessible pages from a site to another server. Copying a publicly accessible site is automated through the use of programs called 'spiders'. You will find many programs freely available that are designed to copy whole websites so that the program user can read a web site offline rather than have to stay connected to the Internet.

Some spider activities are legitimate – maintaining mirror copies of the site to improve accessibility, or search engines looking for text and keywords to add to their catalogues. Search engines also maintain caches of pages for their users so that load times can be reduced. Other spider activities may not be. A web site owner may be aware that a 'spider' is reading his site, but he cannot know its real intent any more than you can know why a person reads a book.

The technical defenses against this attack are few. Sites have to make a large amount of their content publicly available – so naturally they want people to obtain that content. Software is badly placed to be able to decide what name a user really intended (what is a typing error?) and there would be an adverse reaction if systems were written to decide such things on behalf of the user.

Sometimes the copying may be to present your information as that of another site. This is sometimes referred to as deep linking, not spoofing. A number of schemes for 'watermarking' images have been invented in recent years to help detect this kind of attack, and an excellent reference is <http://www.cl.cam.ac.uk/~fapp2/steganography/> where the background, history and actual effectiveness of the technique(s) that have been developed is discussed.

## Name similarity

The simplest spoof is to catch the people who mistype the web URL they are looking for, or put the wrong locator at the end. For examples of similar names try [www.whitehouse.com](http://www.whitehouse.com) (it should have been .org) or [www.nasa.com](http://www.nasa.com) (similar problem). Other famous examples include [www.nescape.com](http://www.nescape.com) and [www.mcrosoft.com](http://www.mcrosoft.com). Sometimes the content makes it obvious to the user that the site is not the one they were expecting, but it doesn't have to. That is down to the creator of the similar site. To give some examples where sites are being directed to what might be considered an unexpected destination, most likely because of a spelling mistake: (please note that there is no suggestion that these are hacker sites, but the redirection stated here was verified on 2 April 2002):

[www.mcrosoft.com](http://www.mcrosoft.com) = 195.184.248.163/ML\_HomePage.aspx

[www.whithouse.org](http://www.whithouse.org) = [www.amatuervideos.nl](http://www.amatuervideos.nl)

[www.harods.com](http://www.harods.com) = [www.shopndrop.com](http://www.shopndrop.com)

[www.walmaat.com](http://www.walmaat.com) = [www.search.linksponsor.com](http://www.search.linksponsor.com)

On 24 April 2002 the author was offered the ability to register the web site [WWW.VERISIGN.COM](http://WWW.VERISIGN.COM). (Shown in a different font this is [www.verislgn.com](http://www.verislgn.com) but how would you have known?) So registering web sites with similar names to real businesses is not so difficult, and a fraudster will have used a stolen credit card so they won't be found!



For an excellent tutorial on how to carry out this kind of attack and the uses it can be put to see the web site [www.reamweaver.com](http://www.reamweaver.com) which claims to provide downloadable software for this purpose.

The most effective defense against this kind of confusion is probably procedural. Users still have to take care that they do not put in the wrong name. But you cannot expect users to get their typing to be perfect on every occasion.

Altering the registration rules for Internet names to prevent registering names that are very close to those of registered companies or organizations could well help prevent this problem.

Usually national laws prevent people from registering company names that are similar to existing ones for exactly this reason (there is, in some jurisdictions, the offence of passing off), so the justification for allowing it on the Internet is muddy at best. It has taken us many hundreds of years to establish the rules for trade in the terrestrial world and it seems unreasonable to take a stance that says somehow the Internet can ignore the experience that made those laws necessary, any more than Newton's apple could have tried to ignore his law of gravity. (Selling domain names is a business whilst registering a company is something controlled by law. As a result, there may be differences in approach. Law evolved to make company registration a formal process for good reasons that the Internet does not appear to have fully recognized.)

### **Link alteration**

Another attack, that offers far more gain to the hacker for rather less actual work, is to alter the return address in a web page sent to a user to make it go to the hacker's site rather than the legitimate site. This is done by adding the hacker's address before the actual address in any page that has a request going back to the original site. Literally, where they see a reference to <http://www.mysitemname.com> they add their own address to it to make <http://hackersite/http://mysitemname.com>. You will notice that the fake site is recognized as a valid URL address.

The hacker only need to do this once to get a link into the communication between browser and server and they can reprocess all the communication from then on, including SSL connections. Since the user is familiar with seeing the site connections and names and even server certificate details constantly changing without any explanation or obvious reason, they are not likely to notice this change at all.

The commonest form of defense used by web sites at the moment is to apply 'digital signatures' to their web pages, which are checked as they are leaving the server to ensure that nothing has been changed. The idea is to prevent altered pages from being able to enter the Internet. The big drawback with the approach is that pages can easily be altered if cached on web servers. The end user cannot see either the original web site checks (and caches have no checks) and therefore has no idea if the pages are valid when they arrive. They may have been cached at one or more locations where they could have been attacked, which is rather easier than trying to alter them as they go by.



Recent developments by ArticSoft have produced a system that provides end users with continual verification of pages back to site URLs. This is a much more powerful technique for several reasons:

- Most important is that it gives the end user actual information to act on rather than leaving them guessing;
- Secondly, because, as well as detecting page alteration, their method prevents pages from being routed through a site that is not the originating site, and it therefore prevents this type of spoofing;
- Thirdly, it also resolves the problem in SSL that a hacker can, by altering addresses, get into the middle of the connection between the user and the real site and read all the supposedly protected information.

This latter feature is possible because most sessions do not, or cannot verify the user identity to the server, and the user does not know what identity the SSL connection should have.

SSL is a technology that has succeeded largely because few users understand it (or the padlock on the browser) at all. An SSL link for commercial sites is started by the browser, without validating where it is linking to. Nothing happens in the browser to confirm what the link is with, or if it is valid and there is no checking that the source of the information passing across the link has come immediately from the expected site. A great deal of faith is put in the user checking all the details for themselves, which stands in great contrast to the idea that systems are intuitive and easy to use.

A variety of academic papers have been published detailing attacks that defeat SSL and demolish many of the claims made for its capabilities have been published. These include:

[www.cs.princeton.edu/sip/pub/spoofing.html](http://www.cs.princeton.edu/sip/pub/spoofing.html),  
[www.bau2.uibk.ac.at/matic/spoofing.htm](http://www.bau2.uibk.ac.at/matic/spoofing.htm),  
[www.cs.dartmouth.edu/~pkilab/demos/spoofing/](http://www.cs.dartmouth.edu/~pkilab/demos/spoofing/)

User lack of understanding is further undermined by techniques that are common industry practices which confuse security. 'Wildcard' server certificates are used by many sites rather than proper individual names. ISP certificates are often used as the common certificate for all their hosted web sites. The use of third party secure services for payments systems with completely different site names also confuses the situation. Users can hardly be expected to understand what to them are arcane practices that have no apparent explanation.

On balance, it is just as well that the end user remains blissfully ignorant. However, this is the very ignorance that fosters hacking and spoofing. Changing this situation is not a matter of expecting all users to become expert technologists. Education is required, but so are appropriate methods of development behaviour that increase understanding and security as principles and best practice.

## **IP addresses changing attacks**

Hackers are able to configure themselves (their messages over the Internet) to have any IP address that they want, so they can appear to be part of an internal network when in fact they are external, or appear to be the address that you want to connect to. This is a subtle attack because it may be used on the Internet, intranet or extranet equally well. Many networks are set up to dynamically allocate addresses, and software monitoring techniques to reveal information flowing around networks allow hackers to select valid addresses so that they can impersonate valid sessions. Alternatively, the hacker may try to capture a valid available address.

Defenses against this kind of attack are often firewall based. Firewalls can be set to perform network address translation so that internal addresses are not disclosed to the outside world. Also, firewalls can be set to discriminate between connections that are internal from those that are external but appear to be using internal addresses. However, if an attacker can gain access to an internal network they can bypass external firewall checks. To guard against this situation some organizations, particularly financial ones, use internal firewalls to control and limit the potential for this kind of attack.

## **E-mail address changing**

In particular services, such as e-mail, the potential to spoof the apparent source address continues to be a problem. Most users are unaware that the apparent address is unreliable, and that replying to the apparent address may actually send a message to an unintended destination. (An example is replying to a message from an individual that was forwarded by a distribution group. The reply goes to the group, not the individual who appeared to be the source. This is not regarded as an error, although it is actually a security failure because the user is not aware of what is happening and in theory they are in charge.)

E-mail with secure attachments may be prone to spoofing as well. Users may assume that the header of the e-mail and the secure body are related to each other when that is not the case. User education is much more difficult in this case because it is not in the least clear to the user why secure messages aren't. The only way to protect e-mail effectively is to ensure that the whole object is protected, not just parts of it. Users will also have to accept that the addressing information in e-mail cannot be private.

## **Review of the current situation**

Spoofing attacks are based upon the ability to make a user believe that they are securely connected to a network address, or receiving e-mail from a specific source, when that is not the case. The problem stems from the fact that the whole of the addressing system on the Internet is not secure. This creates problems of spoofing in many areas outside web addresses, including e-mail.

Since there are currently no effective means of securing the addressing (unless everyone 'knows' everyone else, the attempts to secure links between address points are flawed, and unless there is a move to mandate absolute identification of all Internet users (politically unlikely given requirements for anonymity that exist in US law for certain types of transactions) they will remain so.



## **The best way forwards**

A change is needed to move from relying on networking systems that don't solve the problem to content management – signing and protecting the actual information itself and not just the unproven link(s) it is traveling over. That prevents all the typical network IP attacks from having any effect, and provides genuine control over the information itself.

A change to securing content, rather than links, offers the e-business community significant benefits. For e-business, there is an imperative for the honest trader to identify themselves by clearly identifying their content. (How you link to them is then, actually irrelevant.) That way all their users can verify any content reaching them, and rely upon what that content is, regardless of how it got to them. The same would go for instructions to computer systems, services and networks.

By switching to that approach, the business community can achieve major trading benefits: certainty that the quality of their information can be proven; certainty of secure trade for them and their customers; certainty of privacy for them and their customers; certainty that payment details cannot be misused.

Conversely, traders not following such an approach identify themselves as leaving their customers open to fraud, misrepresentation, uncertainty and lack of confidence. Right now schemes to separate the good from the bad have little effect.

ArticSoft have provided some novel steps in the direction of proof by content rather than proof by network connection. For Internet technologies this is a more pragmatic way to proceed because content may reside anywhere on the Internet. It also allows for protecting information that is confidential by much simpler methods than are offered by network based solutions.

Such a change faces significant opposition, not least from the network providers, network analysts and managers, who risk being relegated to a lower status (and relative income) as a result. In practice, with the tools available, they have done the best job that could be done. Unfortunately, scripting attacks and cook book hacking methods are making those methods more vulnerable, and a change in approach is needed moving forwards.