Global Information Assurance Certification Paper

NetWare 4 and 5 Security Guide and Checklist

Mark Sanderson

5 September 2001

GSEC Practical Assignment
Version 1.2e, December 2000 (amended May 22, 2001)

Table of Contents

Introduction

NetWare security is a topic that is not frequently or voluminously addressed. Novell Directory Services (NDS), the underpinning for NetWare security, however apparently has had a significant influence on Microsoft's Active Directory. If only from an academic standpoint, understanding NetWare security may be a good introduction to Directory security[1] and for those who administer NetWare, hopefully, this guide and checklist will add value to security operations.

This paper has two main parts. The first part is a collection of NetWare 4 and 5 security topics in the form of a guide. The second part is a checklist based on the guidance. The format of this paper is patterned after the SANS Step-by-Step guides and checklists, but for NetWare. Numerous references in the form of Internet addresses are contained throughout the body of the document. Some additional references are included in the endnotes at the end of the document.

Physically Secure NetWare Servers

NetWare Server Physical Security

NetWare servers must be physically secure for at least the following two reasons: 1) NetWare server design and implementation does not offer much in the way of protection, and 2) NetWare servers start and run without accounts; i.e., there is no way to know or audit who does what at the keyboard of a NetWare server. Because of the lack of server protection, anyone with physical access to a server could do almost anything wuch as cause irreparable damage to the system or steal NDS information that contains passwords, user and system information. Because of the lack of accountability, there would be no way to find out who had done malicious activities at the server console. In sum, NetWare server security depends on physical security of the server itself. This anomaly is mostly well known throughout the NetWare community.

Many NetWare administrators use the lockscreen function of the *Monitor* NetWare Loadable Module (NLM) to lock a NetWare server in a server room. The *Monitor* lockscreen simply takes a password when loaded. Thereafter, typing the matching password unlocks the console. This is a reasonable mechanism to prevent someone from accidentally accessing a NetWare server, but it is not a robust mechanism. Among a number of NetWare servers in a server room a casual observer would occasionally find a small percentage of server screens unlocked because the *Monitor* lockscreen cannot be invoked during remote administration using *RCONSOLE*. Further, there is no password policy enforcement or limit on failed attempts, for example. An impatient attacker could always simply power off the server and restart it. By default, when the server restarts, it comes up without the lockscreen and the console is completely unprotected.

Attacks

To attack the *Monitor* lockscreen, an attacker could simply try a variety of likely values, based on any understanding of the personnel, mission, facility and so on. If none of these values succeeded, the attacker could simply power off the server and then power it back on and restart the server. This attack requires physical access to the server. Also, the attacker's activity must be unobserved.

Recommendations

- **SecureConsole** – SecureConsole (http://www.serversystems.com/SecureConsole.htm) is a third-party fileserver console security application that adds an additional level of control and accountability to the NetWare server. SecureConsole can control the level of access of individual users of NetWare groups, including which console commands can be used, which console applications can be seen, and whether the actions of individuals are audited. The SecureConsole console autolocking and screen saver features automatically locks the server console.
- **CONLOG.NLM** – CONLOG.NLM trakcs all activity at the server console. Once the NLM is loaded, NetWare creates a logfile, called consol.log in the SYS:ETC directory. In this logfile, NetWare stores all console information, including error messages. This file can be viewed anytime, tracking errrors and looking for unauthorized activity.
- **Lockable Equipment Rack** – An equipment rack with lockable doors can add a measure of protection for the NetWare server.

Server Room Security

Server room security should include physical, procedural and personnel security measures. At the least, all server rooms require locking doors that limit access to unauthorized personnel. A number of additional physical security may be needed depending on operational necessity. Examples of additional physical security includes some combination of the following:

- Guards to observe entrances and exits
- Security escorts for server room access
- Identification badge or token with authentication and/or audit scheme
- Cipher locks on doors that require a pin value
- Cameras to record activity
- Sensors and alarms to detect and alert to unauthorized entry

Procedures to enforce and test/audit physical security controls should be instituted. Procedural policy should also limit or prohibit personnel from bringing most types of computers and storage media into and out of the server room. A policy to restrict or prohibit data transmission devices, such as hand-held and wireless devices should be considered.

Server room security is also affected by personnel security policy because personnel security includes background checks and efforts to investigate and prosecute personnel policy abuses. Personnel roles may be used to help specify which persons require server room access. Personnel security should also include rigorous security education and awareness measures.

Attacks

Server room attacks vary depending on personnel and procedural security policies and the physical configuration and components of the server room facility. Examples of attacks include the following:

- Unauthorized entry because entries are not locked or because of inadequate locks
- Use of keys, tokens or cipher or combination values that are stored insecurely
- Entry via floor or ceiling or other unconventional access space
- Using authorized access to read, write or delete information in an unauthorized or unintended manner; downloading information to portable media or wireless device
- Convincing person with server room access to grant access to unauthorized person

Recommendations

- **Physical Security** – Doors, windows and any other access spaces must be lockable, at a minimum. A physical access device with strong authentication that records and audits access information, such as biometrics, a badge swipe or token is preferable. Guards or escorts and additional physical security may also be needed.
- **Procedural Security** – Procedures to enforce, test and review physical and personnel security controls should be instituted. Data storage and transmission devices should be controlled or prohibited.
- **Personnel Security** – Security education and awareness policies should be instituted and frequently updated. Personnel roles may be helpful in determining access.

Check for Existence of Supervisor Accounts

The Supervisor Account and Backward Compatibility Issues

Supervisor is the NetWare 3.x and earlier version of the Admin account and is included in NetWare 4.x by default for backward compatibility. The *Supervisor* object in the NetWare tree is invisible to the standard NDS (non-bindery) utilities and its existence surprises many. Discovery of the *Supervisor* object and analysis of NDS files is addressed in the section titled "Reading the NDS Files; Analyzing Output" below. The *Supervisor* account is a potential vulnerability, even though the account and its password may be well managed, because the password could be compromised using hacker programs described below. For this reason, it is good to determine the existence of the *Supervisor* account so that it can be disabled or removed (if it is not needed for backward compatibility with NetWare 3.x).

As stated in *Inside NetWare 4.1*, Doug Bierer, "NetWare 4.1's bindery services mechanism recognizes a bindery user *Supervisor*, which is a special user designed for programs and clients that need bindery-based full access to all volumes, directories, and files on a file server. If you attempt to locate this user through NDS utilities (such as NWADMIN.EXE or NLIST.EXE), you find that this user does not exist. If you examine the server using a binder-based utility (such as SYSCON.EXE), the user *Supervisor* appears. It is safe to conclude that the user *Supervisor* is a pseudo-user."

The *Supervisor's* password is the same as the first password for the Admin user until someone changes the Supervisor password using a bindery administration utility. Although Supervisor in NetWare 2.x or 3.x is a fully privileged user, the bindery user *Supervisor* is limited in its privileges in NetWare 4.x and later. When bindery services are on, bindery-based clients or utilities can see all leaf objects in the server's bindery context as if they were in a bindery. A small number of utilities, such as PCONSOLE.EXE, SBACKUP.NLM, AUDITCON.EXE and SYSCON.EXE have a bindery mode, which can be used by a bindery client. SYSCON.EXE can be used to add, change and remove users.

Logged in as *Supervisor*, the following NCF files are accessible:

etc/initsys.ncf
system/autoexec.ncf
system/bstart.ncf
system/bstop.ncf
system/hreload.ncf
system/hunload.ncf

Bindery user *Supervisor* may be a little-known mechanism to persons with little NetWare 3.x or 2.x experience. To the extent that system administrators do not know of its existence, have not changed its password or otherwise disabled it, it is available to anyone on the system. Further, the first password for the Admin user remains the password for *Supervisor*, even after the Admin password changes. This could lead to the situation that a system administrator

thinks the *Supervisor* password has been changed with the Admin password when, in fact, it has not. On some systems, the Supervisor user may have a simple "default" initial password used for the Admin account such as "netware." To change the *Supervisor* password, one must login as *Supervisor* and issue the SETPASS.EXE DOS command. If one does not know the *Supervisor* password, that person must use a bindery-based utility, such as SYSCON.EXE, to modify the *Supervisor's* password.

Although the existence of *Supervisor* is not visible in NDS utilities (e.g., NWADMIN.EXE), it should be assumed that any actions taken by *Supervisor* are logged in NetWare's audit system. It may be possible, however, to make use of AUDITCON.EXE to disable or modify the audit system since AUDITCON.EXE is one of the utilities available to *Supervisor* in bindery mode.

Attacks

If an attacker has access to the system, he or she can try SUPE.EXE, KNOCK.EXE or other NetWare *Supervisor* password hacking programs found on the Internet.

If the *Supervisor* password were obtained by a malicious user, a potentially effective attack would be to simple add the line

load remote <a simple password>

to the autoexec.ncf. The attacker would then wait until the affected server is rebooted at which time the attacker would have remote access to that server. This type of access would allow the attacker to do anything as if physically at the server console. There are probably other ways to exceed the privilege of a typical user and there would be many ways to damage the system using *Supervisor* privilege.

An example of a denial of service attack using the *Supervisor* account is to run an old NetWare bindery utility such as FCONSOLE.EXE. FCONSOLE.EXE has the ability to down a file server and represents a potential denial of service attack if it can be launched by an unauthorized entity.

FCONSOLE.EXE basically does not run on NetWare 4.x/5.x for an unprivileged (non-Admin) user. If an unprivileged user somehow gains Admin status, FCONSOLE.EXE represents a meager subset of the authority wielded by an Admin user. However, FCONSOLE.EXE is fully functional for a *Supervisor* user in NetWare 4.x/5.x and the "Down File Server Request" packet generated by FCONSOLE.EXE can be used in denial of service attacks.

If the Supervisor user can be exploited, it may be possible to run FCONSOLE.EXE from a floppy disk and down any server, including remote servers, on which the default bindery password is used or where a hacking utility is able to uncover the Supervisor password.

An example attack scenario would run as follows:

- Get the Supervisor password using a hacking utility or by guessing or discovering the default Admin password
- Access a NetWare 2.x or 3.x system to obtain FCONSOLE.EXE and related files
- From the 2.x/3.x server PUBLIC directory, copy FCONSOLE.EXE, *.OVL, *.DAT and *.HLP to a floppy disk
- Login to the target system using the bindery Supervisor account
- Run FCONSOLE.EXE from the floppy drive
- Review connection information
- Use the change current file server option, if desired

- Use the down file server option to down the server to which you are connected
- Downing the server terminates the session with the client
- Re-run above steps as needed

With FCONSOLE.EXE, transaction tracking can be disabled prior to downing the server or any other subsequent activity. Disabling transaction tracking itself generates an audit log entry and therefore may raise rather than diminish the visibility of an attack.

Recommendation

Supervisor Account – Disable or remove this account (testing to ensure it is not needed). Otherwise, ensure that the Supervisor account is restricted to least privilege and that the password is well managed.

Manage RCONSOLE Security

RCONSOLE Security shortcomings

NetWare servers come with REMOTE.NLM, which can be loaded with a password at the server console, or from a start-up file, allowing remote access to the server from client workstations. Normally, an administrator loads REMOTE.NLM (along with a transport protocol) at the server console and supplies a password, as required by REMOTE.NLM. REMOTE.NLM enables the use of a DOS utility called RCONSOLE (remote console) to remotely access the server console from a workstation. When launched from a client, RCONSOLE prompts for a password and then sends a hash of that password to the server for authentication. For RCONSOLE to be enabled, the RCONSOLE password hash must match the REMOTE password hash stored in memory at the server.

RCONSOLE has some inherent problems. The main problem is that RCONSOLE, like the server console, does not use NDS accounts to achieve accountability. It is usually difficult or impossible to know who performed what remote server activity. Because of the lack of accountability, RCONSOLE cannot limit levels of access or control console commands or applications. Possession of the RCONSOLE password grants a person complete control of the given server as if standing physically at the server console. Protecting the RCONSOLE password, therefore, is paramount in securing NetWare.

Because it is so important to protect the RCONSOLE password, therefore it is generally considered a good practice to avoid storing the password in plain-text configuration files, even if the file system permissions prevent non-privileged persons from viewing the configuration files (file system permissions could change or be mis-managed; NDS objects could move in and out of privileged contexts, groups and organizational roles; hacker tools or scripts may appear on the Internet; developer tools can be used to read the file system,…).

Attacks

According to an Internet hacker site, it may be possible to replay the RCONSOLE hash sent to the server to gain remote administration (total) access. This attack requires software that can capture, modify and send packets. The attacker must first capture a legitimate RCONSOLE login session. The attacker would start the RCONSOLE session. When prompted for the password, the attacker then sends a packet(s) with the correct hash, host IPX address and probably also the correct NCP sequence number. This attack probably requires the NCP packet signatures be set to 1 at both the client and the server. Additional network and other programming may also be required.

If an attacker could write a program that could read the NetWare file system, it may be possible to defeat file system permissions and access control to read the configuration files in SYS:SYSTEM and SYS:ETC. This would allow an attacker to find an RCONSOLE password if it is stored in a start-up configuration file. Also, if users ever are given access to SYS:SYSTEM or SYS:ETC because of accidental mis-configuration, passwords and any other sensitive information in the configuration files would be exposed.

A brute force attack could be programmed to occur over a long weekend when the console is unattended. The attack would be to program a workstation to rapidly generate requests for remote console access using "common" and/or dictionary values. The larger the server license limit, the more feasible the attack. This attack takes advantage of an intrusion detection gap. RCONSOLE has no lockout. There are predictable delays in remote console authentication. Valid passwords are acknowledged immediately while invalid password response times are delayed several seconds. A brute force attack would be tuned to ignore delays on negative responses. Further, while failed RCONSOLE attempts are logged, other approaches (e.g., XCONSOLE) avoid effective logging.

REMOTE.NLM allows the use of an encrypted password stored on disk at the server. To obtain an encrypted password, the administrator must first load REMOTE.NLM using the unencrypted password and then determine the encrypted value of the password by executing the REMOTE ENCRYPT command. For example, the RCONSOLE password can be encrypted using the following server console commands:

```
LOAD REMOTE SECRET
REMOTE ENCRYPT
When prompted, enter a value such as 'SECRET'
This command displays a hash such as '870B7E366363' for the word 'SECRET'
Choose 'Y' to write the hash value to disk
```

The hash value '870B7E366363' is now written into the file SYS:SYSTEM\ldremote.ncf and the REMOTE.NLM password is now 'SECRET.'

In the 1999 timeframe, an attack against encrypted REMOTE.NLM passwords called REMOTE.EXE was published on the Internet and described in an InfoWorld Security Watch article by McClure and Scambray (I can no longer find references to this). This attack takes the hash value stored in the SYS:SYSTEM\ldremote.ncf file and produces its unencrypted value (i.e., the RCONSOLE password). The Novell algorithm to hash REMOTE passwords uses a time byte for salting, one of 255 constants in a hash table, byte reordering, and bitwise XORing. REMOTE.EXE uses the following steps to decrypt REMOTE.NLM passwords:

1. Isolate the time byte, low-order and high-order bytes
2. Realign bits of the high and low-order bytes and drop the first four bytes
3. Use a hash table to do a look-up for each byte
4. Subtract the password length (excluding the time byte) from each byte
5. Do a modulo 2 addition of a value (0xFF – time byte) with each byte

This attack requires access to the SYS:SYSTEM\ldremote.ncf file. In an October 1999 InfoWorld Security Watch column, McClure and Scambray described at attack using Compaq's Insight Manager tool that could make it possible for an attacker to access the ldremote.ncf file (http://www.infoworld.com/articles/op/xml/99/10/18/991018opsecwatch.xml, see also http://www.guard.dubna.ru/cgibug.html). Though these vulnerabilities may not apply or may have been patched, they serve to remind us that occasionally vulnerabilities are uncovered that expose separate vulnerabilities and that combination of vulnerabilities can sometimes be used to

defeat security. Specifically, there may be unconventional methods to access "protected" NetWare files.

Recommendations

- **SecureRemote** – SecureRemote, a third party application (http://www.serversystems.com/SecureRemote.htm), allows individual and group access control, including which console commands and applications can be used and whether they are audited.
- **Configuration Files** – Loading REMOTE and supplying the (RCONSOLE) password in configuration files should be avoided.
- **Encrypted Remote Password** – Writing the REMOTE password hash to disk should be avoided.
- **Password Management** – Good password management for the RCONSOLE password should be employed

Set NCP Packet Signatures

How to Protect Against Session Spoofing

Packet signatures are used to prevent spoof, replay, data and session modification attacks. Packet signatures, if enabled, are created and added to every packet sent between stations based on a negotiated agreement to use packet signatures. The rationale behind a packet signature is that in an insecure environment, the server or client can require packet signatures to ensure validity of every transmission. Novell added packet signature capability to version 3.12 of NetWare at least partly in response to well-publicized hacks from a group of students in the Netherlands. Proper generation of a packet signature requires a valid initial authentication and the Virtual Local Machine (VLM) software on the workstation with the RSA.VLM and SECURITY.VLM components. If a hacker attempts to generate phony packets, the server detects this immediately and can shut down the connection. A high level of packet signatures increases network security but degrades network performance from approximately three to seven percent. Server packet signature levels are as follows:

0	disable packet signatures
1	add packet signatures only if requested by the client
2	add packet signatures if the client can, but do not require them
3	require packet signatures

Corresponding packet signature levels are set at each client.

Attacks

One indication of how NCP packet signatures might impact NetWare security is to review what are the popular hacker tools that take advantage of a lack of packet authentication and integrity. The following information is bundled with the Pandora NetWare hacker toolset found at the Nomad Mobile Research Centre (www.nmrc.org):

	Server Level 1	Server Level 2	Server Level 3
Client Level 1	IPX hijacking Wild pad attack GameOver attack Havoc & other DoS	Level3-1 attack GameOver attack Havoc & other DoS	Level3-1 attack GameOver attack Havoc & other DoS

Client Level 2	GameOver attack Havoc & other DoS	GameOver attack Havoc & other DoS	GameOver attack Havoc & other DoS
Client Level 3	GameOver attack Havoc & other DoS	GameOver attack Havoc & other DoS	GameOver attack Havoc & other DoS

The table shows some of the attacks that work at the different NCP packet signature levels.

Recommendations

- **Use Packet Signatures** – Set both server and client levels to 3.

Configure Server Settings

Achieving Secure Server Configurations

As stated above, there may be various ways that access to server configuration files could be obtained. Therefore, it is important to never include passwords in startup files.

A description of security-relevant server SET parameters is maintained in the *NetWare Enhanced Security Administration* Novell document (http://www.novell.com/documentation/lg/nw4/pdfdoc/nesa_enu.pdf).

Servers can be configured to reference the NetWare script file secure.ncf, which contains server settings for U.S. Class C2 security and European Class F-C2/E2 security criteria (see the following document for more details: http://www.novell.com/documentation/lg/nw4/docui/index.html#../nesa_enu/data/h8y31mbo.html). The secure.ncf script file comes with a standard NetWare server installation and can be enabled by writing the following configuration line into either startup.ncf or autoexec.ncf:

```
SET Enable secure.ncf=ON
```

The secure.ncf script contains server settings as follows:

- SET Allow Unencrypted Passwords=OFF
- SET Allow Audit Passwords=OFF
- SET Automatically Repair Bad Volumes=ON
- SET Reject NCP Packets with bad lengths=ON
- SET IPX NetBOIS Replication Option=0
- SET Reject NCP Packets with bad components=ON
- SET Additional Security Checks=ON
- SET Check Equivalent to Me=ON
- SET NCP Packet Signature Option=3
- SECURE CONSOLE

Recommendations

These settings address several of the issues raised in this checklist. It is recommended that the various settings secure.ncf be compared against the current NetWare server security policy and adjusted, as needed.

Install the latest server patches from www.support.novell.com. Be certain to have at least the patches Novell lists on its "Minimum Patch List" (http://support.novell.com/cgi-bin/search/searchtid.cgi?/2930772.htm). Search through the other patches for those that specifically apply to your system.

Infrastructure Security Settings[2]

The Principle of Least Privilege Applied to Network Services

In general, most servers run a number of default, common services based on whatever the manufacturer deems to be the set of default server functionality. Typically, many of these services are not needed in a given configuration and often running unneeded services adds potential vulnerabilities.

It is a good security practice to turn off all unneeded network services and one by one enable needed network services. In selecting needed network services, care should be taken to avoid services with known vulnerabilities, outdated versions of network services, and nonsecure services where there is a more secure alternative. For example, clear text protocols like telnet and ftp should be avoided where possible and never used for remote administration. To the extent possible, use a Secure Socket Layer (SSL) service like Secure Shell (ssh) or other similarly secure service for remote administration.

Recommendations

Research port and application vulnerability information from security experts, books and publications and on the Web. For example, Mitre Common Vulnerabilities and Exploitations (CVE) (http://cve.mitre.org/cve/) notes issues with ports 7 and 19 as "Echo and chargen, or other combinations of UDP services, can be used in tandem to flood the server, a.k.a. UDP bomb or UDP packet storm." Also, older versions of many applications often have posted attacks. Some Internet sources of network service vulnerability information include the following:

- Mitre CVE (http://cve.mitre.org/cve/)
- CERT Coordination Center (http://www.cert.org/)
- Internet Security Systems (ISS) X-Force (http://xforce.iss.net/)
- System Administration, Networking, and Security (SANS) (http://www.incidents.org/)
- U.S. Department of Energy Computer Incident Advisory Center (CIAC) (http://www.ciac.org/ciac/)

To the extent that vulnerable ports cannot be eliminated, these services must be blocked at a firewall or screening router and/or accessible only through a proxy. If the information and/or service on the hosting server is considered critical, a double firewall or other multiple layers of security may be required to ensure that no single misconfiguration can expose the information or service to an external network.

Monitoring and Updating Security and Responding to Incidents[3]

Some Guidelines for Achieving Ongoing Security

As security policies, baselines and implementations mature, there is still the ever present need to monitor and update security and respond to incidents.

Recommendations

- **Monitor and update patches**: Periodically review "Minimum Patch List" at http://support.novell.com/cgi-bin/search/searchtid.cgi?/2930772.htm
- **Regularly review Admin and other elevated-privilege accounts**: Periodically report [Root]-level elevated-privilege accounts, container-level elevated-privilege accounts, and users with elevated-privilege to significant file system and application objects to the information security officer and to the appropriate application or information owners
- **Regularly monitor and update group, user, and file security status**: remove user permissions immediately upon termination of employment; set contractor accounts to expire periodically; establish a regular monitoring program to review security policies; disable idle user accounts; set a policy for dealing with accounts of users leaving the organization
- **Regularly run a port scanning tool and a vulnerability scanning tool**: Be sure that server administrators know which service ports are running and why; regularly use a vulnerability scanning tool and address all significant vulnerabilities reported
- **Establish procedures and call lists for responding to incidents**: Consider the following five step incident response action plan:
    1. Perform event detection and categorization
    2. Establish an emergency response team call tree
    3. Assign urgency and response levels and procedures
    4. Describe any known solutions and mitigation procedures
    5. Address any legal issues

Run Security Analysis Tools

Review NDS Tree and File System Security Attributes

If your organization can justify the cost of a commercial NetWare security analysis tool, such a tool can greatly improve detection and response for a number of NDS object and file system security issues. Specifically, a commercial NetWare security analysis tool can detect invisible objects, review rights to objects and files and greatly improve the security administration of user accounts. The following are some queries that a system administration or security team can run:

1. Detect and list invisible containers or objects; document who has rights to any invisible objects
2. List and review any objects with Supervisor or elevated rights to the Root object
3. Detect OUs with no intruder detection
4. Detect OUs with no intruder attempt reset interval
5. Detect OUs that do not lock accounts after intruder detection
6. Show rights to file servers
7. Show rights to volumes
8. List rights to critical objects
9. List rights to critical directories and files
10. Review user template settings

The following are some queries that a help desk or data security team can run:

1. Detect inactive accounts
2. List locked accounts
3. Detect users who do not require a password
4. Show users with password that never expire
5. List days between forced account password changes
6. Show users who do not require unique passwords
7. Detect users whose passwords are less than 6 characters

8. Detect users with easily guessed passwords
9. Show accounts with more than 4 concurrent connections

Some of the commercial vendors of NetWare security analysis tools include the following:

- BindView bv-Control for NDS eDirectory (http://www.bindview.com/products/Control/eDirectory.cfm)
- Visual Click DSRAZOR (http://www.visualclick.com/)
- Intrusion.com Kane Security Analyst (http://www.intrusion.com/product/product.asp?lngProdNmId=4)
- Pentasafe Vigilent Security Agent for NetWare (http://www.pentasafe.com/products/vsanetware.htm)
- NetVision Directory Alert/Server Alert (http://www.netvisn.com/products/diralert.html)

If your organization does not wish to purchase a NetWare security analysis tool, these or similar tools can be run and evaluated by consulting companies such as the following:

- Novell Consulting (http://www.novell.com/consulting/index.html)
- BindView Razor Team (http://www.bindview.com/razor.cfm)
- Visual Click (http://www.visualclick.com/)

Recommendations

If your organization does not already own a set of commercial NetWare security analysis tools, consider purchasing such tools or regularly contracting security evaluation services with a qualified commercial NetWare security analysis team.

Reading the NDS Files; Analyzing Output

A Glimpse of How NetWare Security Works Under the Hood

Reading the NDS files in the hidden _NETWARE directory requires loading an NLM. Loading NLMs requires physical access to a server console or use of the client workstation remote console RCONSOLE utility with a password. There are very few NLMs that read or copy the NDS files. Two such NLMs are NETBASIC.NLM and DSMAINT.NLM.

NETBASIC.NLM comes with NetWare and is available by default. Loading NETBASIC.NLM and typing "shell" enters the user into a shell with basic functionality. In the NETBASIC.NLM shell, the user can change directory into the _NETWARE directory, list and copy any files. Copying each NDS file to a non-hidden directory makes the NDS files available to anyone with rights to that directory.

DSMAINT.NLM is available from various NetWare support packs (the support packs that contain DSMAINT.NLM can be searched at the Novell Internet Web site at http://support.novell.com/filefinder/). Loading DSMAINT.NLM and NDSCOPY produces a file called BACKUP.DS in the SYS:SYSTEM directory. BACKUP.DS is a concatenation of the NDS database files and can be parsed by CONVERT.EXE (available from http://www.nmrc.org) and possibly other hacker tools found on the Internet, or read using a hexidecimal editor.

A third NLM that can read or copy the hidden NDS files is called JCMD.NLM, which has been available on the Internet. There may be other Novell, commercial or proprietary NLMs that can read or copy the NDS files.

With the NDS files out of the hidden _NETWARE directory and into an accessible directory, hacker tools found on the Internet (namely http://www.nmrc.org) can read the files and crack passwords the Admin and user accounts. One tool called "IMP" both shows the NDS tree

structure and cracks passwords.

Attacks

If the NDS files can be obtained from the hidden _NETWARE directory, NetWare password cracking tools, like IMP, can be used to discover users with no passwords and to crack any other target user accounts. This could be accomplished with physical access to a server console, compromise of the RCONSOLE password, exploitation of the Supervisor account, other attacks described in this paper or some other attack. It may be possible to use file I/O libraries of programming languages supported by NetWare (such as Java) to read the contents of the hidden _NETWARE directory. Such a capability could give an attacker Admin-level access to and control of a target system.

How an Attacker Could Backdoor NetWare

If after gaining supervisory container access, an unauthorized user commandeered an existing user object or created a new user object, this action would be readily apparent to the network administrator through the NWADMIN or CONSOLEONE administrator tool. Therefore, the attacker would want to disguise or conceal the existence of his or her user object from the administrator.

Access to all objects in NDS is controlled through object rights. The same is true for directory and file access on the NetWare server through property rights. Object and property rights can be explicitly granted or implicitly flow down from the parent object. A feature called an Inherited Rights Filter (IRF) controls the implicit assignment or flow down of rights. Concealment of an object can be achieved by manipulating the IRF.

In summary, the specific actions taken to conceal a user object are as follows:

- Create an organizational unit
- Create the user object
- Place the user object within the organizational unit
- Grant the user Supervisory privileges at the [Root] of the NDS tree
- Make the user a trustee for the organizational unit (this is necessary to complete the remainding of the step)
- Block all rights to the organizational unit from outside the organizational unit, this step effectively cuts the organizational unit out of the NDS tree

A knowledgeable attacker would crack the Admin password and create a number of Admin-equivalent or other privileged users. The attacker would then create organizational units (OU) in various places in the NDS tree and block all outside rights as described above. Standard Novell administrator tools cannot detect the hidden OU and user objects.

Recommendations

- **Password Management** – Output from the IMP password cracking tool suggests that a password with at least nine lower-case alpha characters would take more than one month to crack, on average, using an Intel Pentium II 300 MHz computer. A password with length of at least seven mixed alpha, numeric and special characters would probably take much more than a month to crack. This information suggests that a password policy of at least seven mixed characters with a password expiration of two months for typical user accounts might reduce some of the risk of cracked passwords although it is still possible to crack passwords using this standard. The password policy for Admin and privileged accounts should be much more strict.
- **Security Analysis Tools** – A number of highly effective commercial NetWare security administration and scanning tools exist including the tools listed in the *Run Security Analysis Tools* section above.

Most of these tools are helpful with implementing accounts and passwords, NDS object and file system rights, discovering invisible objects, trustee assignments and IRFs, and other security attributes. There may also be other similarly useful tools available from Novell or other commercial entities.

Other NetWare Security Measures

There are many other general NetWare and IT security measures that should be considered. A few additional NetWare security measures are listed below.

Recommendations

- **System Security Policy** – Create, update and maintain a NetWare system security policy.
- **Tools and Training** – Consider procuring computer and network security management tools and training for administrators and users. Enterprise spending authorities may also benefit from general security training. Novell posts some ideas about tools and training online such as its DeveloperNet (http://developer.novell.com/) and training (http://www.novell.com/education/index.html) Web sites.
- **Audit** – Determine security and critical events based on your system security policy. Enable AUDITCON, or a commercial NetWare auditing tool, to monitor and review these events. Review conso.log in SYS:ETC for console events.

Some NetWare Assessment Items

Getting Started with an In-house NetWare Security Assessment

This section describes some generic things that a security team could do to get started with an enterprise NetWare security assessment. To avoid legal liability and violation of ethics standards, those with authority to approve such activities should pre-approve all interactive assessment methods based on a test plan with clearly identified scopes and mitigation procedures for unexpected events. The enterprise's legal counsel should be involved in the approval process.

Internet Information

Depending on the objectives of the test plan, Internet information should be thoroughly searched to determine that sensitive enterprise information is not publicly available. Sensitive information that should not be publicly available may include enterprise confidential or proprietary information, privacy information, information about network infrastructure, servers, application and desktop operating systems, other technology used within the enterprise, and any information that could help an attacker derive user account and password information. To the extent that remote users log into a NetWare or any other system from the Internet, the user ID and password and/or password hash and any other credential information used to log in should be encrypted during transmission and not stored on any client system.

All Web pages and applications that accept user input should be continuously checked to ensure that the Web server and application programming can safely handle all types of mal-formed and overflow input.

Consider implementing the security policy that all Internet activity associated with IP addresses registered to the enterprise should not be allowed to expose or embarrass the enterprise. For

example, non-secure test or development servers (and production servers) should not be reachable from the Internet and enterprise workstations should not be allowed to access Web sites or exchange email not in keeping with enterprise standards. One rationale for this policy is that the international IP address registry is publicly available on the Internet (http://www.arin.net/) and the fact of hacked servers or participation in non-business-related Internet activities could be publicly linked with the enterprise.

Intranet Information

Sources of Intranet information may include Intranet Web servers, directory information, enterprise databases and database applications and network infrastructure information. Depending on the objectives of the test plan, Intranet information should be reviewed to determine whether too much information is available to potential insider threat agents.

The Intranet should be searched thoroughly for logistical information that could help an attacker to identify, locate and target sensitive information and systems. Information such as server names, addresses and locations, lists of accounts and identities associated with sensitive systems, and perhaps even program funding, managers and developers, schedules, and other logistics information should be restricted to the extent reasonable.

NetWare-specific Intranet information may include information available through the use of public NetWare utilities such as those in the PUBLIC NetWare directory typically mapped to the Z:PUBLIC NetWare drive. NetWare administration utilities such as NWADMIN and CONSOLEONE should not be available to typical users. Restricting powerful utilities like NLIST should be considered. SYS:SYSTEM, SYS:ETC, the login directories of others and other sensitive directories should be inaccessible or limited to those with privilege or need-to-know.

Domain Name System (DNS) Information

DNS information can quickly give an attacker with insider access a quick mapping of internal networks and mail systems. In the absence of a network management or network inventory system, DNS can be used to roughly determine which hosts and servers participate in a given namespace. If DNS is used and a name server is reachable, a transfer of all zone host and server records can be accomplished as follows:

```
nslookup
set type=all
ls –d domain > file.ext
```

In the example above, the *domain* could be any DNS domain known to the name server (such as the domain listed by the "nslookup" command). All records from this zone transfer would be written to the file "file.ext."

Output from the zone transfer could be parsed for IP addresses and used as input to the shareware port-scanning tool NMAP. To quickly find a particular type of server, the following NMAP syntax could be used:

```
nmap –O –iL ParsedZoneTransfer.txt > nmapout.txt
```

In the example above, the ParsedZoneTransfer.txt file would contain only unique IP addresses as parsed from the output of a zone transfer. The output file nmapout.txt could then be parsed for entries of a particular type.

Public-Access Terminals/Kiosks

Public access terminals and kiosks should be physically separated from the enterprise's Intranet or, at least, logically separated using at least two layers of security, such as two layers of

firewalls. The desktop should be secure such that a person cannot access a command shell or interact with the file system (or registry) unless read-only access is provided to designated files. Any public LAN servers should be behind at least one firewall.

If the public access terminals and kiosks are not physically or logically separated from the enterprise's Intranet or backbone, it may be possible for an attacker to plug a laptop computer into the connection media for one of the public access terminals. This assumes that layer-two security has not been implemented. If this vulnerability is available, the attacker may be able to scan the backbone, detect vulnerabilities, access and change information, disrupt host and network functionality and so on.

If it is possible to directly access a DOS command shell or to break out of an application shell to access a command shell, it may be possible to use nlist and other NetWare commands to derive information that could be used to attack the system. A sampling of commands that an attacker might run is as follows:

Syntax	Description
cx /r nlist user /d > userDB.doc	Change context to [Root] Generate a detailed listing of all users including descriptions, group memberships, login information security equivalences, etc.; output listing to file userDB.doc; this file can then be searched for security attributes or used in social engineering attacks
cx /r nlist "organizational role" show description, occupant	Change context to [Root] Generate a listing of all organizational roles showing the description and all occupants; organizational roles are often used for administrators and other privileged users; this information can be used to target accounts and in social engineering attacks
DSREPORT Select START CONTAINER (container to review) Select Objects & Attributes Select Object = User Select Attribute = Login Disabled Login Expiration Time Login Grace Limit Login Intruder Address Login Maximum Simultaneous Password Allow Change Password Expiration Interval Password Expiration Time Password Minimum Length Password Unique Required Passwords Used	The DSREPORT command will show detailed user login and password information. Note that a report could be generated to show all users with a minimum password length of zero. All such accounts can be accessed without a password.

It would be a good idea to make sure that all public-access servers are separated from the public LAN by at least one firewall so that direct network access to the servers is not possible. Filtering based on both addresses and services should be implemented at the firewall.

Server Room/Facility

Make sure it is not possible to gain physical access to a NetWare server console. If physical access is possible, an attacker could edit one of the startup files to load or remove NLMs. A common attacker approach might be to load remote with a password of his or her choosing.

The attacker could then down and restart a server or wait for a scheduled or predictable restart. Another possible attack would be to load netbasic, dsrepair or another tool to attempt to access and remove the NDS (password) files.

Unattended Workstations

There should never be unattended workstations. One vulnerability associated with unattended workstations is the opportunity for an attacker to access a command shell and run NetWare commands such as the commands described in the table in the public-access terminals/kiosks section above. Another vulnerability is the opportunity for an attacker to access password information for the workstation. For example, from a Windows NT/2000 workstation, an attacker could run the following command:

```
rdisk /s
```

The rdisk command with the /s option creates a repair (floppy) disk and adds the SAM database. The workstation passwords in the SAM database can now be read with the password cracking tool l0phtcrack (www.atstake.com). In many cases, the Windows workstation passwords probably match the users' NetWare passwords.

If the workstation is locked or powered off, an attacker could use an NTFSDOS, Linux Kernel or other boot disk to boot or reboot the workstation and read the SAM database.

Supervisor Accounts

Check to make sure any Supervisor accounts have been removed, or are limited in rights and have strong password management. Depending on intruder lockout values, an attacker could always try to brute-force log in directly to the Supervisor account with common default passwords using the DOS bindery NetWare login.exe with the following syntax:

```
Login Supervisor /b <netware>
```

Otherwise, an attacker could try SUPE.EXE, KNOCK.EXE or other NetWare *Supervisor* password hacking programs found on the Internet.

Help Desk

In addition to the technical vulnerabilities and concerns described above, an assessment team should carefully plan a variety of social engineering scenarios to assess how the help desk responds.

Checklist

General Security Guidelines

Guideline	Name of Person Responsible	Date Completed	Initials
1. Enforce the least privilege principle			
2. install minimum applications and services; add only needed applications and services			
3. Carefully plan groups and their permissions			
4. Identify the owners of data on the system			
5. Implement enough elevated privilege accounts to successfully administer the system, but not more than is needed			

6. Implement remote access and administration that uses strong user authentication and encrypts all sensitive data in transmission			
7. Do not allow modems in servers			
8. Build and run server in physically secure area			
9. Use NDS/eDirectory authentication to the extent possible			
10. Keep system and security software and patches up to date			

Physically Secure NetWare Servers

Guideline	Name of Person Responsible	Date Completed	Initials
1. Place critical NetWare servers in a building with restricted access			
2. Secure the server room			
2.1. (Advanced) Use access authentication, such as a badge or pin, that ties access to an individual and can be monitored and audited			
2.2. (Advanced) Employing guards to observe entrances and exits and/or using cameras to record activity			
2.3. (Advanced) Use escorts for server room access			
2.4. (Advanced) Add cipher locks to doors that require a pin			
2.5. (Advanced) Use sensors and alarms to detect and alert to unauthorized entry			
2.6. Provide temperature and humidity controls			
2.7. Provide chemical-based fire control system			
2.8. Install a UPS and test			
2.9. Lock CPU case			
3. (Advanced) Use `SecureConsole` (www.serversystems.com), or a comparable third-party file server console security application, that adds an additional level of control and accountability to the NetWare server. `SecureConsole` can control the level of access of individual users or NetWare groups, including which console commands can be used, which console applications can be seen, and whether the actions of individuals are audited. The `SecureConsole` console autolocking and screen saver features automatically locks the server console.			
4. Use `CONLOG.NLM` to track all activity at the server console. Once the NetWare Loadable Module (NLM) is loaded, NetWare created a logfile, called `consol.log`, in the `SYS:ETC` directory. In this logfile, NetWare stores all console information, including error messages. This file can be viewed anytime, tracking errors and looking for unauthorized activity.			
5. (Advanced) Use a lockable equipment rack			

Remove Supervisor Accounts

Guideline	Name of Person Responsible	Date Completed	Initials
1. Disable or remove the Supervisor account (testing to ensure it is not needed) to prevent Supervisor password grabbing attacks			

2. If the Supervisor account is essential for backward compatibility, ensure that the Supervisor account and passwords are well managed and the account privileges are reduced to a minimum.			

Manage RCONSOLE Security

Guideline	Name of Person Responsible	Date Completed	Initials
1. Disable `RCONSOLE` or use very good `RCONSOLE` password management.			
2. Do not `load remote` and supply a password in a configuration or startup file; audit startup files from time to time to ensure that no passwords are present			
3. Writing the remote password hash to disk with the `remote encrypt` utility should be avoided; storing the remote password in server RAM is more secure			
4. (Advanced) Use `SecureRemote` (www.serversystems.com), or a comparable third party application, that allows individual and group access control, including which console commands and application can be used and whether they are audited			

Set NCP Packet Signatures

Guideline	Name of Person Responsible	Date Completed	Initials
1. Consider setting both servers and clients to level 3, packet signatures required			

Configure Server Settings

Guideline	Name of Person Responsible	Date Completed	Initials
1. Don't allow passwords in startup files.			
2. Use the following set commands:			
2.1. set allow unencrypted passwords = off			
2.2. set reject NCP packets with bad lengths = on			
2.3. set reject NCP packets with bad components = on			
2.4. set check equivalent to me = on			
2.5. secure console			
2.6. If possible, remove DOS from the server.			
3. Remove unneeded utilities from the PUBLIC directory.			
4. Restrict NWAdmin from common users.			

Infrastructure Security Settings

Guideline	Name of Person Responsible	Date Completed	Initials

1. Turn off all unneeded network services and run needed services safely			
1.1. Turn off all network services that you do not need			
1.2. Verify that the services you are running are the latest versions			
1.3. Make sure that local accounts with the same name on different machines use different passwords			

Monitoring and Updating Security and Responding to Incidents

Guideline	Name of Person Responsible	Date Completed	Initials
1. Monitor and update patches			
1.1. Periodically review "Minimum Patch List" at www.support.novell.com			
1.2. Apply other relevant patches			
2. Regularly review Admin and other elevated-privilege accounts			
2.1. Periodically report Root-level elevated-privilege accounts to the Information Security Officer and to the appropriate application or information owners			
2.2. Periodically report Container-level elevated-privilege accounts to the appropriate application or information owners			
2.3. Periodically report users with elevated-privilege to significant file system and application objects to the appropriate application or information owners			
3. Regularly monitor and update group, user, and file security status			
3.1. Remove user permissions immediately upon termination of employment			
3.2. Set contractor accounts to expire periodically			
3.3. Establish a regular monitoring program to review security policies			
3.4. Disable idle user accounts			
3.5. Disable accounts for users leaving the organization rather than delete them, so that Data Security and Human Resources can maintain information to meet legal requirements. The account and its information should be stored for a minimum of 90 days. If necessary to conserve licensing costs, an account can be deleted and the user's local and network files stored on a network server for the required period. Accounts should never be reassigned to another person.			
3.6. The data owners are responsible for determining appropriate access			
4. Regularly run a port scanning tool and a vulnerability scanning tool			
5. Establish procedures and call lists for responding to incidents			
5.1. Prepare a call list of people to contact and managers with authority to make key decisions			
5.2. Create, document, and test a recovery plan			

Run Security Analysis Tools

Guideline	Name of Person Responsible	Date Completed	Initials
1. Run system and server checks (System Administrators Team)			
1.1. Detect and list invisible containers or objects; document who has rights to any invisible objects			
1.2. List and review any objects with Supervisor or elevated rights to the Root object			
1.3. Detect OUs with no intruder detection			
1.4. Detect OUs with no intruder attempt reset interval			
1.5. Detect OUs that do not lock accounts after intruder detection			
1.6. Show rights to file servers			
1.7. Show rights to volumes			
1.8. List rights to critical objects			
1.9. List rights to critical directories and files			
1.10. Review user template settings			
2. Run user account checks (Data Security or Help Desk Team)			
2.1. Detect inactive accounts			
2.2. List locked accounts			
2.3. Detect users who do not require a password			
2.4. Show users with password that never expire			
2.5. List days between forced account password changes			
2.6. Show users who do not require unique passwords			
2.7. Detect users whose passwords are less than 6 characters			
2.8. Detect users with easily guessed passwords			
2.9. Show accounts with more than 4 concurrent connections			

Endnotes

[1] See also, Mark Sanderson and David Rice, Guide to Securing Microsoft Windows 2000 Active Directory, The National Security Agency, Version 1.0, December 2000, Report Number: C4-056R-00 (http://nsa2.www.conxion.com/win2k/index.html)

[2] This section is patterned after the *Phase 6 Networking and Internet Security Settings* section of the SANS *Windows NT Step by Step* security guide, version 3.03, February 2001, Shahram Alavi, et al.

[3] This section is patterned after the *Phase 7 Monitoring and Updating Security and Responding to Incidents* section of the SANS *Windows NT Step by Step* security guide, version 3.03, February 2001, Shahram Alavi, et al.