DOCUMENT RESUME

ED 387 110 IR 017 327

AUTHOR	Snyder, Robin M.
TITLE	Netware-Specific Network Security.
PUB DATE	95
NOTE	10p.; In: Association of Small Computer Users in Education (ASCUE) Summer Conference. Proceedings (28th, North Myrtle Beach, South Carolina, June 18-22, 1995); see IR 017 305.
PUB TYPE	Reports - Descriptive (141) -- Speeches/Conference Papers (150)
EDRS PRICE	MF01/PC01 Plus Postage.
DESCRIPTORS	Access to Information; Computer Networks; *Computer Security; Computer Software Evaluation; *Computer System Design; Cost Effectiveness; Information Networks; *Information Systems; *Local Area Networks; Users (Information)
IDENTIFIERS	Client Server Computing Systems; Computer Users; *Network Management; Novell Netware; *Passwords

ABSTRACT

This paper focuses on practical and cost-effective NetWare-specific approaches to information systems and computer security. A series of real world experiences is presented that illustrate fundamental information systems and security concepts. A NetWare network is a client-server network which uses a file server to share files while client workstations access the file server via some network typology (usually ethernet or token ring). Passwords are used to authenticate users; thus, password protection is a cornerstone of network security. Passwords can be encrypted, but this feature can be disabled. The bindery, which stores information about users, groups, and printers, is critical to the network and must be backed up. Trade secrets can be hidden from users by restructuring menus. Login scripts should be protected and not used for security purposes. Guest account access should be restricted or removed. Finally, the "security.exe" program can be used to find potential security loopholes. (Contains eight references.) (AEF)

Netware-specific network security

Robin M. Snyder
Director of Academic Computing
Associate Professor of Information Systems
The Byrd School of Business
Shenandoah University1460 University Drive
Winchester, VA 22601

Abstract

Computer network security is becoming an increasingly important problemin a society that is becoming more and more dependent on informationsystems and computer technology. As Novell NetWare is the currentleader in market share among network operating systems software, thistalk will focus on practical and cost-effective NetWare-specificapproaches to information systems and computer security. This will bedone with a series of specific and practical real world experiences inthe area of information systems and computer security specific examplesthat, at the same time, illustrate general and fundamental informationsystems and computer security concepts.

Introduction

Computer networks are increasingly used to share information andresources in order to reduce the costs associated with the duplicationand sharing of such information and resources. And computer securityhas become important in direct relationship to this increase in the useof information systems and computer technology.

The area of information systems and computer security forms an everexpanding body of knowledge. A short paper can only touch the surfaceof this knowledge (see, for example, Forcht, 1994; Stallings, 1995;Kaufman, et al, 1995). So, instead of a general overview of networksecurity, this paper should be considered a continuation of the paperpresented last year (Snyder, 1994b) and will consist of a series ofspecific examples with which the author has personal experience andthat, at the same time, illustrate some fundamental and generalprinciples of network security. And, as Novell NetWare is the currentleader in market share among network operating systems software,practical and cost-effective NetWare-specific approaches to informationsystems and computer security will be featured (for general anddetailed information on NetWare 3.x, see, for example, Heywood, et al,1994).

Where appropriate, a command-line approach, as opposed to a full-screenapproach, to network commands will be used since the execution of acollection of command-line commands can be automated by placing thecommands in a batch file and executing the batch file.

And, since the goal is cost-effective and practical network security,only readily available and low cost solutions will be addressed.

Background

The purpose of a network is to share resources, typically files,printers, and information in general. But, in order to balance sharingwith security, user accounts, identified with a user identification, oruserid, are created with password access in order to limit sharing. Forconvenience, users can be made members of groups so that entire groupsof users can be given certain rights by giving those rights to thegroup, although NetWare is not designed to make group managementparticuarly easy (Snyder, 1994c). A NetWare network (3.x will beassumed in this paper) is a client-server network in that the fileserver is used to share files (and information) while clientworkstations access the (centralized) file servervia some networktopology (usually ethernet or token ring). In practice, a file serveris a high powered workstation with Novell NetWare software installed.A printer server is another type of server, but, in practice, if thefile server is not being fully utilized, the print server (and otherservers, such as SQL, modem, fax, etc.) can be installed on theworkstation comprising the file server. A local area network (LAN) mayconsist of a large number of interconnected file servers, often calleda wide area network (WAN). The entire process is sufficientlycomplicated, ever changing, and important enough that a full-timenetwork administrator and assistants are often hired to maintain thenetwork.

Passwords

A password is used to authenticate that the user is who the user claimsto be. There should be a one-to-one correspondence between users(people) and userids (user accounts). If not, consider the commonaccount STUDENT (with no password). If STUDENT has email access, thenSTUDENT can send a nasty message to the president. Who is responsible?Anyone could have used that account. On the other hand, suppose thatuser LAYNE leaves his workstation unattended. A student uses theopportunity to use the workstation to send a nasty message to thepresident. Who is responsible? In this case, the person assigned toLAYNE is responsible. This is a simple example that illustrates theimportance of password protection and the example can be used to warnusers about giving their passwords to other individuals. User mustunderstand that account access is associated with responsibility forthe actions done by that account.

For this and other reasons, password protection is an importantcornerstone of network security. Given the proper password(s), a personcan get access to anything on the network (for which it is possible toget access; even the SUPERVISOR cannot access password information inNetWare without physically disassembling the file server and dissectingthe hard drive).

For example, user RSNYDER can login to file server HORNETS with the command

```
login.exe HORNETS/RSNYDER
```

whereupon the login.exe program requests a password to authenticate theuser as RSNYDER. If the person types the proper password, the networkassumes that the person is, indeed, the person assigned to the RSNYDERaccount.

But, and here is the rub, the password must move from the clientworkstation to the file server over a wire. What if someone has accessto the wire and watches the messages go back and forth? So longpassword protection. Hardware (and software) are available for suchtasks, and the price is coming down. Think of it this way. A bridge canbe used to connects two network (of similar or dissimilar topologies).Messages from one network that are destined for the other network arepassed through the bridge.. In this sense, bridges are one way ofreducing traffic in a congested network (that is, split the network inhalf and connect the networks with a bridge). Although hardware bridgescan be purchased, a NetWare software bridge can be created from aworkstation by adding two network adapter cards to the workstation,connecting each of the adapter cards to one of the two networks to beconnected with the bridge, and installing the software to take messagesfrom either side and place it on the other, as required. (Note: NetWaresoftware bridges are somewhat limited if certain non-Novell protocolsor software are to be used, e.g., TCP/IP, PC Support). In action, thebridge software takes messages from the adapter card, as required, intothe CPU (and memory), and then to the other adapter card, from the CPU(and memory). There is nothing to stop someone, given the propersoftware and/or programming techniques, from looking at the messages asthey are being transferred.

Luckily, the NetWare login.exe program (NetWare 3.x and after) has abuilt-in encryption feature that uses the RSA algorithm that works, insimplified form, as follows.

- The login.exe program on the client workstation requests a public key from the file server.
- The file server generates a public and private key. The private key is kept at the file server. The public key is sent to theworkstation. Anyone listening (watching the wire) could obtain thepublic key.
- The client workstation gets the password from the user and encrypts it with the public key. The encrypted password is sent tothe file server. Anyone listening (watching the wire) could obtainthe encrypted password, but decrypting the message would requirethe private key, which is very difficult to determine given onlythe public key. (This is the trapdoor part of the algorithm).
- The file server decrypts the encrypted password with the private key. If the decrypted password matches the password stored at thefile server (and to which even the SUPERVISOR does not haveaccess), then the user is assumed to be valid, and login succeeds.

This scenario ignores the problem of "spoofing" where a clientworkstation attempts to look like a file server and fool the clientinto revealing information (such as a critical password) to a fake fileserver.

So, no problem. Just encrypt the passwords. But, NetWare allows thenetwork supervisor to issue the following command

Set Unencrypted Passwords On

This command woulo typically be placed in the autoexec.ncf file on thefile server (autoexec.ncf is similar to the autoexec.bat file in DOS inthat the autoexec.ncf file contains commands that are automaticallyexecuted when the file server is booted). But why?

In one particular case, the author, as Director of Academic Computingand academic network supervisor, received a request from the Registrarto install a direct connect print box (and, as is common practice, wasnot contacted before purchasing the box). It turns out that certainhardware devices do not have built-in support for NetWare encryptedpasswords. In this case, the print box could be set up in RPRINTER orPSERVER mode. In RPRINTER mode, the printer acts as a remote printer,requiring one user license on the network (a 100 user license becomes,in effect, a 99 user license). In PSERVER mode, the printer acts as aprint server, not requiring a user license on the network. RPRINTERmode does not require that encrypted passwords be turned off, butPSERVER mode does. For these reasons, the author chose to set up theprinter in RPRINTER mode. On the other hand, the administrative networksupervisor (and again, as is common practice, there was littlecommunication between the administrative network supervisor and theacademic network supervisor) chose to set up the printer in thePresident's office, using the same type of print box, in PSERVER mode.Well, when the administrative network supervisor left the university(for reasons that were never revealed), the author was called in tocheck the state of the administrative network. The userlist.exe programrevealed that a print server was active and the name of the printserver indicated that it was in the President's office. Immediatelywalking down the hall and looking at the box, the author asked the VicePresident for Business Affairs why they were not using the encryptedpassword feature on the network, since anyone watching messages on thewire would be able to determine passwords and gain access to importantinformation. The Vice President was somewhat indignant that I wouldsuggest such a possibility and the Directory of AdministrativeComputing expressed doubt as to whether the (former) administrativenetwork supervisor would have allowed it. So, we walked back down thehall to the file sever console which was, as usual, running monitor.nlmbut not locked (always leave the file server console runningmonitor.nlm lock the console whenever the file server is leftunattended, especially if remote file server console access isenabled). With a few keystrokes I brought up the autoexec.ncf file (editing the autoexec.ncf file is one of the options on the monitor.nlmmenu) and, there and behold, was the statement

Set Unencrypted Passwords On

at the end of the file. And this is a statement that must be put intothe autoexec.ncf. It just does not get there by itself. To the best ofthe author's knowledge, the print box is still run as a PSERVER, but,on the other hand, the author is no longer asked to check the state ofthe administrative network.

The moral of the story is twofold. First, the specific lesson is thatpassword access can be compromised by turning encrypted passwords off.Second, the general lesson is that subtle influences and circumstancescan undermine the security of the network, which needs constantevaluation in order to determine possible weaknesses.

Bindery

NetWare 3.x stores all of its information about users, groups,printers, and such in a data structure called a bindery. Think of thebindery as a database. Calls can be made to the bindery (using theappropriate NetWare API, application programmers interface, or SDK,software development kit). NetWare 4.x uses a somewhat moresophisticated data structure called NDS, NetWare Directory Services,that is supposed to provide bindery emulation for those file serversand applications that require it. We will limit discussion to theNetWare 3.x bindery. Information such as file and directory rights arestored in the network file system. The bindery and file system worktogether to define user rights.

A fundamental assumption of security is that any potential adversaryhas access to all public and published information. In the case of anetwork file server, what exactly is public information. Well, in termsof a file server, public information consists of any accessibleinformation in the bindery and file system. File system rights arefairly well understood by most network supervisors (read, read-write,shareable, etc., rights for users and/or groups). Bindery informationis not as well understood by network supervisors and users in general.But there is a considerable amount of network information available tomost users from the bindery.

The author has written a program that uses NetWare API calls to dumpall user accessible information in the bindery to a tree data structurethat can be printed or used for future comparison. The author intendsto implement a tree merge routine to allow comparison of the bindery atvarious points in time (the current program allows the binderyinformation to be collected before the comparison program is done).This serves a number of purposes.

- The author can see exactly what is public knowledge (from an attacker point of view) and take appropriate action.
- The author can track changes in the network over time. Since the author has written a number of software programs that are used onthe network for classroom purposes, it is important to find outabout changes sooner rather than later.

The author wrote a similar type of program, in BLISS, in 1982 to trackwhat was happening on a DEC-10 used in a Research & Development Center.Within weeks, the author knew more about what was happening as far asusers and computer usage, than the computer staff who had been therefor years. The same thing happens on a NetWare network. Within weeks,one begins to have a better picture of the network than even thenetwork staff (the author is now teaching full time and no longerDirector of Academic Computing, so things can happen without theauthor's knowledge). For example, the author can say to the networkadministrator, "I happened to notice that EVERYONE now has access tothe MALTHUS (PostScript laser printer) when before, just BUSSCH (thebusiness school) had access." (it's sometimes best not to reveal yoursource of information; it just makes the network administratornervous). To make matters worse, some institutions have policies wherea record must be kept of the users, groups, etc., that are on thenetwork. And this record is usually kept manually. But this informationis already available from the bindery. And getting it from the binderyis much less error prone than maintaining it by hand (Snyder, 1994a).In essence, maintaining a series of snapshots of the bindery allows amuch better picture of what is happening. And, as mentioned before,this is critical in being able to react to subtle influences andcircumstances can undermine the security of the network.

Since the bindery is critical to the network, it is important to backup the bindery (the bindery files in NetWare 3.x are stored in theSYS:SYSTEM directory as net$obj.sys, net$prop.sys, and net$val.sys).This can be done as follows.

- Insure that there are no other users on the network.
- Login as SUPERVISOR.
- Disable login (from fconsole.exe or from the file server console).

- Run bindfix.exe, supplied with NetWare, as SUPERVISOR from a client workstation. While fixing the bindery, the bindery is alsocompacted. Note any error messages and take appropriate action.The old bindery files are stored as net$obj.old, net$prop.old, andnet$val.old.
- If there were no problems, run bindfix.exe again. This essentially makes the old bindery files the updated bindery files.
- Enable login.
- Copy the files net$obj.old, net$prop.old, and net$val.old to the client workstation so that they are not lost should the fileserver irrevocably crash.

At a later time, the command bindrest.exe, supplied with NetWare, canbe run as SUPERVISOR from a client workstation in order to restore thebindery.

And there is always the problem that a GUEST, or other user, can stuffthe bindery by creating large amounts of bindery entries such that theperformance of the file server is compromised.

Trade Secrets

It may, on occasion, be sufficiently secure to just keep certaininformation secret. In the case of the academic file server, theRegistrar had a program called transman (transcript management) thatwas used to manage transcripts. As a practical consideration, all userson the network had access to the same menu system. In a submenu, theRegistrar could run the transman program. Now, even though users neededsufficient rights to actually run the program, the appearance on themenu system might alarm certain administrative persons. A compromisewas to rename the menu option from transman to the less obvioustechnical manual. The few people in the Registrar using the program hadlittle trouble adapting and there was less cause for alarm. (Of course,funds for a more sophisticated menu system would have allowed theproblem to be solved in another manner).

Just remember, trade secrets do not work if the trade secret is publicknowledge. That is, if the knowledge is discernible from the bindery bya normal user (or GUEST), as would be the knowledge that the MALTHUSPostScript laser printer was available to EVERYONE. Yes, printer accesscan be a security problem, especially if confidential information issent to a network printer. Just imagine a printer that "spoofs" theprinter that prints paychecks (or other confidential correspondence) bypretending to be that printer (and no one notices the difference).

Login Script

When a user uses the login.exe (or other similar) program to login to afile server, the system login script, stored as the text fileSYS:PUBLIC/net$log.dat, is run. One purpose of the login script is toset up initial drive mappings, set default printer queues, etc., thatare specific to that network. Usually maintained by the networkadministrator, some network administrators depend on this login scriptfor some form of security, such as running certain programs at start-up(e.g., anti-virus software) or for auditing purposes. Of course, thiscan be misused. The current login script at the author's universityruns the anti-virus software if the user is STUDENT (intended for thehard drives in the lab). Naturally, the case of a STUDENT login to ateachers workstation and causing the virus software to run may havedisastrous side effects, not in finding viruses, but in possiblycorrupting the hard drive or crashing the workstation of the teacher.But a client can create their own login script as, for example, thetext file C:\my$log.dat and bypass the system login script with thefollowing command.

```
login.exe /S C:\my$log.dat HORNETS/RSNYDER
```

So, do not depend on the system login script for security purpo.ses.

Another weakness (or feature) is that users can automate passwordentry. Why would someone want to automate password entry? To avoidtyping the password, of course. Automating password entry in NetWare isas easy as creating a file called C:\rsnyder.pwd that contains theplain text of the password and using the following command thatredirects the input from the file C:\rsnyder.pwd instead of from thekeyboard.

```
login.exe /S C:\my$log.dat HORNETS/RSNYDER < C:\rsnyder.pwd
```

The problem here is that anyone with physical access to the workstationhard drive can determine the network password for RSNYDER.

In terms of avoiding typing, the author is no exception. In the courseof network software development, it may be necessary to logout andlogin to the network many times during the course of a day. And, usingthe OS/2 Warp operating system with Microsoft Windows and DOS, it iseasy to open many (private) network sessions concurrently. One partialsolution to the automated password entry problem, and the one used bythe author, is to dynamically create the password file on a memorydrive the first time after the computer is turned on that the passwordis needed (this is done via a batch file). (Note: There goes my tradesecret since the scheme is now public knowledge). Thereafter, thepassword need not be typed to login to the network. But, when the powerto the workstation is turned off, the memory drive, and the passwordfile, disappear. For security purposes, however, physical access to theworkstation is restricted by always locking the office door wheneverthe workstation is left unattended and the workstation is powered downat the end of each working day.

The system login script can also be avoided by attaching, as opposed tologin, to the file server. Many network file servers maintain a GUESTaccount whose primary purpose is to allow users to attach to a networkfile server in order to user a given printer or other resource. TheGUEST account is created, with no password, when NetWare is installed.Some network supervisors may not even know of the existence of theGUEST account. One can attach to a file server with a GUEST accountwith the following command.

```
Attach Admin/guest
```

Again, no login script is executed, so that all drive mappings must bemade by the user. But a GUEST may have browsing rights to a substantialamount of information. In particular, GUEST can access the bindery as alogged in (or attached) object and can obtain a good deal ofinformation about the infrastructure of the file server (via binderyand other calls). The author can just imagine the chagrin of theadministrative network'supervisor the night that the author's entirenetworking class attached to the administrative file server as GUESTand browsed through the information available to GUEST (note: twoemployee's of administrative computing were taking the course, so itwas for demonstrative, and not devious, purposes).

One might consider either removing the GUEST account, if it is notneeded, or, at least, restricting GUEST access to certain resources byremoving the GUEST account from the group EVERYONE (which conveys asubstantial amount of read access on the file server). But, the GUESTaccount issue does need to be addressed.

Loopholes

The security.exe program, provided with NetWare, can be used by theSUPERVISOR to attempt to find potential security loopholes such asinsecure passwords (that is, the user used the userid as a password),no passwords, supervisor equivalences, root directory privileges, nologin script, and excessive rights in a certain directory. Since theprogram generates a lot of output, a suggested way to run the programis from the (secure) SUPERVISOR client workstation as follows.

```
security.exe > C:\SECURITY\95-04-18.dat
```

This command redirects the output of the security.exe program to thefile called 95-04-18.dat in subdirectory C:\SECURITY. The date is usedfor the filename so that a record can be kept of the security messages.Note: This program generates a lot of output and spurious messages. Onemight want a program to filter the output of security.exe into a moremanageable form.

In the case of the administrative file server, running the security.exeprogram revealed that less than half of the about ninety user accountshad passwords (supposedly new user accounts were being created,manually, and had not been given passwords). Repeat. All user accountsshould have passwords assigned to them. Use automated (and tradesecret) means for the initial password generation and require the userto change the initial password).

Conclusions

This paper has attempted to use a series of specific examples toillustrate general security concepts. There has been so much that hasnot been covered, but the purpose of the paper is to highlight somepractical real world experiences in the area of information systems andcomputer security that can be addressed with low cost solutions.Hopefully this objective has, in some measure, been accomplished.

References

Forcht, K. (1994). Computer security management. Danvers, MA: Boyd &Fraser.

Heywood, D., et al. (1994). Inside NetWare 3.12, 4th ed. Indianapolis,IN: New Riders Publishing.

Kaufman, C., Perlman, R., & Speciner, M. (1995). Network security:private communication in a public world. Englewood Cliffs, NJ:Prentice-Hall.

Sawicki, E. (1992). LAN desktop guide to security. Carmel, IN: SAMSPublishing.

Snyder, R. M. (1994a). The noncomputer-checked redundancy problem.Proceedings of the 6th Annual Meeting of the International Academy ofBusiness Disciplines. Pittsburgh, PA.

Snyder, R. M. (1994b). Proactive approaches to information systems andcomputer security. Proceedings of the 27th Annual Conference of theAssociation of Small Computer Users in Education. Myrtle Beach, SC.

Snyder, R. M. (1994c). Automatically managing groups in a networkenvironment. Proceedings of the 22nd Annual Conference of theInternational Business Schools Computing Association. Baltimore, MD.

Stallings, W. (1995). Network and internetwork security: principles andpractice. Englewood Cliffs, NJ: Prentice-Hall.