Global Information Assurance Certification Paper

GIAC Certified Incident Handler

Practical Assignment Version 4.0
Option Two

By

Ray Hawkins

March 9, 2005

**"Incident Handler Case File:
A New Twist to Social Engineering – The Threat to Publicly Traded Companies"**

1.0 Summary of Purpose

Each day information security professionals are confronted with an ever-changing landscape of new vulnerabilities as well as published exploits. The constant battle of patching and securing the network perimeter and critical internal systems will never diminish. Security professionals have become adept at designing and implementing complex controls at the network edge to mitigate the potential catastrophe of an exploit being launched at, and compromising mission-critical systems. In my experience of auditing security architectures and processes, the common theme is to design these controls such that the network edge is protected to a sufficient degree to allow for normalization while internal patching processes wind through testing and change management processes. Higher risk systems at the edge are frequently patched within a short time of critical patch release. Internal systems, critical or not, follow a predictable path of testing, validation, change management approval, and then install at the next available maintenance window. This practice, not uncommon, inevitably leaves a "soft underbelly" in the security blanket covering many organizations. There is no shortage of automated exploits, and exploit tools such as H.D. Moore's MetaSploit Framework. To a degree, these exploits and tools have made compromise idiot-proof. Companies with mature and evolved security departments design network edges with multiple firewalls, intrusion detection and prevention, reverse proxies, and file integrity checkers to limit with the MetaSploits of the world can do. There is however, one thing those devices cannot control – the human factor.

The purpose of this paper is to describe an actual attack against the security architecture of a publicly traded company. The exploit used does not involve reverse engineering, exploit code downloaded from K-Otik[1], or launching a pre-packaged executable from Metasploit[2]. The exploit is Social Engineering[3][4]. The vulnerability is people, or more specifically, poorly trained employees[5]. The most common form of Social Engineering nowadays is phishing. Phishing, put simply, is the practice of enticing unsuspecting victims to visit malicious websites so sensitive information such as credit card numbers can be harvested (amongst other methods)[6]. The form of Social Engineering described herein is

[1] http://www.k-otik.com/english
[2] http://www.metasploit.com
[3] http://www.us-cert.gov/cas/tips/ST04-014.html
[4] http://www.gartner.com/gc/webletter/security/issue1/article1.html
[5] http://www.securitydocs.com/library/2694
[6] http://www.webopedia.com/term/p/phishing.html

the old-fashioned form. A good story, backed with some credible information, is leveraged against a trusting victim to gain access to information systems. The details of the attack and subsequent incident response were performed as part of an engagement to test the control environment of a particular company. Two different teams were constructed to perform the work. One team would perform the attack; another team would perform the Incident Response. After a successful Social Engineering attack, the "Attack Team" would attempt to further exploit internal IT Systems. The "Attack Team" and "Incident Response Team" were unaware of the activities of each other. The Incident Response Team was engaged as though this were a normal incident response, not a planned event. Names and places have of course been changed. The dialogue used during the attack was recorded during the actual attack process. As described in the Exploit/Attack section, the method of Social Engineering used is of particular interest to publicly traded companies. Companies as such, due in part to regulatory requirements such as the Gramm-Leach-Bliley Act[7] and the Sarbanes-Oxley Act[8], are more likely to have better and more information security controls. The "soft-underbelly" is still there.

[7] http://www.ftc.gov/privacy/privacyinitiatives/safeguards.html
[8] http://www.entrust.com/governance/sox.htm

2.0 The Exploit

2.1 Attack Name

2.1.1 Social Engineering

Social Engineering can be described as the "'art and science of getting people to comply with your wishes', 'an outside hacker's use of psychological tricks on legitimate user's of a computer system, in order to obtain information he needs to gain access to the system', or 'getting needed information from a person rather than breaking into a system'"[9]. By this definition, social engineering attacks can take a number of different forms. While there is no definitive Common Vulnerabilities and Exposures ("CVE") listing for social engineering attacks, the CERT Coordination Center at Carnegie-Mellon has published a broad-based bulletin in regards to Social Engineering:

CERT Advisory CA-1991-04 Social Engineering
http://www.cert.org/advisories/CA-1991-04.html
Original publish date: April 18, 1991
Revised publish date: September 18, 1997

Additionally, other hybrid attacks with Social Engineering elements are occasionally published:

CERT Incident Note IN-2002-03 Social Engineering Attacks via IRC and Instant Messaging
http://www.cert.org/incident_notes/IN-2002-03.html

CERT Advisory CA-2003-08 Increased Activity Targeting Windows Shares
http://www.cert.org/advisories/CA-2003-08.html

CERT Advisory CA-2001-03 VBS/OnTheFly (Anna Kournikova) Malicious Code
http://www.cert.org/advisories/CA-2001-03.html

Secunia Security Advisory SA11828 Aspell word-list-compress Word List Processing Buffer Overflow Vulnerability
http://secunia.com/advisories/11828/

PSECU Security Article 02/25/05 CUNA Web Site Phished
http://www.psecu.com/About_Us/News/Security/2005/20050225.html

[9] http://www.securityfocus.com/infocus/1527

2.1.2 Samdump.dll Injection Via PWDump2

As further described in the Attack Process section, a second attack will be used once Social Engineering has been successful. Obtaining Microsoft Windows password hashes is easier said than done nowadays. With appropriate administrative access to a Windows Domain Controller (or of lesser value a member server) with SYSKEY installed, the password hashes from the NT Security Accounts Manager ("SAM") database can be obtained. SYSKEY provides 128-bit encryption to the database to safeguard against malicious users[10]. Prior to syskey, the vulnerability to the SAM database was that "Windows NT Server stores user account information, including a derivative of the user account password, in a secure portion of the Registry protected by access control and an obfuscation function. The account information in the Registry is only accessible to members of the Administrators group. Windows NT Server, like other operating systems, allows privileged users who are administrators access to all resources in the system"[10]. Using the pwdump2 tool, the syskey protection can be bypassed using a technique called "dll injection". This is done by having "one process (pwdump2.exe) force another process (lsass.exe) to load a DLL (samdump.dll) and execute some code from the DLL in the other process's (lsass.exe's) address space and user context"[11] thereby allowing the password hashes to be retrieved. While there is no specific CVE or CERT bulletin for this exploit, the pwdump2 exploit has been previously leveraged as part of a broader attack against Microsoft SQL Servers:

CERT Incident Note IN-2002-04 Exploitation of Vulnerabilities in Microsoft SQL Server
http://www.cert.org/incident_notes/IN-2002-04.html

2.2 Affected Operating Systems

2.2.1 Social Engineering

Social Engineering is not used as an isolated attack against specific operating system. It is generally used against people to obtain, among other things, unauthorized operating system access.

2.2.2 Samdump.dll Injection Via PWDump2

The use of PWDump2 is known to be successful against the following operating systems with Syskey installed:

---

[10] http://support.microsoft.com/kb/q143475/

[11] http://razor.bindview.com/Services/RAZOR/Utilities/Windows/pwdump2_readme.cfm?Print=1&

- Microsoft Windows NT 4.0 (Workstation and Server) – all service pack levels. Syskey is included by default in all service packs starting at "3".[10]

- Microsoft Windows 2000 (Professional and Server) – all service packs levels.[13]

PWDump2 as well as the newer pwdump3 have been used in other security suites to perform password dumps from Windows 2003 as well.[14]

2.3 Affected Protocols/Services/Applications

2.3.1 Social Engineering

Social Engineering does not specifically leverage a vulnerability in a protocol, service, or application; but rather is used to obtain unauthorized access to otherwise functioning and secure protocol, application, or service.

2.3.2 Samdump.dll Injection Via PWDump2

PWDump2 is operated by (preferably with console access) obtaining the process ID of the lsass.exe process. This can be obtained by starting up the Windows Task Manager, selecting the "processes" tab, identifying the lsass.exe process, and then recording the process ID ("PID") associated with it. Once the process ID has been obtained, the following commands from the home directory of pwdump2 will extract the password hash:

c:\pwdump2 pwdump2 256>c:\samdump.txt (where 256 is the LSASS PID)
(note: newer versions of pwdump2 "should" automatically collect the lsass pid – you may still have to do it manually)

As mentioned previously, pwdump2.exe is forcing lsass.exe to load the samdump.dll into its (lsass) memory space. What is happening here? In basic dll injection, a dll is injected into the memory space of an existing process. This is done using the SeDebugPrivilege (supporting why we want administrator console access – we'll cover this later). Once the code has been injected, it will execute under the same privilege as the running process. In the case of pwdump2 (specifically samdump.c), it is using the same Windows API that is used by the WinLogon process[15]. Winlogon uses the MsV1_0.dll to access

---

[10] http://support.microsoft.com/kb/q143475/

[13] http://www.bindview.com/Services/RAZOR/Utilities/Windows/pwdump2_readme.cfm

[14] http://www.crackpassword.com/products/prs/mswin/pwsex/

what is referred to as the "samr interface"[16]. The samr interface provides direct access to the Security Accounts Manager subsystem to perform all account operations (refer to Appendix B for additional details).

Pwdump2 uses the following samr functions:

- SamrConnect
- SamrOpenDomain
- SamrOpenUser
- SamrQueryInformationUser
- SamrEnumerateUsersInDomain

2.4 Exploit Description

2.4.1 Social Engineering

Social Engineering is exploitable because "of the natural human tendency to trust. Generally agreed upon as the weakest link in the security chain, the natural human willingness to accept someone at his or her word leaves many of us vulnerable to attack. Many experienced security experts emphasize this fact."[9] A limited number of professions employ people who are by default, suspicious or skeptical. Those professions may include auditors, law enforcement, and security professionals just to name a few. It is without question that humans are the weakest link in any security protection framework, be it logical or physical. Social Engineering as an exploit seeks to play on a person's innate willingness to trust another human being. The exploit is usually carried out in one of three methods: phone, online, and in person. For any method, the success of the exploit is directly proportional to how believable the "story" is, what supporting props are used, how much trust the unsuspecting victim gives to the attacker, and how much training employees are given to identify social engineering exploits.

2.4.2 Samdump.dll Injection Via PWDump2

As described previously, the Security Accounts Manager ("SAM") database on Microsoft Windows Servers is vulnerable to revealing the password hashes to an application or process with the SeDebugPrivilege. The SAM database holds

[15] http://phlak.freeunixhost.com/Microsoft/modifying-nt-credentials.txt

[16] http://www.hsc.fr/ressources/articles/win_net_srv/ch04s07s02.html

[9] http://www.securityfocus.com/infocus/1527

user accounts and passwords. With the advent of Windows NT 4.0 Service Pack 3, the syskey was applied to provide 128-bit encryption to the database. However, with or without syskey installed, a properly used utility such as pwdump2 can be used to extract the password hashes. The password hashes can then be run through a password cracking utility such as L0phtcrack[17]. Given sufficient time, tools as such can crack passwords that are not complex.

3.0 The Attack Process

3.1 Attack Introduction

In this incident, Consulting Company Delta ("Delta") was hired to test the control environment of a large, multi-location, multi-national company ("the victim"). The overall engagement was an unannounced internal and external penetration test and involved multiple teams performing different components of the test. The component of the test detailed here was the internal penetration test. The goal of the test was to obtain unauthorized physical and logical access to the internal computer network. The test was conducted at a location several time zones away from the corporate headquarters of the company ("Corporate"). The rules of engagement stipulated, that the testing to gain access could not involve physically breaking into the facility (e.g. smashing a window) or breaking an IT infrastructure component (e.g. cache-poisoning network switches). It did not go without saying that the engagement team could not impersonate law enforcement or government officials. Given these circumstances, the team began to devise ways of obtaining unauthorized access. The first hurdle was physical access. It was determined that the team would obtain this access one of two ways; either walking right into the facility, or developing a Social Engineering attack to gain access. Either option carried a high risk of getting caught. No prior knowledge of the security controls was provided to the team. In order to develop a successful Social Engineering attack, a dossier of any and all information on the company was compiled. The dossier included but was not limited to:

- A printout of all the pages of the company web site.
- A printout of the most recent annual reports of the company.
- A listing of company officials.
- A compilation of any newsgroup postings made by company employees.
- A listing of all company sites: address, time zone, and phone number blocks.
- A listing of all company owned Internet domains and IP addresses.
- A diagram of known organizational structure.

To fully build the dossier the attack team contacted various company employees posing as account executives from companies with whom the victim would likely

[17] http://www.atstake.com/products/lc/

have a business relationship with (IBM, Sun, Microsoft, Iron Mountain, Sungard). After initially speaking with operators, the team was forwarded to mid-level managers and/or staff level personnel to glean basic yet valuable information. The team also visited the victim site (located in an East Coast metropolitan area, but not in a densely populated area) to perform light reconnaissance of the area. These activities help add the following to the dossier:

- Times of the day when most workers arrive and leave (Arrival 7:45 a.m.-8:30 a.m., Departure 3:45 p.m.-5:00 p.m.)
- The approximate number of employees at the site (based on the number of occupied parking spaces on a mid-week work day – approximately 300).
- The presence, or lack, of security guards in or around the facility.
- A general idea of the building entry points (one main entrance, a side entrance used by smokers, and a rear building delivery garage door).
- A general idea of the access method to the building (most employees had badges secured to belts, purses, or necklaces).
- A general idea of the technology in use ("we haven't fully upgraded to Windows 2000 yet").
- An idea of the potential Social Engineering victims (middle-aged or older female receptionists).

In reviewing this information, the attack team identified and ranked several Social Engineering possibilities. In deciding which of the available "attacks" to use, the team elected to leverage the fact that the company is a publicly traded company. The annual report provided a wealth of information that ensure the success of this attack:

- As a public company, it is required to comply with the Sarbanes-Oxley Act ("SOA"). In a nutshell, this would require the company to perform detailed documentation and testing on their internal control environment (including IT controls).
- Every company is required to have an external public accounting firm review and sign-off on the financial statements. This same firm reviews the control environment – largely reviewing the internal work and performing additional control testing.

Since members of the team had worked with the Internal Audit functions at other companies, it was no secret that SOA compliance had created a lot of tension on the part of many company executives and thus various components of the organization such as Internal Audit and IT. This tension often extended to the relationship with the external auditor. Nothing could be worse for a company than for the external auditor to have to identify a material weakness in the control environment of the company. Satisfying the requirements of the external auditor would become paramount in year one of SOA, 2004 for the victim company. Thus the attack was born – the team would pose as members of the

company's external audit firm (easily identified from the annual report) to gain unauthorized access into the physical building and then the network. The "soft-underbelly" would be exposed.

Successful Social Engineering attacks must be well constructed, thought out, and tested. To ensure the success of the attack, the following steps were taken:

- Attack team members developed business cards with the company logo of the external audit firm. The phone number listed on each card was the mobile number (with caller ID) of one team member in a different time zone who could potentially verify the presence of the auditors.
- Each team member pulled together their most professional looking attire, polished shoes, etc. Presenting an image as an external auditor was important.
- Binders with the logos of the victim company and the external audit firm were put together. Innocuous documentation with company header information was placed in the binders.
- Additional probing phone calls to the victim company were made to glean any names from the external auditor who may be assigned to the victim company. This involved posing as a billing manager from the external auditor requesting the names of auditors who had worked onsite during the last quarter.
- An SOA story was constructed. The story had to brief and straightforward – "we're here for today only to perform some follow up testing for your SOA compliance – can you please escort us to a conference room with networking". The story was the crux of the attack. Knowing that the company was in the midst of a Sarbanes-Oxley project was critical to adding validity.
- A test of the story was performed. Attention to detail and body language was given extra scrutiny. Multiple scenarios were used with different personality types to gauge how the team would respond. The test was conducted in a room unfamiliar to the team, and they were quizzed on detail recollection of the room at the conclusion of the test.

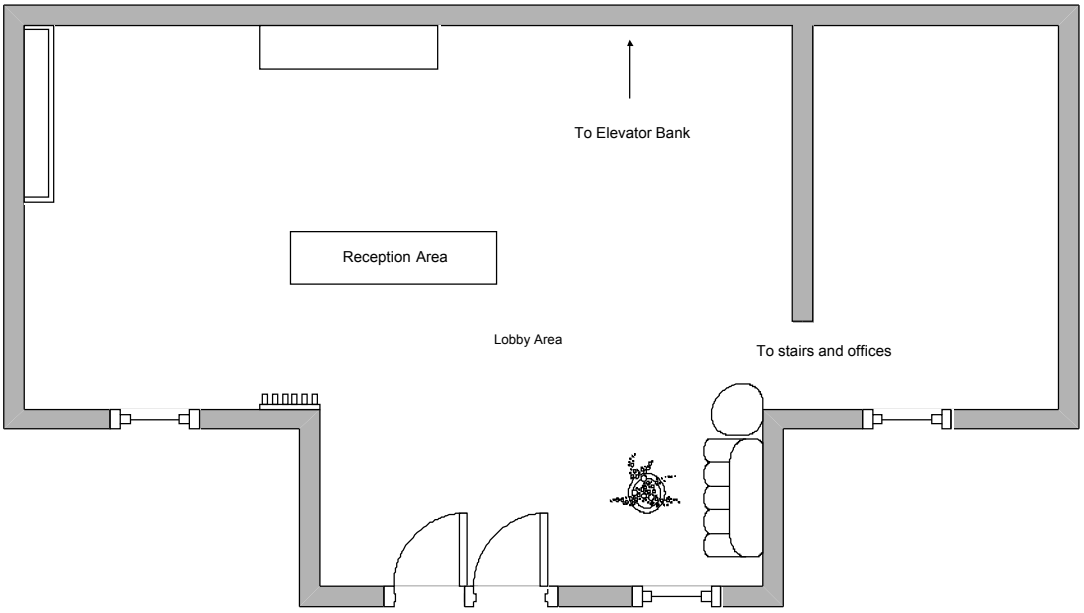
3.2 Attack Execution

1. Attack Part One - The attack team recognized that several unknowns would have to be contended with. It was important for the attack to be flexible enough so that depending on whom the team encountered within the company, the team could adjust accordingly to gain access. Armed with sufficient planning and testing to execute the attack, the team visited the company site on a Monday morning, arriving at the same time as most of the employees. The attack unfolded as follows:

- Two members of the attack team drove in together and parked in a visitor parking space so as not to draw attention (by taking an employee spot).

- Carrying a binder each with attacker and victim company logos prominently displayed, the team entered through the front door.
- The team immediately noticed that there was no guard posted. Employees displayed their badges on their person; however, there was no physical access mechanism for the badges. The entrance foyer was located in the front center of the building with a wing to each side and an elevator bank directly behind the reception area.
- There were two receptionists on duty. Both seemed preoccupied with discussing their weekend activities, sipping coffee, answering the phones, and waving to employees as they entered.
- The team approached the reception area and initiated the greeting to the receptionist:

Figure 1 Lobby Area:



- Team Lead – "Hi, I'm Jerome Garcia and this is Mickey Lesh. We're from your external audit firm Touche Young, and Price. We flew in from our office in San Francisco last night and are here to see Janis Slick from Internal Audit. (Janis is a staff level auditor).
- Friendly Receptionist Donna – "Ok, she just came in a few moments ago, let me call her. Is she expecting you?"
- Jerome – "Probably not. We just received our assignment from San Fran on Friday to come here today."
- Donna – "On Friday? Wow – you must be tired. (calls Janis) Janis this Donna downstairs, I have two gentlemen here from TYP from San Francisco here to see you."
- (Janis comes to the reception area) Janis – "I wasn't expecting

you. Who sent you?"

- Jerome – "Brent Godchaux from San Fran. Here is his business card. The local TYP partner had Brent send us to do some of the follow-up testing for SOA. We're only here for today."
- Janis – "Well – I need to call Bruce from the local office then, because I knew nothing about this. We're not even ready for you guys."
- Jerome – "Bruce isn't in town today – I'm pretty sure he's at the IIA Conference in Las Vegas. Brent said to give him a call if there were any issues."
- Janis – "Well – I'll probably do that later. I don't really have time for this. What do you need from me?"
- Jerome – "Very simple. Like I said, we're only here today. If you could set us up in a conference room with some network connections – we already have network IDs from corporate. Also, we were told to meet with the local network admin – I can't remember his name – we have to test some of the account provisioning controls for network access."
- Janis – "You mean Bob Constanten? He can help you with the testing – I thought they were doing all that at corporate?"
- Jerome – "For the most part yes. We just have to ensure there are no differences in the controls between you guys and corporate. Any chance we can get a copy of your documentation and test plans? That'll make the testing go much easier."
- Janis – "I guess. When is Brent available? I just want to confirm all this."
- Jerome – "He's an earlier riser. He is still on the West Coast so just give him an hour or two. He runs in the morning. Also, if you're available, we'll take you to lunch later on. That way we can discuss any prelim findings."
- Janis – "Now you're talking. Lets head upstairs. You can use the IA conference room for now. I'll get Bob to come up there as well."

With that, the team signed the guest book, Internal Audit signed the guestbook, and the attack was successful. The team was ushered into the building, into a conference room, and proceeded to hook up their laptops for additional testing. Given the relative size of the building and number of parking spaces, the team estimated that no more than 300 employees worked this location. In observing areas of the building as they were escorted, it appeared that several standard corporate departments such as HR, Accounting, Marketing, and Internal Audit were onsite, albeit with limited staff. Building on this, the team reasoned that IT operations were most likely onsite rather than a co-location facility offsite. There were probably a limited number of servers such as file/print, DNS, directory services (e.g. domain controller), probably some ERP, and some applications servers. There most likely would not be any web servers or mid-

range/mainframe type servers.

2. Attack Part Two - Once the team was ushered into the conference room, the IT Network Administrator was summoned. The team did not expect to meet with this person, perhaps even at all. The key to a successful Social Engineering attack is flexibility in approach. The team disconnected the laptops, opened the binders, spread out some documentation, and generally tried to look busy preparing to perform audit work. Once the IT Network Administrator entered, the team observed that he was somewhat younger than expected, appeared nervous and rushed. They sensed an opportunity to extend the Social Engineering attack:

- Jerome – "Good morning, you must be Bob. I'm Jerome and this is Mickey. Did Janis explain anything to you?"
- Bob – "Um, she said you guys were from TYP and had to do some more testing for SOA. I thought we were done with that."
- Jerome – "That is correct. We just have to do some refresh testing. You may not have heard but your corporate was nailed on some access control issues. We need to test to ensure that the recommendations were implemented. You have access on the local domain controller right?"
- Bob – "Yes – but if something was changed on network accounts it would propagate to our servers. That all comes from corporate."
- Jerome – "I understand. They sent us here to test because corporate is so tied up with the process re-testing with Finance. You don't even want to know what a mess that is. The sooner we get this done the sooner we'll be out of your hair."
- Bob – "Um, ok. What do I need to do?"
- Jerome – "Janis is providing us some documentation on your local testing efforts. All we need to do is sample the account controls from your domain controller and verify that the computer room is secure."
- Bob – "I can get you some screen shots. That's what we've done before for Internal Audit."
- Jerome – "Right. We actually need to see the controls directly on the box. We've had some clients make changes to the system and then do screen shots. Doing it this way we can evidence that you haven't changed anything on the fly. We test it, document the results, and then you never see us again."
- Bob – "Ok. I have meetings from 9-11 this morning. Do you want to do it now or later?"
- Jerome – "If we can do it now that would be great."

The IT Network Administrator, eager to get the ordeal over with, took the team downstairs to the computer room. The room was smallish, not really designed to be a computer room. It was very well secured and contained approximately

15-20 servers. There was a mix of old and new servers, some tower servers, some rack-mount. The room was not staffed, but a door at the back of the computer room appeared to lead to some offices. The conclusion of the Social Engineering attack was about to take place:

- Jerome – "Which one is your domain controller?"
- Bob – "This Compaq ML530 over here. It's the only we have."
- Jerome – "You should say something to corporate. Don't you think you should have more than one?"
- Bob – "I complain about that all the time. They always say there isn't enough money – something about 'we're not a high enough priority site to have more than one domain controller'. Our link back to them is god-awful slow. If we lose this one people here will scream because it'll take forever to login."
- Jerome – "I feel your pain. Listen, if you want to log onto the machine we'll get started. I'll do the actual screen shots, print them out, and have you sign them. You guys have a cafeteria here right?"
- Bob – "Uh – yeah."
- Jerome – "Tell you what, if you can show Mickey where it is we'll treat to coffee since we're hassling you on a Monday morning. By the time you get back I should be done."
- Bob – "I'm not supposed to leave you guys alone in here."
- Jerome – "Its ok. We have clearance and IDs. Janis already cleared it with corporate."

With this, the attack was wrapping up. As soon as the IT Network Administrator left with Mickey, Jerome pulled a USB flash drive from his pocket and inserted it into the domain controller. As part of the prior Social Engineering, it was noted that this site was still using Windows NT Server 4.0 (See Network Diagram – Appendix C). The team made an assumption that the domain controller was running the most recent service pack version for that operating system: Service Pack 6a. On the USB drive was one file: pwdump2. Pwdump2 is a utility used to dump password hashes from the Microsoft Windows NT Security Accounts Manager (SAM) database. The utility "uses a technique known as DLL injection. One process (pwdump2.exe) forces another process (lsass.exe) to load a DLL (samdump.dll) and execute some code from the DLL in the other process's address space and user context."[13] Using the Windows Task Manager to identify the LSASS Process ID (pid), the following command was executed directly on the server:

c:\pwdump2 pwdump2 256>c:\samdump.txt (where 256 is the LSASS pid)

This process took all of about 2 minutes. The samdump.txt file was copied to

[13] http://www.bindview.com/Support/RAZOR/Utilities/Windows/pwdump2_readme.cfm

the USB drive. This file could then be loaded onto a different computer whereby the passwords would be cracked using a tool such as L0phtcrack[17]. He then pulled some screen shots of the account policy so the story presented to Bob would play out correctly. As Jerome figured, since the company had multiple sites, each physical site was most likely a separate Microsoft domain. The domain would thus have mostly accounts for the local employees to login to the network, plus some accounts for the corporate IT Network Administrators. With the screen shots of the actual account policy demonstrating that the company had fairly strong password controls the team realized the password cracking process may took a couple of days. The team spent a couple of hours building fake SOA testing documentation, took Janis and Bob to lunch where they were praised for having such an effective control environment, then left the premises in the afternoon having successfully executed a Social Engineering attack.

The success of the attack was dependent on several key factors. Absent one or more factors, the attack may not have been such a success.

- The target site was specifically selected due to size, location, and the likelihood the controls over human behavior would not be as prevalent. The team did not feel the attack would be successful at a larger site or corporate headquarters.
- The Team developed a concise "story", practiced it, and developed supporting tools to make the story more believable (business cards, binders, a supporting resource to support the story).
- The Team played upon and to, both the fears and motivations of the victims they encountered. Social Engineering depends greatly on the ability of the attacker to recognize and interpret verbal cues given by the victim(s). Different personality types must be recognized and handled accordingly[18].
- The Team specifically selected a day and time (time of day, time of audit activity) that help ensure the success of the attack.
- Lastly, the team leveraged the idea that even though corporate headquarters may have an excellent training program to educate employees against Social Engineering attacks, distant sites from the headquarters are more likely follow local custom of operations – meaning, these organizations "follow" corporate policy, but often interpret and apply policy a little differently.

[17] http://www.atstake.com/products/lc/

[18] http://www.securityfocus.com/guest/5044

3.3 Signatures of the Attack

3.3.1 Social Engineering

Signatures of the Social Engineering Attack included the following:

- Unexpected phone calls or visits from a person(s) seeking benign and/or sensitive information.
- Unexpected visitors proclaiming undocumented official business.
- Overly friendly or forceful personalities attempting to acquire information or access.
- Anomalies in guest book signings.
- Anomalies or oddities in documentation presented to employees. This may be company reports, business cards, name badges, etc.
- Successful exploit of information systems wherein there is a highly mature and evolved information security program and supporting architecture. While the remote site may not have specific security management maturity, sufficient security controls were in place to otherwise thwart an attack. This is addressed is section 3.3.2.

In this particular case, there were several measures that could have prevented the social engineering attack:

- Person(s) claiming some measure of authority (external auditor) should be immediately verified. Verification should be done with known and established contacts.

- Pre-social engineering that was done over the phone should be met with skepticism. Any inquiries into internal business practices and/or installed hardware/software should again be verified against known and established contacts. Every vendor has established contacts, and generally speaking, changes in contacts (especially a seller-buyer relationship) are updated when there is a change.

- Clear relationship guidelines should be developed when dealing with external auditors. This can include definition of project teams, prior phone contact with visiting auditors, and setting expectations on what auditors will bring with them (laptops, binders, etc).

- The presentation of identification should always be scrutinized. There is

no limit to what can be forged (business cards, ID badges, Driver's License).

- Conduct training with all personnel as to the methods and danger of social engineering.

3.3.2 Samdump.dll Injection Via PWDump2

The exercise of this vulnerability did extend to cracking passwords and compromising accounts, however, signatures of a samdump.dll injection are identified in the following manner:

- Attackers may leave the source files directly on the victim machine in a directory or in the recycle bin.
- If the server is not sufficiently sized, there may be a slight performance drop while the utility is being run. This would require a solid performance baseline and tools or utilities to identify performance drops.
- The most recognized signature is the compromise of user accounts, usually accounts with elevated privilege. Assume that if 2 or more accounts have been compromised, then the entire SAM database may have been compromised.

There are effective ways to prevent and/or mitigate this type of attack. These approaches would include:

- Limit the number of, and ensure authorization of any person(s) or process with administrator level access on Microsoft Windows Server platforms. This would include any person or process with the SeDebugPrivilege on the server.

- Never, under any circumstance, permit an auditor to have access to administrative level applications or process unless directed by a Court order. Any internal requests for such access should originate with corporate management and should be documented. Cases as such should require monitoring by internal personnel.

- Audit all access performed by administrative accounts. These audits should be reviewed frequently be system administrators and periodically by management.

- Establish non-administrative logon IDs for system administrators to use when not performing administrative tasks. This limits an attacker's ability if a workstation is left "logged-in".

4.0 The Incident Handling Process

4.1 *Preparation*

As mentioned previously the target company had retained Consulting Company Delta for Incident Response services. These Incident Response services partnered with an ad hoc team located at the company headquarters. The internal team was compromised of core technology professionals charged with identifying and proclaiming an incident. Incident Response was not considered a primary role or responsibility for any of the members. The team included tertiary members located at each company location, including the location attacked. These onsite members provided identification and support services only. Incident Response (IR) procedures for employees were included as part of orientation and an IR Response Line (phone number) was set up for employees to report any security incident.

The Delta Team was comprised of IR Professionals geographically dispersed to provide appropriate response time to an incident. Each IR Team member maintained an IR kit. The kit contained a variety of items such as:

- A dedicated dual boot laptop (Windows XP/Fedora)
- Basic Computer tools (screwdrivers, cabling, etc.)
- New hard drive disks for imaging
- Evidence bags
- Camera
- Gloves
- Interview template documents
- Tape recorder
- Blank CD-ROMs
- Blank floppy diskettes
- USB Drive
- Wireless Card
- Hub
- EnCase Forensics Software
- Dedicated credit card for ad hoc purchases

Each team member was trained in Delta guidelines to Incident Response and was tested on a periodic basis to ensure readiness for a response.

The victim company had deployed complex security appliances throughout the network perimeter, including monitoring devices on the connections from

locations coming into the corporate headquarters. Warning banners were placed on all systems, applications, and infrastructure devices; as well as at the entrance to the facilities and higher security areas. The security architecture has successfully detected malicious activity in the past, varying between external and internal sources. Malicious activity had never been detected from the victim site tested in this engagement. Corporate headquarters had sent out security warnings in the past in regards to phishing attacks via electronic mail. The emails generally described social engineering as an attack method, and provided employees with email subject lines to be leery of. Specific implemented countermeasures included:

- Symantec Enterprise Firewall 7.0
- TrendMicro Antivirus – Corporate Edition
- Internet Security Systems RealSecure Network Intrusion Detection
- Tripwire for Servers
- Symantec Intruder Alert Host-based Intrusion Detection

As stated previously, the Incident Response specific to this attack was a component of the overall engagement. The Incident Response activity operated independent of, and uninformed of the team performing the attack. The goal of this approach was to gauge the speed and accuracy with which an Incident could be identified and responded to. Consideration was given to the point at which Delta was engaged.

4.2 *Identification*

The initial report of an issue was raised during a project status meeting for the Company's Sarbanes-Oxley project team. The team was reviewing the status of outstanding items to be completed during the engagement. During the meeting, Janis noticed that IT Controls re-testing to be performed by the external auditor had not been marked as complete. She informed the project team that this should be updated to reflect the work that was recently performed. The project team disagreed as the external audit team had not even scheduled a visit to perform the retesting. Janis and her manager agreed to take the conversation offline.

During discussion with her manager, Janis insisted that the IT testing had been completed several weeks prior. She supplied the audit report that had been left with her by the Delta Attack Team as well the business card given to her. The manager reviewed the document and immediately sensed that something was wrong. The referenced managing partner from the external audit firm on the audit report was not the assigned person to their facility. A call was placed to

the local external audit office. After speaking with the managing partner, it was determined that auditors that were onsite were not employees of the external audit firm. An incident was born.

The manager contacted the Incident Response hotline to file an incident. The corporate staffer indicated that no malicious activity had been detected originating from the East Coast site. They were to refer the matter to the Delta IR Team for further investigation.

Upon contact from the corporate Incident Response, we recognized that more and better information was needed before this issue could or should be identified as an incident. We contacted the East Coast Site and spoke with Janis and her manager via conference call. We could establish only the following facts:

- A conflict between expected and actual documentation existed for their Sarbanes-Oxley project.
- Two individuals who, although they claimed to be employees of the external audit firm, could not be confirmed as employees supplied the documentation.
- The two individuals had been given physical access to the building, access to network connection ports within a conference room, and access to the local computer room.

These facts qualified the issue as an incident. We would be required to visit the site and determine the extent of damage, if any, had been caused by the two individuals. Once onsite, we conducted thorough interviews with any personnel who may have come in contact with the two individuals. We spoke with the receptionist on duty that day, Janis and Bob. The interviews were documented and recorded. From this process, we obtained these additional facts:

- Both individuals had signed the guest book with obviously fictitious names: “Jerome Garcia” and “Mickey Lesh”.
- According the internal DHCP Server, if leases were issued to the two, the leases had already expired.
- One person had been left alone at the console of the domain controller that was logged in with the Administrator account.
- A forged business card had been handed over. The phone number listed was a West Coast number.
- Both the receptionist, Janis, and Bob had been Socially Engineered into granting physical and logical access to the company resources.

We elected at this point not to call the phone number listed. We wanted to ensure that we identified if anything had been compromised. Calling the intruder(s) may alert them and increase the difficulty in catching them. We had Bob walk us through precisely what the conditions were in the computer room at

the time they were in there. The only machine logged in at the time was the domain controller. We secured the computer room, donned our gloves, and began to examine the domain controller. Before shutting down the domain controller for further analysis, we reviewed the following:

- The Domain Controller logs did not indicate any change in permissions at the time of the incident.
- The local antivirus server indicated that as of that date, all system were running the most current version of program code and pattern files.
- The firewall logs did not indicate any denied traffic sourced from the domain controller.
- IDS appliances did not indicate any malicious activity sourced from the domain controller.
- There were no temp files and trash files indicating malicious activity on the machine.
- The machine was last rebooted prior to the incident.
- The Task Manager did not indicate any unauthorized processes.
- The program files directory and the registry hives did not indicate any unauthorized programs.

Puzzled as to why someone would go to great lengths using a Social Engineering attack to access a domain controller, and then not do anything, we literally and figuratively took a step back. Was this a prank? Would we need to remove the computer and perform forensics? While I began to ask questions regarding the hardening process for this server, another Delta Team member began inspecting the room, looking for anything that may be out of place. I was able to determine that, to date, the process for hardening the server was to patch, patch, patch. Common guidelines such as NIST had not been implemented. This indicated to me that while patched, the server was susceptible to other forms of compromise – compromise that would require physical access to the machine. My colleague additionally observed that while the company appeared very diligent regarding cable management, there were two cable ties pulled apart at the back of the domain controller. We inquired as to whether maintenance had been performed on or around the machine. Bob indicated no. The machine hadn't been touched since it was installed.

For the Incident Response Team this was a red flag. Open and available serial and USB ports were located at the rear of the machine where the cable ties were pulled apart. We photographed the rear of the machine. Since we were not authorized to perform forensics on the machine until after business hours we elected to roll the dice, and call the number provided on the business card. I instructed a team member from the Midwest to use his mobile phone rather than calling directly from the company phone. As soon as the call connected it was evident as to what had really happened. The Attack Team member from the West was instructed to reveal the attack basics to us if and when we called. A letter was then faxed to the East Coast site to our attention from the corporate

headquarters enumerating in detail the attack process. We then conference called the project sponsor, the corporate Chief Financial Officer and the Attack Team to confirm the details and next steps. For purposes of the exercise we would make one additional assumption:

- We would assume that the SAM database had been fully compromised, the passwords for all users in the domain had been cracked, and all network accounts were available for compromise.

4.3 *Containment*

While the containment phase of Incident Response process typically involves disconnecting systems, applications, and/or networks; the situation presented a different set of circumstances and challenges. The fact that the Social Attack was successful, in and of itself, would not lead our Incident Response team to physically lock down the facility to "contain" the affected entity. The internal sponsor of the project, the corporate Chief Financial Officer (CFO) was understandably distressed that security had so easily been compromised. Based on that reaction, we elected to perform containment under the guise that our own colleagues had not executed the attack as part of a sponsored engagement. Our approach to containment in this situation was to immediately address the internal corporate resources that were compromised; exercising caution not to reveal all of the details of the engagement less we diminish to criticality of what happened. In conjunction with the local Human Resources representative, we spoke to each person individually and reviewed existing policy and procedures regarding visitor access to the facility. Furthermore, as a lead into our eradication of this vulnerability, we detailed how the Attack Team was able to exploit each of them to gain access to company resources. Care was given to help each person understand how and why they were selected, and how the Attack Team leveraged the following key actions against them:

- Receptionist - The receptionist were ready and willing to assist the Attack Team. That was the nature of their personalities. As a service oriented company, the employees should still treat each person as a customer or potential customer. Any non-employee must be treated the same when it comes to access to the building, or access to IT resources. Each and every access must have prior approval from management, and must be listed on the daily visitor list. No exceptions.
- Internal Audit and IT - Any unplanned visit is an unauthorized visit. That includes external auditors. An external auditor would rather be vetted through a security approval process than be given unchallenged access to the building and IT resources.
- Other – All employees were shown questioning and modeling techniques to help them identify when someone is attempting to take advantage of them.

As for the Primary Domain Controller SAM database that had been compromised, and working under the assumption that all account passwords had been compromised, we would have to enact more severe containment measures. The compromise of network accounts poses an immediate and critical risk to the IT Assets and Intellectual Property of the victim corporation (in this case – a single site of the company). In a case as such the following containment measures are required and immediate:

- To prevent any immediate unauthorized access to the network, all non-employees must be removed from areas where network connectivity is provided.
- All employees must be required to save existing work and logoff the network.
- To prevent immediate and unauthorized access to the network, remote access services (if any) must be immediately disabled.
- To ensure that no additional malicious activity was engaged on the target Primary Domain Controller, the machine must be removed from the network.
- To provision a return to normal operations, the Backup Domain Controller is promoted to become a Primary Domain Controller. The Domain Administrator password is now changed.
- Network Account Policy must be set to require all users to change their passwords at the next login.
- Employees are engaged to login into the network and immediately change their password.

4.4 *Eradication*

Eradication of the vulnerability attacked here (poorly-trained employees) would involve processes and procedures far beyond simply removing malicious code or re-imaging a compromised system. Since the vulnerability was human, the Incident Response Team had to develop realistic solutions to eradicate the vulnerability to safeguard against reoccurrence. After discussion with Corporate Management, the following eradication measures were proposed and agreed upon:

- Review, update, and communicate Information Security Policies and Procedures to include sections on identifying Social Engineering attacks.
- Establish and conduct Information Security Training at all Corporate levels focusing specifically on Policies and Procedures.
- Establish testing procedures to ensure employees understand Information Security practices and principles.
- Distribute periodic security reminders that include reward-based incentives for employees to review and test out on the content provided.
- Leverage existing key card access systems protecting sensitive areas to

public access areas where appropriate. Remove the responsibility of physical security from receptionist personnel.
- Where extending key card access on front access doors is not possible due to public access requirements, implement security measures at non-public access areas such as elevator banks and stairwells.
- Periodically inspect adherence to Information Security Policies and Procedures through audits or additional penetration tests.
- Educate key business partners, contractors, and consultants as to the changes to physical and logical access. Where possible, amend existing contracts to include termination language for violations of updated security policies and procedures.

Additionally, and working under the assumption that all the network account password were compromised, there are several potential eradication steps to engage:

- Establish appropriate steps in secure data from the offline former PDC. This would include using direct-attached backup devices to capture any user data files that may have been stored on the server and save to tape media.
- To provision additional investigative measures, the compromised machine could be examined in a secure networked environment to ensure no additional malicious activity was performed on the machine.
- While not applicable in this engagement, wherein criminal activity is suspected, the first two steps should be not be engaged until forensic examination of the machine and hard drives can be finished. If forensics is performed, two copies of the drives must be obtained. One copy is used for the actual forensics work, while the second copy is saved as a pristine copy – locked away in a safe and untouched unless the primary copy is corrupted or damaged. If multiple forensics teams are used, and thus multiple copies made, at least one pristine copy must be saved to ensure integrity of the disk state.
- Implement a true "lights-out" operation in the data center such that personnel are not normally staffed or working in the data center. Employees requiring access must have prior management authorization to enter.
- Standardize maintenance windows were the Administrator account would be required for use. Configure IDS systems to alert on any Administrator use outside that window.
- Consider changing the Administrator account name as well as any accounts with elevated access on the network.
- Consider implementing additional hardening procedures to disable unnecessary hardware devices such as CD-ROM, Floppy, USB, etc.
- Install motion activated security cameras to record data center activity.

This works best if the data center is “lights out”.

4.5 *Recovery*

Due to the severity of the compromise, recovery was partially initiated in the containment phase. There are activities to be performed to verify that full recovery has been achieved:

- Periodic review if guest sign-in logs to match visitors present against actual sign-in.
- Review of system and security logs to ensure that any changes to Information Security Policy and Procedures are being followed. If the policy was changed to restrict console use of the Administrator account, then this should be evidenced in the log files.
- Review of remote access and server security logs to identify a preponderance of failed login attempts (indicative of someone trying to use the previously compromised password).
- Identification of spikes in Network account lockouts post-incident to identify if the previously compromised passwords are being attempted.

Additionally, we re-visited the same site with a different set of personnel to evaluate the effectiveness with which additional controls were implemented. A key factor in ensuring recovery from the attack was to informally test a sample of employees on updated Information Security Policies and Procedures.

4.6 *Lessons Learned*

After the implementation of additional security controls and informal re-testing, a full report was issued to Corporate Management. Critical content in the report included re-interviewing the attacked personnel to determine what additional mitigating controls could have been in place to prevent another such attack. Experience has taught that many employees view an exercise as such as waste of time, money, and effort – “we’re overworked and underpaid already!” In the discussions we identified a key lesson learned – many employees were aware to varying degrees as to what Social Engineering is. They were not aware that an attack as such does need email as a delivery vehicle. The employees were not prepared to deal with the interaction dynamics that occurred during the attack. Specifically, employees felt that a short lesson or class on interpersonal communication skills would help them better deal with a situation in the future. Based on this feedback, we made the additional recommendation that such training be provided. The most important lesson learned - No one expects to be socially engineered in a face-to-face situation.

Appendix A

Key Exploit References

Avoiding Social Engineering and Phishing Attacks
http://www.us-cert.gov/cas/tips/ST04-014.html

Unmasking Social Engineering Attacks
http://www.gartner.com/gc/webletter/security/issue1/article1.html

CERT Advisory CA-1991-04 Social Engineering
http://www.cert.org/advisories/CA-1991-04.html
Original publish date: April 18, 1991
Revised publish date: September 18, 1997

CERT Incident Note IN-2002-03 Social Engineering Attacks via IRC and Instant Messaging
http://www.cert.org/incident_notes/IN-2002-03.html

CERT Advisory CA-2003-08 Increased Activity Targeting Windows Shares
http://www.cert.org/advisories/CA-2003-08.html

CERT Advisory CA-2001-03 VBS/OnTheFly (Anna Kournikova) Malicious Code
http://www.cert.org/advisories/CA-2001-03.html

Secunia Security Advisory SA11828 Aspell word-list-compress Word List Processing Buffer Overflow Vulnerability
http://secunia.com/advisories/11828/

PSECU Security Article 02/25/05 CUNA Web Site Phished
http://www.psecu.com/About_Us/News/Security/2005/20050225.html

CERT Incident Note IN-2002-04 Exploitation of Vulnerabilities in Microsoft SQL Server
http://www.cert.org/incident_notes/IN-2002-04.html

References

Exploits and PoC

http://www.k-otik.com/english/

Metasploit Project
http://www.metasploit.com

The need for Security Testing An Introduction to OSSTMM 3.0
http://www.securitydocs.com/library/2694

phishing
http://www.webopedia.com/term/p/phishing.html

The Gramm-Leach-Bliley Act: The Safeguards Rule
http://www.ftc.gov/privacy/privacyinitiatives/safeguards.html

Information Security Governance: The Sarbanes-Oxley Act and the Impacts on Non-Compliance
http://www.entrust.com/governance/sox.htm

Social Engineering Fundamentals
http://www.securityfocus.com/infocus/1527

Windows NT System Key Permits Strong Encryption of the SAM
http://support.microsoft.com/kb/q143475/

Proactive Windows Security Explorer
http://www.crackpassword.com/products/prs/mswin/pwsex/

Modifying Windows NT Logon Credential
http://phlak.freeunixhost.com/Microsoft/modifying-nt-credentials.txt

samr interface
http://www.hsc.fr/ressources/articles/win_net_srv/ch04s07s02.html

PWDUMP2
http://www.bindview.com/Support/RAZOR/Utilities/Windows/pwdump2_readme.cfm

@stake LC5
http://www.atstake.com/products/lc/

NLP-powered Social Engineering
http://www.securityfocus.com/guest/5044

Appendix B

SAMR Operations

Source: http://www.hsc.fr/ressources/articles/win_net_srv/ch04s07s02.html

Interface	Operation number	Operation name
12345778-1234-abcd-ef00-0123456789ac v1.0: samr		
	0x00	SamrConnect
	0x01	SamrCloseHandle
	0x02	SamrSetSecurityObject
	0x03	SamrQuerySecurityObject
	0x04	SamrShutdownSamServer
	0x05	SamrLookupDomainInSamServer
	0x06	SamrEnumerateDomainsInSamServer
	0x07	SamrOpenDomain
	0x08	SamrQueryInformationDomain
	0x09	SamrSetInformationDomain
	0x0a	SamrCreateGroupInDomain
	0x0b	SamrEnumerateGroupsInDomain
	0x0c	SamrCreateUserInDomain
	0x0d	SamrEnumerateUsersInDomain
	0x0e	SamrCreateAliasInDomain
	0x0f	SamrEnumerateAliasesInDomain
	0x10	SamrGetAliasMembership
	0x11	SamrLookupNamesInDomain
	0x12	SamrLookupIdsInDomain
	0x13	SamrOpenGroup
	0x14	SamrQueryInformationGroup
	0x15	SamrSetInformationGroup
	0x16	SamrAddMemberToGroup
	0x17	SamrDeleteGroup
	0x18	SamrRemoveMemberFromGroup
	0x19	SamrGetMembersInGroup
	0x1a	SamrSetMemberAttributesOfGroup
	0x1b	SamrOpenAlias

	0x1c	SamrQueryInformationAlias
	0x1d	SamrSetInformationAlias
	0x1e	SamrDeleteAlias
	0x1f	SamrAddMemberToAlias
	0x20	SamrRemoveMemberFromAlias
	0x21	SamrGetMembersInAlias
	0x22	SamrOpenUser
	0x23	SamrDeleteUser
	0x24	SamrQueryInformationUser
	0x25	SamrSetInformationUser
	0x26	SamrChangePasswordUser
	0x27	SamrGetGroupsForUser
	0x28	SamrQueryDisplayInformation
	0x29	SamrGetDisplayEnumerationIndex
	0x2a	SamrTestPrivateFunctionsDomain
	0x2b	SamrTestPrivateFunctionsUser
	0x2c	SamrGetUserDomainPasswordInformation
> Windows 2000	0x2d	SamrRemoveMemberFromForeignDomain
-	0x2e	SamrQueryInformationDomain2
-	0x2f	SamrQueryInformationUser2
-	0x30	SamrQueryDisplayInformation2
-	0x31	SamrGetDisplayEnumerationIndex2
-	0x32	SamrCreateUser2InDomain
-	0x33	SamrQueryDisplayInformation3
-	0x34	SamrAddMultipleMembersToAlias
-	0x35	SamrRemoveMultipleMembersFromAlias
-	0x36	SamrOemChangePasswordUser2
-	0x37	SamrUnicodeChangePasswordUser2
-	0x38	SamrGetDomainPasswordInformation
-	0x39	SamrConnect2
-	0x3a	SamrSetInformationUser2
-	0x3b	SamrSetBootKeyInformation
-	0x3c	SamrGetBootKeyInformation
-	0x3d	SamrConnect3
-	0x3e	SamrConnect4
-	0x3f	SamrUnicodeChangePasswordUser3
> Windows XP and Windows Server 2003	0x40	SamrConnect5
-	0x41	SamrRidToSid
-	0x42	SamrSetDSRMPassword
-	0x43	SamrValidatePassword

To connect to the SAM server, one of the following operations are used:

- SamrConnect (0x00)
- SamrConnect2 (0x39)
- SamrConnect3 (0x3d)
- SamrConnect4 (0x3e)
- SamrConnect5 (0x40)

Then, available domains in the SAM server can be enumerated using the following operation:

- SamrEnumerateDomainsInSamServer (0x06)

The following operation is used to obtain the SID of a domain, given its name:

- SamrLookupDomainInSamServer (0x05)

This operation typically returns the BUILTIN domain (S-1-5-32) and the machine domain (local domain for a non-domain controller machine, NT 4 or Active Directory domain for a domain controller machine).
The domain SID can then be used to open a given domain:

- SamrOpenDomain (0x07)

General information about the opened domain can be obtained or set with the following operations:

- SamrQueryInformationDomain (0x08)
- SamrQueryInformationDomain2 (0x2e)
- SamrSetInformationDomain (0x09)

Once a domain is opened, it is possible to enumerate groups, aliases and users, using the following operations:

- SamrEnumerateGroupsInDomain (0x0b)
- SamrEnumerateAliasesInDomain (0x0f)
- SamrEnumerateUsersInDomain (0x0d)

RID and names resolution inside an opened domain are implemented by the following operations:

- SamrLookupNamesInDomain (0x11)
- SamrLookupIdsInDomain (0x12)

Domain password policies can be obtained with the following operations:

- SamrGetUserDomainPasswordInformation (0x2c)
- SamrGetDomainPasswordInformation (0x38)

To create a new group, alias or user in the opened domain, the following operations can be used:

- SamrCreateGroupInDomain (0x0a)
- SamrCreateAliasInDomain (0x0e)
- SamrCreateUserInDomain (0x0c)
- SamrCreateUser2InDomain (0x32)

To open an existing group, alias or user in the opened domain, the following operations exist:

- SamrOpenGroup (0x13)
- SamrOpenAlias (0x1b)
- SamrOpenUser (0x22)

To delete an existing group, alias or user in the opened domain, the following operations exist:

- SamrDeleteGroup (0x17)
- SamrDeleteAlias (0x1e)
- SamrDeleteUser (0x23)

To obtain a list of members in groups or aliases, the following operations can be used:

- SamrGetMembersInGroup (0x19)
- SamrGetMembersInAlias (0x21)

To add or remove a member to a group or alias, the following operations are available:

- SamrAddMemberToGroup (0x16)
- SamrAddMemberToAlias (0x1f)
- SamrRemoveMemberFromGroup (0x18)
- SamrRemoveMemberFromAlias (0x20)

For aliases, it is also possible to add or remove multiple members to or from an alias:

- SamrAddMultipleMembersToAlias (0x34)
- SamrRemoveMultipleMembersFromAlias (0x35)

To obtain or set information about a given group or alias, the following operations exist:

- SamrQueryInformationGroup (0x14)

- SamrQueryInformationAlias (0x1c)
- SamrSetInformationGroup (0x15)
- SamrSetInformationAlias (0x1d)

Similar operations exist for accounts management:

- SamrQueryInformationUser (0x24)
- SamrQueryInformationUser2 (0x2f)
- SamrSetInformationUser (0x25)
- SamrSetInformationUser2 (0x3a)

A list of groups containing a given user can be obtained with the following operation:

- SamrGetGroupsForUser (0x27)

Finally, handles returned by the following operations are supposed to be closed, using the SamrCloseHandle (0x01) operation:

- SamrConnect (0x00)
- SamrConnect2 (0x39)
- SamrConnect3 (0x3d)
- SamrConnect4 (0x3e)
- SamrConnect5 (0x40)
- SamrOpenDomain (0x07)
- SamrOpenGroup (0x13)
- SamrOpenAlias (0x1b)
- SamrOpenUser (0x22)
- SamrCreateUserInDomain (0x0c)
- SamrCreateUser2InDomain (0x32)
- SamrCreateAliasInDomain (0x0e)
- SamrCreateGroupInDomain (0x0a)

Appendix C
Network Diagram