Global Information Assurance Certification Paper

VNC – Is it the answer to ***your*** remote control needs?

Christopher Plensdorf

Practical Version 1.2e

June 25, 2001

Introduction

It seems as though a constant struggle exists between the availability of resources and their security in the information technology field. For many systems administrators, full availability and easy access to all resources from any location at any time is the highest priority. Without careful investigation, a compromised network is sometimes the result. This article looks into the popular Virtual Network Computing, or VNC, remote control application. Following the lead of Perry Déraps in the article titled *VNC – A Call Centre Perspective* this document has been created to serve as a single source for understanding the security implications of implementing VNC as a network remote control solution. Also examined are methods by which VNC is used to gain privileged access into a network and a review of the options available within VNC to alleviate those vulnerabilities.

VNC

Virtual Network Computing is a product of the AT&T Research Labs that allows both viewing and complete control of a remote computer's desktop over any LAN or an Internet connection. It is multi-platform and is available for any win32 based operating system including Microsoft Windows 95, 98, ME, NT and 2000 (WinVNC) along with UNIX, Linux, Solaris, DEC Alpha and Macintosh. It is even ported for PDA's, both Palm based and Windows CE extending VNC's reach to the wireless platform. This paper focuses on the configuration of the Windows based port (WinVNC) and deals most with the built-in features of the software. Described are those items modifiable for the average administrator and fall short of describing the process of modifying the source code to correct known vulnerabilities, incorporate encryption, etc.

A typical use for VNC is as a network management tool, much like other remote control software, for such tasks as viewing a remote UNIX server in another room or office from a Windows computer. What makes VNC unique (and in some ways attractive to malicious users) is a combination of its platform independence, its ability to be viewed through any Java capable browser, its 'small and simple' design (the WinVNC viewer is only 150K) and the fact that it can be run directly from a floppy disk. Another big plus is that VNC is open-source and free of charge and is distributed under the GNU Public License. (http://www.uk.research.att.com/vnc/)

VNC consists of two parts. The Server component is installed on the computer that is to be controlled. The Viewer component is either installed on a remote machine or run from a floppy disk and it provides the ability to view and control the session. Authentication is achieved through a single challenge-response password scheme.

Installation

The installation of VNC takes several forms. The server component can be executed as an application or can be set to run as a service upon booting. The application is installed via an executable or a script either locally or remotely. All that is required is a single reboot to have the

service start after a remote install. To complete an automatic installation, you must simply download the appropriate version for the target server and execute the installer.

Using Sysdiff reveals that the default installer creates the following changes during the default installation. Being open-source, this also assists in determining that no malicious code has been introduced into the distribution.

(Edited for brevity)

C:\Program Files\ORL\VNC

SFN: C:\PROGRA~1\ORL\VNC

Add/change Msvcirt.dll

Add/change Msvcrt.dll

Add/change omnithread_rt.dll (SFN: OMNITH~1.DLL)

Add/change Uninst.isu

Add/change VNCHooks.dll

Add/change VNCHooks_Settings.reg (SFN: VNCHOO~1.REG)

Add/change vncviewer.exe (SFN: VNCVIE~1.EXE)

Add/change WinVNC.exe

HKLM\SYSTEM\CurrentControlSet\Services\winvnc

DisplayName: REG_SZ/REG_EXPAND_SZ VNC Server

ErrorControl: REG_DWORD 0x300bb0

ImagePath: REG_SZ/REG_EXPAND_SZ "C:\Program Files\ORL\VNC\WinVNC.exe" -service

ObjectName: REG_SZ/REG_EXPAND_SZ LocalSystem

Start: REG_DWORD 0x300c40

Type: REG_DWORD 0x300c50

HKLM\SOFTWARE\ORL\WinVNC3\Default

AutoPortSelect: REG_DWORD 0x300b60

IdleTimeout: REG_DWORD 0x300b70

InputsEnabled: REG_DWORD 0x300b80

LocalInputsDisabled: REG_DWORD 0x300b90

OnlyPollConsole: REG_DWORD 0x300ba0

OnlyPollOnEvent: REG_DWORD 0x300bb0

Password: 85 7E 92 6B 85 64 34 20 (p/w included for illustration purposes)

PollForeground: REG_DWORD 0x300bd0

PollFullScreen: REG_DWORD 0x300be0

PollUnderCursor: REG_DWORD 0x300bf0

QuerySetting: REG_DWORD 0x300c00

QueryTimeout: REG_DWORD 0x300c10

SocketConnect: REG_DWORD 0x300c20

Observing the behavior of the default installation raises initial concerns. Most important to note is the advertised location of the password in:

HKLM\SOFTWARE\ORL\WinVNC3\Default\Password

The encrypted passwords, along with IP restrictions are viewable directly from the registry.

To install via a script, it is only necessary to copy the dll's, registry keys and the server executable to the target machine. Flag the executable to run as a service and restart the computer. All of this can be done remotely, and can be accomplished via a Trojan application. This is the ideal situation for a malicious user to install VNC on a target machine. Unlike NetBus and Back Orifice, WinVNC commands complete control of the remote computer. One example of remote installation is available at: http://cwashington.netreach.net/script_repository/view_scripts.asp?Index=519&ScriptType=kixtart.

Vulnerabilities

Pick up any security journal today and you will quickly learn that the solution to solving the most prevalent vulnerabilities in most applications is simply keeping abreast of its fixes and patches. VNC is no exception. Earlier releases of VNC have been known to have serious security weaknesses with authentication, password protection and buffer overflows, among other things. Several of these still exist in the current release (V3.3.3 at time of writing). If you are looking for a secure implementation of WinVNC, knowing these weaknesses will most definitely force a strategy other than simply accepting the default installation. The following information will assist in designing the most secure WinVNC implementation.

Protocol

WinVNC runs by default on the well-known listening ports TCP 5800 and TCP 5900 although alternate ports are allowed. Making the following changes to the registry will modify the default port:

HKLM\SOFTWARE\ORL\WinVNC3\Default

AutoPortSelect: REG_DWORD	**Remove**
PortNumber: REG_DWORD <New Port Number>	**Add**

WinVNC uses a proprietary protocol that is not encrypted after authentication, but is compressed by default. Although the transmission itself is not difficult to identify, the compression does make network sniffing a bit more difficult to accomplish. The classic man-in-the-middle attack is still present (See Figure 1) though, requiring only a good network-sniffing tool such as dsniff (www.monkey.org/~dugsong/dsniff) to accomplish the task.

Although sometimes referred to "security through obscurity", it is recommended that the default port be changed. The default ports 5800 and 5900 are identified as VNC by common port scanners such as SuperScan (http://www.foundstone.com/rdlabs/proddesc/superscan.html) all but begging attackers to attempt to break through using documented vulnerabilities. Because it is so easily identifiable, it is clearly a risk having VNC Server on the default port running all the time.

Authentication

The authentication scheme for VNC is inherently weak. Looking at the registry entries created during the installation, the encrypted password is stored in the newly created:

HKLM\SOFTWARE\ORL\WinVNC3\Default\Password.

This area of the registry is by default quite insecure in some unpatched operating systems, such Windows NT 4. The default parent key in WNT4 gives Administrator and SYSTEM full control and Everybody has Special Access (Read and Modify). This newly created child key inherits these properties. Because of this, almost anyone can view and modify the encrypted password to allow unauthorized server access to an attacker. For example, an attacker could edit the registry, set a null password and delete another value to enable unauthenticated access to the server (see Authentication, below) The attacker would then have complete control of the WinVNC service.

Network access to the registry is a little more difficult in Windows 2000 and Windows NT 4 with the registry permissions patch applied. But, as many companies migrate to Windows 2000 without Active Directory in place, a flawed method for solving interim registry problems is to grant all users Power User or Administrator status on the local computer. If using WinVNC and Windows 2000 in this manner, all users would have access to the registry and thus the server password. This may not be appropriate if the implemented password scheme is one in which all VNC servers use the same password. (This is discussed later) Ideally, only Administrators should have access to these keys using this scheme. If nothing else, remove the "Standard Users" and "Everybody" from the permission on the HKLM\Software\ORL\WinVNC key.

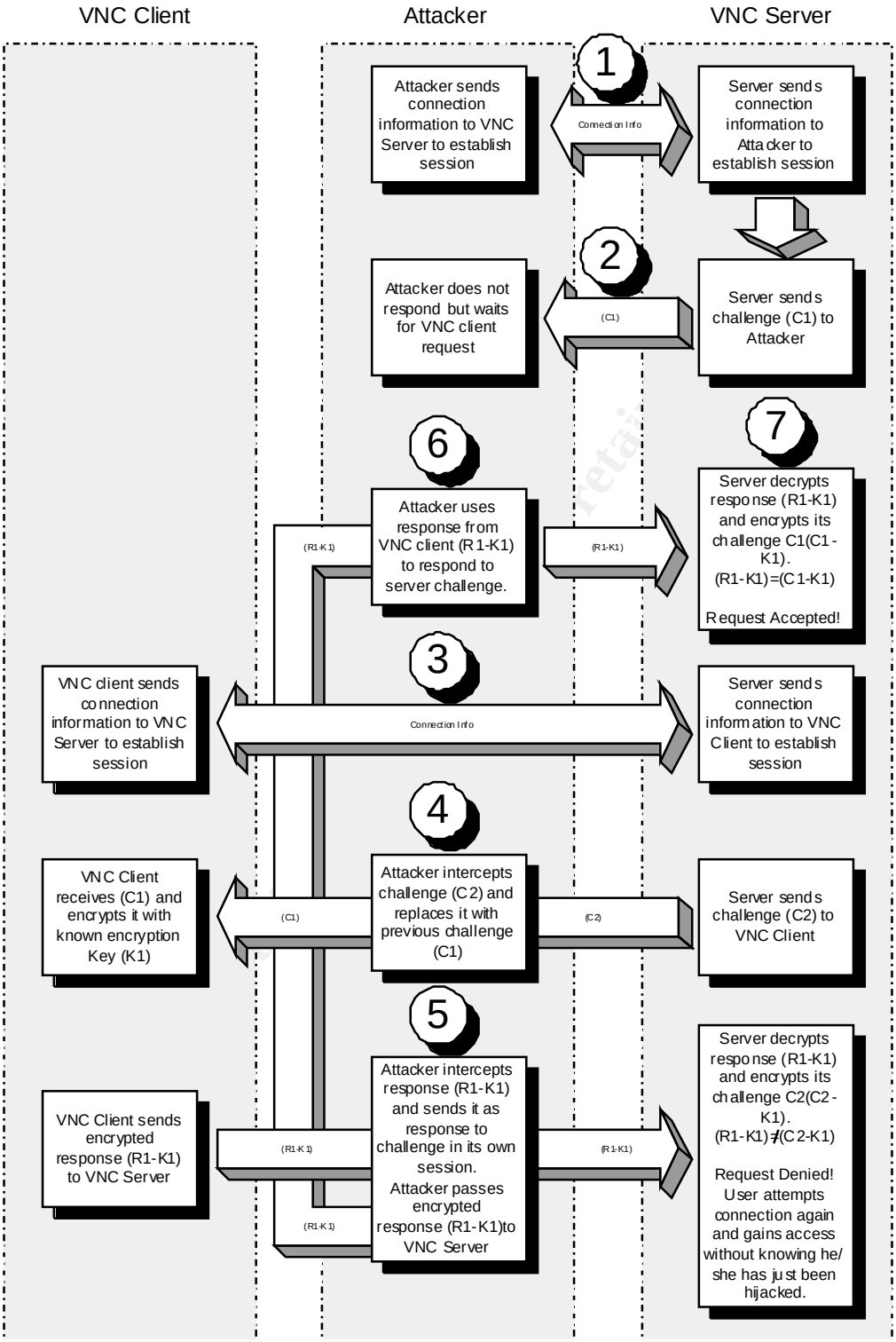

Figure 1 – "Man-in-the-middle" attack on VNC

VNC uses the MD5 challenge-response authentication method, and the server must be able to decrypt the password. It is therefore not hashed. The password is encrypted using 3DES (good encryption) and stored in the registry. Unfortunately it is encrypted with a

fixed key (23 82 107 6 35 78 88 7) whenever a password is saved. Source code is available to extract the password from the key in seconds (http://packetstorm.securify.com/Crackers/vncdec.c).

Armed with the fixed encryption key, and access to the machine's registry, one can easily grab the encrypted key from the registry, and at his/her leisure decrypt the password and use it to compromise the VNC server.

Another method for defeating earlier versions of WinVNC (prior to V3.3.3R6) is to use a simple dictionary attack against the VNC server from a modified VNCviewer. This patch, available from http://www.securiteam.com/tools/Brute_forcing_VNC_passwords.html, can be applied to a Unix VNC viewer and will create a viewer that will guess passwords until it succeeds or until the dictionary wordlist file is exhausted. Worst of all, you will not even know this is happening, as VNC does not log failed login attempts and the server provides no sign that it is being breached. The best method of protecting against this type of attack is to enforce strong passwords that are at least 8 characters in length and do not reside in the dictionary.

It is imperative to not only know what you are protecting, but also whom or what you are protecting *against* and tailor the security for the threat. As with any remote control solution, VNC has both internal and external security concerns. The external or 'outside attack' is usually the focus of the majority of the security effort and is often thought, albeit naively, to be entirely controllable by firewalls and routers. VNC can be attacked from an outside source as if it were an internal attack if it is installed on a server directly connected to and used over the Internet. The internal attack is commonly viewed as a 'management issue' and is sometimes erroneously overlooked even though current surveys estimate that up to 80 percent of computer attacks come from the insider[1]. As a tool created to assist the support and maintenance of the end-user's desktop PC, and having it installed on the end-user's PC with an icon in the system tray, it is not uncommon for VNC to be seen as a 'Big Brother' tactic. Because of this, the internal attack may become your most significant problem. Although current WinVNC releases' icon changes color from white to black when in use by a client VNC user, earlier versions allow running the service transparent to the end user. Using the transparent tactic clearly ratifies the perception of snooping and may give credence to attempts at disabling it. Even the authors of VNC attest that they can find no legitimate reason for NOT displaying the server status via the system tray icon and have included this in their source code for that very reason. Malicious users, on the other hand, revel in the fact that prior versions do not display the default system tray icon. These are the versions of choice for 'stealthy' remote control. The end-user may attempt to disable or manipulate the service to stop what they perceive as unwarranted surveillance. It is therefore essential to fully explain to the end user, what role VNC, or any remote control software, will play in your environment. Clear policy should dictate the proper use of VNC (end-user will be contacted prior to its use), how to detect if it is in use (icon color change), and when to contact the security department if they feel their machine is being controlled without their consent. (See *VNC – A Call Centre Perspective* for additional policy). Because many

[1] Computer Security Institute, 2001

machines may have the same password, direct policy and education can assist in determining the breach of a VNC password and the breadth of its reach.

Several methods of password implementation are available. Some offices implement a single password scheme for all VNC 'servers', giving full control of all computers on the LAN after a single password is cracked. It makes perfect sense in this case, to instruct the user to assist in identifying any breach of security, as methods of differentiating between good and malicious VNC use on the network is not easily discerned. On the other hand, some may choose to allow the user to select the password and only reveal it to the systems administrator upon request. Doing so can introduce a security hole that is easily exploited using timeless social engineering tactics. One method of compromise is to utilize host perimeter protection, such as ZoneAlarm (http://www.zonealarm.com/) or BlackIce Defender (www.networkice.com) in conjunction with a global password to allow the user to accept or deny access to their computer based on incoming port or host number.

Native Countermeasures

To counteract the manipulation of the server settings, and mitigate the 'internal attack', VNC includes features to deny the end-user from changing user settings, including the password, whether authorization is required, whether the icon appears on the desktop, whether they can disable the service and even whether they can access the properties at all.

Looking at the installation changes, we see the default registry keys and values added by the default install (HKLM\SOFTWARE\ORL\WinVNC3\Default). Several helpful keys are left from the install. By adding these keys/values, VNC becomes much less vulnerable to internal attack to the server configuration.

AllowProperties	Adding this key with a value of 0 (zero) locks the user from manipulating the settings, including the password through the VNC icon.
AllowShutDown	Adding this key with a value of 0 (zero) denies the user from stopping the server service and closing down VNC.
AuthRequired	The default installation of WinVNC will not accept incoming connections unless a non-null password is set in the registry on the server computer. Adding this key with a value of 0 (zero) allow connections without having a server password set. Without protection (default), the location of this registry key can allow an attacker with little to no authority to change this value and obtain access equal to the current server session. Set permissions on the server computer to disallow anyone but administrators to view or change this area of the registry.

AuthHosts	This setting is used to specify an IP address template which all incoming connections must match in order for the server to accept it. By default, all connections are accepted. The values are in the form of: +[aaa.bbb.xxx.xxx] ?[aaa.bbb.xxx.xxx] -[aaa.bbb.xxx.xxx] with the first two octets serve as the filter range and the prefix (+/?/-) corresponding to the QuerySetting settings.
QuerySetting	This key sets behavior when encountering a host request defined by the AuthHosts value. The default value of 2 is: + :Accept, ?:Query, ?:Reject Values range from 0-4 with a greater degree of 'paranoia' associated with an increasing value.
Idle Timeout	This value determines the amount of time the VNC client can remain idle before being automatically disconnected. The value is in seconds. This value should be set to mitigate the probability of someone remotely initiating a session and inadvertently leaving the client connected for long periods of time thus enabling his or her connection to be hijacked.
LockSetting	This value determines the behavior of the server after the client disconnects. To prevent a user from disconnecting from the server and leaving the server session logged in with privileged access, it is advisable with any remote control software to have the server log off the user upon disconnect. Although, as of this writing, WinVNC does not support locking the workstation on disconnect, it does allow a logoff on disconnect. To set this, set the value to 2. Set to 0 to disable logoff.

Protection after authentication

Once identified, VNC traffic can reveal a myriad of information to a network sniffer. We discussed the need for protecting the service password and its settings, but what information does WinVNC pass after authentication? All transmission through WinVNC is compressed, but not encrypted, meaning that all data including account information that is typed into a login dialog

box, for instance, is fair game to a snooper. This is one of the greatest vulnerabilities of WinVNC. Fortunately, using some form of 'tunneling' technology can alleviate this.

Several methods of tunneling packets are available for Windows. AT&T provides information on protecting VNC traffic through SSH (http://www.uk.research.att.com/vnc/sshwin.html) for a Windows client to a Unix server. A great source of SSH info is available from http://www.employees.org/~satch/ssh/faq/ssh-faq.html, which includes a link to all of the current ports of SSH to the Windows platform. A few quick observations are:

- SSH will NOT protect against attacks to the host machines. So, it will not assist in protecting the WinVNC password. It only protects the communication between the hosts. Once an attacker gains administrative privilege, SSH can be subverted too.
- Two versions of SSH are available. SSH1 is no longer being developed. SSH2 is the current standard.
- Use of SSH1 is still vulnerable to the 'man-in-the-middle' attack.
- SSH1 has more options for authentication including Kerberos.
- SSH1 is faster than SSH2.

Typical Remote Control Vulnerabilities

All remote control applications suffer from some form of security vulnerability. VNC contains several of these as inherent flaws. Figure 2 summarizes typical remote control vulnerabilities and how they are addressed in WinVNC.

Password enable	Password must be defined on the server by default. This behavior can be overridden by the AuthRequired Value.
Strong Password	No innate password auditing available. Subject to dictionary attack, and manipulation because of the poor location of encrypted password in registry. No encryption of transmission after authentication leaves other information in clear text. Passwords can be any length but are truncated to 8 characters (i.e secretpassword = secretpa) after encryption.
Alternate Authentication	Provides an alternate means of authentication than NT authentication on Windows systems, although it suffers from weak fixed key encryption and user modifiable settings to disable authentication. IP rules can be established to allow/query/disallow incoming session requests from specific IP ranges.
Password protect Profile Files and	No user specific profiles used. Setup files by default have no permissions set. Files needed for service setup also not

Setup Files	protected by default.
Logoff User upon Session Completion	User logoff enabled through the use of LockSetting value. Disabled by default. Lock user on disconnect documented but not functional in latest release.
Encrypt Session Traffic	No encryption used after authentication. Must use SSH or similar to encrypt session.
Limit Login Attempts	No method for limiting login attempt.
Log Failed login Attempt	No log creation for failed login attempts.
Lockout User After Failed Attempt	No lockout feature after failed login attempts. Latest release incorporates a login time delay to stall a dictionary/brute force attack.
"Obscure Security"	Ports can be changed to make VNC more obscure to casual port scans.
End-user security	Supports the ability to lock the end-user from manipulating server settings and shutting down the service.
Granularity	One size fits all. All users have the same abilities granted by the server. No method for creating profiles for specific user groups.

Figure 2 - How VNC handles typical remote control vulnerabilities

Current Vulnerabilities (V3.3.3)

Buffer overflow conditions exist in both of WinVNC's core components, the server and viewer components. Specially crafted packets directed at these components, could allow execute arbitrary commands with the privilege level of the user logged in at the server. (http://www.securiteam.com/windowsntfocus/ATT_VNC_Windows_Server_buffer_overflow.html). Worse yet, a maliciously controlled server could send a specially crafted packet that spoofs the server version and challenge response to the VNCviewer's request for authorization and run arbitrary code on the client's computer. This attack is extremely important as "it might imply the escalation of an attack from a less secured network environment to a more secured network environment."[2] A technical description including a patch is available from http://www.securiteam.com/windowsntfocus/ATT_VNC_Windows_Client_buffer_overflow.html.

A few final words

[2] Beyond Security Ltd., ATT VNC Windows Client buffer overflow (2001)

This article could not conclude without a word or two on basic network security. The practice of "Principle of Least Privilege" and basic network security layering is not an option when using WinVNC. A client attacker will interact with the server at the level of the individual logged on the machine at the time of the creation of a session. Therefore, strong networks passwords and the practice of locking or logging off the computer when not at the console are a must. In fact password use should be enforced at all times. If WinVNC is used on servers (not recommended) and the servers are left logged in with an administrator password (common practice in some environments), you have effectively created a huge hole in your security with the possibility of privileged remote access to the network. Additionally there is no reason for installing the viewer component on all computers in the organization (default behavior). Although the viewer is readily available on the Internet, giving this to the users is akin to hanging a lock pick outside your locked door.

A strong, layered approach in conjunction with native controls in VNC will ensure the securest of installations for VNC. Controls such as limiting access to registry editing tools through policies, and setting permissions on critical WinVNC components necessary for the VNC service to run should be exercised. Installing VNC only on computers that *require* it also decreases the chance of multiplying the vulnerabilities. The inability to detect the breach of a VNC server all but dictates this approach.

In all, WinVNC appears to be a good, quick solution to remote control, but its default security settings and weak authentication along with its unencrypted transmission raise serious security concerns. Without combining WinVNC with other methods of encryption, like SSH2, you might find that it falls short of your expectations for a secure remote control application.

References

Steve Acheson. "The Secure Shell™ Frequently Asked Questions" URL: http://www.employees.org/~satch/ssh/faq/ssh-faq.html (16 Feb, 2001)

AT&T Laboratories Cambridge. "Virtual Network Computing – Making VNC more secure using SSH" URL: http://www.uk.research.att.com/vnc/sshvnc.html (1999)

AT&T Laboratories Cambridge. "Virtual Network Computing – WinVNC – The Windows NT VNC server" URL: http://www.uk.research.att.com/vnc/winvnc.html (1999)

AT&T Laboratories Cambridge. "Virtual Network Computing – Windows VNC release history" URL: http://www.uk.research.att.com/vnc/winhistory.html (1999)

Beyond Security Ltd. "ATT VNC Windows Client buffer overflow" URL: http://www.securiteam.com/windowsntfocus/ATT_VNC_Windows_Client_buffer_overflow.html (30 Jan, 2001)

Beyond Security Ltd. "ATT VNC Windows Server buffer overflow" URL: http://www.securiteam.com/windowsntfocus/ATT_VNC_Windows_Server_buffer_overflow.html (30 Jan, 2001)

Carnegie Mellon University, "Vulnerability Note VU#197477 – AT&T WinVNC allows user access to passwords and configuration via weak registry permissions" URL: http://www.kb.cert.org/vuls/id/197477 (25 May, 2001)

Computer Security Institute, "2001 Computer Crime and Security Survey" San Francisco, 2001.

Déraps, Perry. "VNC – A Call Centre Perspective" URL: http://www.sans.org/infosecFAQ/win/VNC.html (2 Nov, 2000)

Internet Security Systems, "vnc-installed-noauth(1988)" URL:http://xforce.iss.net/static/1988.php (2000)

Internet Security Systems, "WinVNC client rfbConnFailed reason string buffer overflow" URL:http://xforce.iss.net/static/6025.php (2001)

Scambray, Joel, McClure, Stuart, Kurtz, George. Hacking Exposed – Network Security Secrets and Solutions. Berkeley:Osborne/McGraw Hill, 2000. 511-527.

Security Focus.com. "AT&T WinVNC Remote Desktop Default Configuration Vulnerability" URL: http://www.securityfocus.com/vdb/bottom.html?vid=1961 (2001)