

# **Royal Holloway University of London**

## **Computer Security**

### **Assignment**

#### **Security Problems Connected to the Address Resolution**

Group Members:

Han Lai

Yuan-Fang Li

Whye Ho Loo

Chin Hann Chong

Kar Lai Toh

Anniya Tsai

## TABLE OF CONTENTS

<b>1.0 Introduction to Address Resolution.....</b>	<b>3</b>
1.1 What is DNS & how DNS works? .....	3
1.2 What is ARP and how ARP works? .....	5
1.3 Security problem related to Address Resolution Protocol.....	8
<b>2.0 Security concerns within DNS.....</b>	<b>9</b>
2.1 Dynamic environment.....	9
2.2 Cache Poisoning.....	10
2.3 Java applet vulnerability .....	11
<b>3.0 Introduction to IP spoofing .....</b>	<b>13</b>
3.1 Man-in-the-middle.....	14
3.2 Routing redirect.....	15
3.3 Source Routing.....	15
3.4 Sequence Guessing.....	16
3.5 Flooding .....	17
3.6 What can be done to prevent IP Spoofing Attacks? .....	18
<b>4.0 Conclusion .....</b>	<b>19</b>

## **1.0 Introduction to Address Resolution**

This section will discuss two types of address resolution. The first type is the Domain Names to IP address resolution and the second type is the IP to MAC (Medium Access Control) address using address resolution protocol (ARP).

Our main focus is on the Domain Names to IP address resolution security related problems but not the least we will also discuss the security problem related to ARP.

### **1.1 What is DNS & how DNS works?**

Computers use their IP (Internet Protocol) addresses to communicate with the other computers in the Internet. In order to communicate with another computer, one needs to know its IP address. IP addresses like 193.124.56.214 or 157.46.163.25 are difficult to remember and therefore they have to be assigned with their domain names respectively. Domain Name Server (DNS) provides such a naming service that enables the translation of the host names to IP addresses. It is a TCP/IP service that maps an IP address like <http://153.20.24.72> to <http://www.np.edu.sg>. It must be noted that though an IP address will map to its host name but it is also possible to associate multiple IP addresses to a single host name. For example, to enable load balancing in a network, a heavy-duty server has to share its task with other servers, such that these servers will have to be assigned with the same host name to response to the clients speedy requests. The IP addresses and names of the respective hosts are stored in databases that reside in the DNS. It is impossible for the DNS to hold the entire domain names and IP addresses of

the hosts in the Internet. Therefore one DNS has to depend on the other to provide such information, as they are stored in a distributed form.

DNS was not designed to be secure. It is vulnerable to a technique known as *DNS Spoofing*. So what is DNS spoofing? DNS spoofing means that a computer on the Internet has been able to advertise a false IP address for itself. This is due to the easy setting up of one's DNS resolver and feeding it with false information. Consider the following two common DNS spoofing:

1. A malicious DNS advertises false IP address to a victim. Subsequent DNS queries are sent bogus IP address, traffic is sent to wrong host.
2. Java applets could be created to spoof IP address in order to penetrate into insecure server. This unprotected server in a network thinks that its being contacted by a trusted host but in fact, the machine on the end of the connection is operated by some malicious users. Access restriction based on the host name of the connecting browser is vulnerable to such attack.

Fortunately there are two ways to prevent DNS spoofing. The Web server can be configured to perform paranoid checking. This means when there is an incoming connection, the Web server will pull the IP address of the browser, then makes two calls to DNS system. First, the server requests the DNS to return the host name registered for the IP address. Next, the server will request the DNS for the IP address of the host name it just

returned. However, there is another drawback for this solution as it is prone to IP spoofing! IP spoofing and its types of attacks will be discussed and evaluated in Chapter 3.

Another way to avoid DNS spoofing is to use a properly configured firewall to provide reliable DNS lookups that are immune to such attacks. Chapter 2 will further discuss the security issues based on DNS spoofing.

## **1.2 What is ARP and how ARP works?**

Address Resolution Protocol (ARP) is a protocol used by IP (Internet Protocol) which is a network layer protocol to map IP network addresses to the hardware addresses which is in the data link layer protocol. The protocol is used when IP is used over Ethernet.

Most of the hosts are attached to a LAN by an interface board that only understands LAN addresses. Every manufactured Ethernet board comes with an equipped 48-bit Ethernet address. The board sends and receives frames based on 48-bit Ethernet addresses. They know nothing about 32-bit IP addresses.

The address resolution refers to the process of locating an address of a machine in the network. The machines in the Ethernet environment operate at the data link layer. These machines do not understand IP addresses, which operate in the network layer. In order to communicate between machines, they must know which IP addresses map to which MAC addresses. This mapping is done in the ARP client and server processes.

on all computers using IP over Ethernet. For example, an IP address of 192.169.20.1 is mapped to a Ethernet MAC address of OA:4B:00:00:07:08, so the sender will know that the IP address of 192.169.20.1 belong to which machine. The processes are normally implemented as a part of the software driver of the network interfaces card.

There are four types of ARP message, which may sent by the ARP protocol. These messages are identified by the values in the operation field of an ARP message. The types of message are:

1. ARP request
2. ARP reply
3. RARP request
4. RARP reply

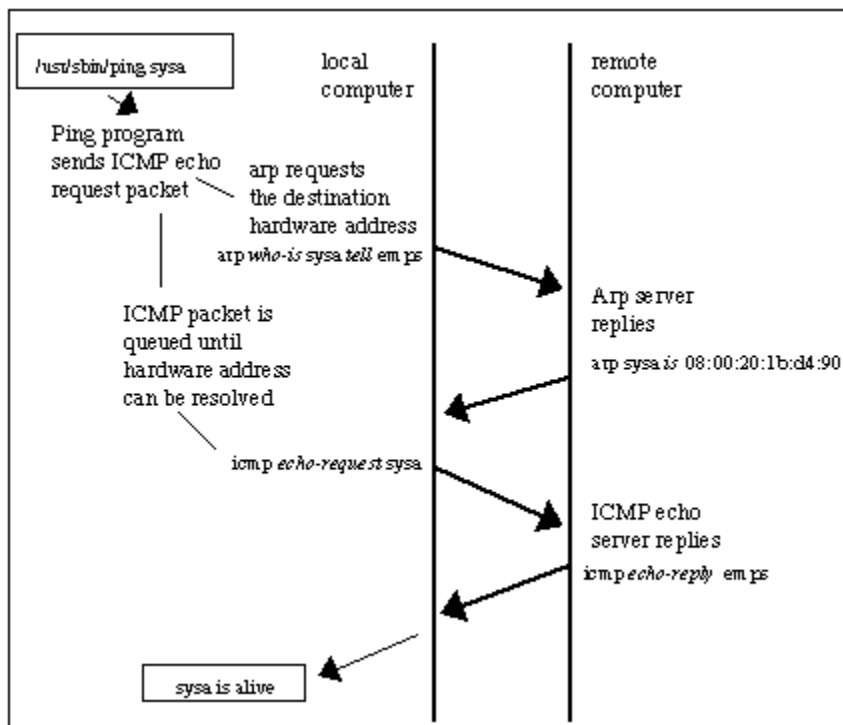
The format of an ARP message is shown below:

0	8	15	16	31
Hardware Type		Protocol Type		
HLEN	PLEN	Operation		
Sender HA (octets 0-3)				
Sender HA (octets 4-5)		Sender IP (octets 0-1)		
Sender IP (octets 2-3)		Target HA (octets 0-1)		
Target HA (octets 2-5)				
Target IP (octets 0-3)				

Format of an ARP message used to resolve the remote MAC address [1]

In order to reduce the number of address resolution requests, the client normally caches the resolved addresses for a short period of time. The size of the ARP cache is of a finite size, it will become full of incomplete and obsolete entries for computers that are not in use if it was allowed to grow unchecked. The cache is periodically flushed of all entries. This deletes unused entries and frees the space in the cache. Any unsuccessful attempts to contact computers which are not currently running are also removed.

*“The figure below shows an example of the use of ARP when a computer tries to contact a remote computer on the same LAN (known as “sysa”) using the ping program. It is assumed that no previous IP datagrams have been received from this computer and therefore ARP must be used to identify the MAC address of the remote computer.” [2]*



### 1.3 Security problem related to Address Resolution Protocol

This section will discuss an example of the attack on the ARP and also the solution to avoid the attack. Other types of related attack will be covers in chapter 3.

The weak part of a system in which machines communicates using ARP is that authentication is based on the hardware address only. This will lead to attack known as *ARP Spoofing*. To communicate with another machine a host must have the machine hardware address (MAC). If the host doesn t have any entry of the MAC address of the machine in its ARP cache, it wills sends a broadcast message (containing the IP address of the machine) to the entire host in the subnetwork. The message is captured by the network interfaces and interrupt request to the operating systems is issued. The host with the corresponding IP will answer to the request. A hacker can exploit this MAC address authentication by disabling a trusted host and spoofing its MAC address. So in doing this the trusting machine is deceived that the hacker s machine is the trusted host.

ARP spoofing can be prevented by either stopping the usage of ARP for obtaining the MAC address of the trusted host by keeping a permanent entry of the MAC address of the trusted host in the ARP caches of the trusting machine.

## 2.0 Security concerns within DNS

The biggest security issues within Domain Name Service protocol are Domain Name issues and IP address management. Users in Client side using browser rely heavily on the Domain Name Service, and are thus generally prone to security attacks based on the deliberate mis-association of IP addresses and DNS names.

DNS Spoofing is a serious attack model based on the strong trust relations between client machine and their DNS server. The DNS server can be attacked in many different ways but all rely on the translation between Domain Name and IP address.

### **2.1 Dynamic environment**

Security has become a major concern when it comes to DNS and DHCP in a dynamic environment. The DNS and DHCP services must be available to the client systems openly to provide host names and IP addresses updates immediately for those mobile users within a dynamic environment. Unfortunately, this makes the system susceptible to attacks.

There are no authentication mechanisms in the DNS protocol and the DHCP service currently. This give the spoofing attacks an easy way to walk in to the systems. There are some new standards being developed that deal with security weaknesses in the original specifications for both DNS and DHCP.

An example of spoofing is an attack combined with DHCP and DNS, which is when an attacker sends information to an IP station, impersonating as a DHCP server and telling the workstation that IP information changed, and configuring it to using a different DNS server than usual. This DNS server can be a server controlled by the attacker with false entries. Now, when the user wants to buy something online, it will not get the user to the web site he intended to go, but to a web site ran by the attacker. The attacker can now obtain credit card information and other personal details, just pretending to be a well-trusted web site. When a user types a domain name in the browser, the browser will query the DNS for the web site s IP address. If the attacker runs the DNS, they can return a false answer.

The solution for against this DHCP attack is to set up a firewall to block incoming DHCP packers, DHCP packets should only pass through within the local network and inaddition making the Domain Name more secure.

## **2.2 Cache Poisoning**

Another security concern regarding to DNS is the process of Cache Poisoning . It is a process where attacks are launched on cache data of DNS in order to misdirect and intercept packets on domain name servers. One simple type of attack is where bogus data from a remote name server is offered to our name server, which in turn stores the misleading data in its cache. By providing false host name and mapping information, the

attacker can misdirect name resolution mapping, opening network data to capture inspection, and potential corruption.

New DNS standards have fixed this bug by establishing discrete cache update security configuration that specify exactly which servers have the authority to provide updates.

To test whether a DNS server is vulnerable to DNS spoofing, queries can be sent to a given name server, assuming that its traffic will flow somewhere over your network link. You can then determine by analyzing the queries, whether or not it is possible to guess the next ID number of a query. If the IDs are predictable, you can assume that it is possible to poison the cache of the server with invalid data.

### **2.3 Java applet vulnerability**

There is a Java applet attack issue within domain name service. An applet on its own cannot carry out this attack. It would have been working in conjunction with a hacked DNS server owned and access to some domains. An applet did this as two ways; firstly, using DNS to translate the name of the Web server into a list of IP addresses; secondly, using DNS to translate the name of machine which the applet wants to connect to into a list of IP addresses. Compare the two lists to check if the addresses exist in both lists, if so, then the two machines are the same and allow the connection; if not, they are different and refuse the connection.

The attack applet would have to know the names and IP addresses of machines inside a corporate firewall, in order to try to establish connection to machines behind the firewall.

The following scenario describes that Java applet go wrong:

If a malicious attacker sets up a Web server called www.attacker.com, with IP address 180.14.14.14 and the attacked Web server called target.victim.com, with IP address 15.15.15.2.

If someone is surfing the Web on stooge.victim.com (IP address 15.15.15.1). It happens to visit the attacker s web site; The Web site contains a Java applet written by attacker. The applet is downloaded to stooge.vivtim.com and generate. The applet will create a network connection to bogus.attacker.com. Because the name is in the attacker.com domain, the attacker s DNS provides an IP address for that machine and is free to provide any IP address that it likes. The attacker s DNS returns the pair of address (15.15.15.2,180.14.14.14), because this list contain the address of the attacker s web server (180.14.14.14), Java conclude that www.attacker.com and bogus.attacker.com are really the same machine; therefore, it allows the connection to generate. After verifying the connection is allowed, Java connects to the first address on the list, target.victim.com. The attacker has achieved his or her aim, and to connect to the target machine.

The attacker can easily search the defenses of the target machine and looking for weaknesses. If the attacker can find any weaknesses, then the victim would be in a big trouble.

A fix for the applet security manager is to be stricter deciding which computers an applet is allowed to connect to. System needs to take note of the actual IP address that the applet truly came from, and thereafter only an applet to connect to that exact same numerical address.

These security problems describing above are within Windows NT as well as within UNIX DNS servers. There are newer versions of DNS and DHCP that take care of these security problems; however, these versions are only available for UNIX currently. Hopefully these concerns will be taken care of in Windows 2000 when it is launched.

### 3.0 Introduction to IP spoofing

The TCP/IP protocol suite is most widely used and can be considered as the basis for the Internet today. It comprised of two protocols, TCP and IP. This section presents a brief introduction of the two protocols and explains the meaning of IP spoofing.

The Transmission Control Protocol (TCP/IP) was designed in the early 1980 s with no security taken into consideration. Today, the TCP/IP protocol has become more of a problem because it lacks many features that are desirable or needed on an insecure network. TCP runs on top of IP and provide a connection-oriented service between the sender and receiver. It uses various mechanisms, such as sequence numbers, acknowledgements, 3-way handshakes and timers.

The Internet protocol (IP) is a network layer of the Internet. It provides a connectionless service by routing and sending packets to the packets destination. Many machines in the Internet rely on IP source addresses for authentication. In other words, machines on the Internet rely on network address based authentication and not application level authentication. That is, the target machine authenticates a session between itself and other machines by examining the IP source address of the requesting machine. It is therefore important to note that these IP source addresses provide identification, not authentication.

IP spoofing is a complex technical attack where an attacker sends out packets using an IP address other than its own. This type of attack is launched against a machine playing on the weaknesses of authentication systems. IP spoofing works because trusted services only rely on network address based authentication and no application level authentication. The following section describes some IP spoofing attacks and prevention that can be taken to protect against such attacks.

### **3.1 Man-in-the-middle**

A man-in-the-middle attack requires that the attacker with knowledge of messaging protocol and have access to network packets that come across the networks. An example of such a configuration could be someone who is working for the Internet Service provider (ISP), who can gain access to all network packets transferred between one network to any other network.

Such attacks are often implemented using network packet sniffer and routing/transport protocols. Packet sniffer provides information about the topology of your network. It is a useful tool used by network administrators to point out areas of weakness on the network. It not surprisingly that attackers find this useful.

The possible uses of such attacks are theft of information, hijacking of an ongoing session to gain access to an Internet network resources, traffic analysis to derive information about a network and its users, denial of services, corruption of transmitted data, and introduction of new information into network sessions.

### **3.2 Routing redirect**

Routing redirect is redirecting routing information from the original host to the hacker s host. This can be consider as another form of man-in-the-middle attack.

### **3.3 Source Routing**

Source routing is one variant of IP spoofing which is a rarely used IP option. In figure 1., the originating host specifies the path route that the receiver should use to reply to it.

Due to this, an attacker will by-passes the real host, and instead directs replies to a path it can monitor. To avoid this, routers can be configured to drop packets with source routing enabled.

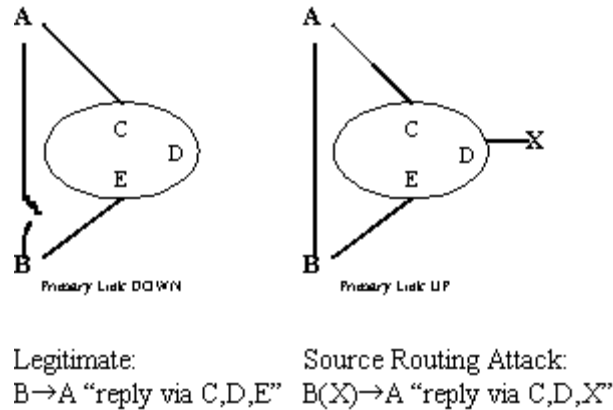


Figure 1. Source Routing[2]

### 3.4 Sequence Guessing

In TCP connections, the sequence number is a 32 bit number. Therefore, the odds of guessing the correct ISN are a rare chance. But it would become easy to guess if the ISN for a connection is assigned in a predictable way. BSD 4.2 is a Unix derivative used for exploiting predictable ISN. This means that the ISN for a connection is assigned from a global counter. This counter is incremented by 128 each second and, by 64 after each new connection. The attacker then can determine the current state of the system's counter by first establishing a real connection to the victim. So the attacker will know the next ISN to be assigned by the victim is quietly likely to be the predetermined ISN, plus 64. Furthermore, if the attacker sends a number of spoofed IP frames, he has an even higher chance of correctly guessing the ISN.

In spite of this, the attacker's connection will be abort, because it never started a connection (the host will indicate this by sending a reset command, called RST), so this

host will reject the SYN&ACK. Actually, when the host receiving spoofed packets completes its part of the three-way handshake, it will send a SYN&ACK to the spoofed host. However, the attacker may use the aforementioned SYN attack to swamp the host it is imitating. And the attacked host then sent the SYN&ACK along with any other packets sent while the host is flooded. Hence, the attacker has free reign to finish with his attack.

In order to avoid this, other operating systems may increment the ISN counter frequently. However, it doesn't do any help because an attacker may still be able to predict the ISN approximately even if the counter is incremented 250,000 times a second. The attacker will establish a connection with the correct ISN within a few hours by repeatedly guessing.

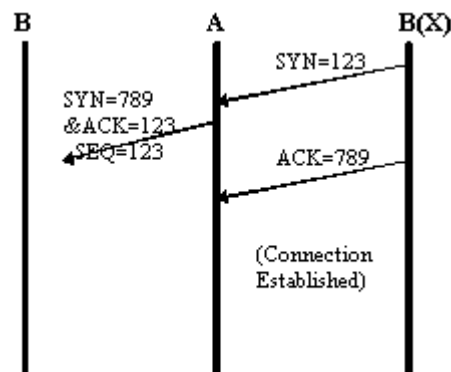


Figure 2. IP Spoofing via Sequence Guessing[2]

### 3.5 Flooding

When host B receives the SYN request from A, it keeps track of the partially opened connection in a listen queue for at least 75 seconds. Therefore, even with long network

delays, it will still be connected successfully. However, many implementations can only keep track of a very limited number of connections. A malicious host can exploit the small size of the listen queue by sending multiple SYN requests to a host, but never replying to the SYN&ACK the other host sends back. And the other host's listen queue is quickly filled up, it will stop accepting new connections, until a partially opened connection in the queue is completed or times out. The ability to effectively remove a host from the network for at least 75 seconds can be used as a denial-of-service attack, or can be used as a tool to implement other attacks, IP Spoofing for instance.

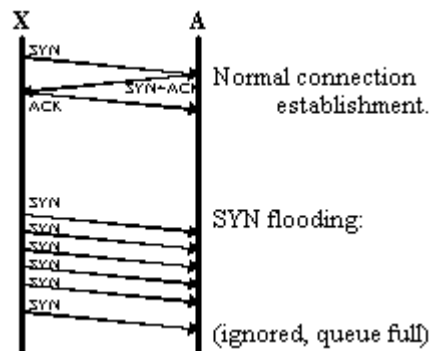


Figure 3. SYN Flooding [2]

### 3.6 What can be done to prevent IP Spoofing Attacks?

The likelihood of IP spoofing attacks can be reduced by configuring the network to reject packets from the Internet that claim to originate from the local address. This is commonly done with proper router configuration in a router. The routers work by applying filters on

incoming packets; for example, they can block particular types of packets from reaching a network. It is important to note that although routers are a solution to the general spoofing problem, they too operate by examining the source address. Thus, they can only protect against incoming packets that purport to originate from within a network. **If the network trusts foreign hosts**, routers will not protect against a spoofing attack that purports to originate from those hosts.

Even if you are running a firewall, this does not automatically protect you from spoofing attacks. If you allow internal addresses to access through the outside portion of the firewall, you are vulnerable to attacks too.

## 4.0 Conclusion

Different types of problems associated with DNS, DHCP and IP addresses have been discussed. All these problems are caused by the trust-relationship between one host and the other. IP spoofing arises due to its simple authentication based on the IP source address of an arriving packet from an external source. With the current IP protocol technology, it is impossible to eliminate IP-spoofed packets. However, steps can be taken to reduce the number of IP-spoofed packets entering and exiting the network as mentioned earlier using a filtering router. Another step to prevent IP spoofing is by using IPv6 or IPsec instead of IPv4, which include two new characteristics authentication header and encapsulated security payload.

This paper has also evaluated the different types of DNS and IP spoofing. Countermeasures are also listed to reduce and to prevent these types of attacks. It is not an easy task for a security administrator to look after and eliminates the loopholes in a network so as to prevent the network from penetration. Every network in the Internet is susceptible to attacks, if the hackers are able to discover just a weakness in the network.

Firewalls and routers are common network tools to guard against these attacks from the malicious hackers. These tools have to be properly configured and placed in a network in order to prevent any intrusions and attacks. Application-level firewalls should be used to filter off the unauthorised incoming IP packets and reliable DNS should be deployed to perform a thorough domain names and IP addresses checking and mapping.

How secure can an organisation networks be if all these countermeasures are taken since new viruses and attacks are created by hackers everyday?

**Reference Books:**

Stein, Lincoln D. Web Security, A Step-by-Step Reference Guide , Massachusetts, Addison Wesley Longman, Inc., 1997.

**Web sites:**

DNS Spoofing

[http://www.csis.gvsu.edu/class/cs437/Notes/Security/03\\_DNS\\_Attacks.html](http://www.csis.gvsu.edu/class/cs437/Notes/Security/03_DNS_Attacks.html)

<http://www.gordias.sk.sk/doc/itbmp/english/t/t578.htm>

Networking Basics — Domain Name Services

<http://www.ascend.se/3010.html>

DNS Spoofing and Java“

<http://java.sun.com/sfaq/dns.html>

Response to DNS-related security security problem

[http://ftp.javasoft.com/people/mrm/dns\\_spoofing.html](http://ftp.javasoft.com/people/mrm/dns_spoofing.html)

[1] Address Resolution Protocol

<http://www.erg.abdn.ac.uk/users/gorry/eg3561/inet-pages/arp.html>

[2] TCP/IP Security Chris Chambers, Justin Dolske, and Jayaraman Iyer

<http://www.cis.ohio-state.edu/~dolske/gradwork/cis694q>

Security Technology

[www.cisco.com/univered/cc/td/doc/cizintwk/ito\\_doc/security.htm](http://www.cisco.com/univered/cc/td/doc/cizintwk/ito_doc/security.htm)

IP spoofing

[www.ladysharrow.ndirect.co.uk/Maximum%20Security/spoofing\\_attacks.htm](http://www.ladysharrow.ndirect.co.uk/Maximum%20Security/spoofing_attacks.htm)

TCP/IP Security problems

[www.ja.net/CERT/Bellovin/TCP-IP\\_Security\\_Problems.html](http://www.ja.net/CERT/Bellovin/TCP-IP_Security_Problems.html)

Spoofing

[www.networkkice.com/advice/Underground/Hacking/Meth/default.html](http://www.networkkice.com/advice/Underground/Hacking/Meth/default.html)

Hello World - Spoofing

<http://www.helloworld.ca/1999/03-mar/spoofing.html>

<http://www.cert.org/advisories>

DNS Security

<http://apollo.mctr.umbc.edu/dnssec/>

DNS and Security

<http://www.apricot.net/apricot97/apII/Presentations/DNSandSecurity/index.htm>

DNS Spoofing and Windows NT DNS

[http://www.securiteam.com/windowsntfocus/DNS\\_Spoofing\\_and\\_Windows\\_NT\\_DNS.html](http://www.securiteam.com/windowsntfocus/DNS_Spoofing_and_Windows_NT_DNS.html)

Software firms issue new security warnings

[http://uk.internet.com/uk-news/article/0,1087,archive\\_101\\_51671,00.html](http://uk.internet.com/uk-news/article/0,1087,archive_101_51671,00.html)

Phrack Magazine Volume 8, Issue 52 January 26, 1998, article 03 of 20

<http://www.phrack.com/search.phtml?view&article=p52-3>