TECHNICAL MEMORANDUM

To: General Distribution

Project: General Information

From: David M. Ihnat

CC:

Date: 29 Jul 11 Last Modified: 31 Jul 11

Subject: Care and Feeding of Standalone Windows Workstations

---

1. INTRODUCTION

In most business environments using Microsoft Windows, one or more servers running Windows Server provide centralized administration and management for the Windows-based workstations in the network. The server provides a number of services, such as centralized user authentication, policies that govern how the workstations are configured and updated, and shared data storage that can be protected using user authentication and permissions.

However, some organizations can't, for various reasons, make use of a Windows Server-based environment; this is referred to as a *standalone environment*. Windows workstations in a standalone environment pose significantly different support and maintenance challenges; many of the tasks that can be automated through a server fall on the shoulders of the individuals using the workstations. Sharing data in such an environment requires coordination and cooperation of all involved users. Maintenance of the Windows operating system and its environment must be carried out on a per-machine basis, again, usually by the user of that machine.

This memorandum provides a guideline for configuration, maintenance, and day-to-day operation of such standalone workstations in a small business or organization.

2. OVERVIEW

This document will discuss issues in the following categories and order:

- *Assumptions*
- *Operating System Configuration*
- *Networking*
- *Data Sharing and Protection*
- *Maintenance*

3. ASSUMPTIONS

There are some assumptions made in the following guidelines:

- The computers are on a common network
- There is a broadband Internet connection of some sort—DSL, Cable, T1—that provides a reasonably capable gateway, even if not a separate firewall appliance.
- The Windows desktop operating system is one of Windows XP SP3, Vista, or Windows 7.

4. OPERATING SYSTEM CONFIGURATION

All computers must have a *computer name*. Avoid using transient names, e.g., *JoesWorkstation*. Over time, people come and go, and computers are repurposed; such specific names get confusing a couple of years out. Use a generic name such as *WKSTN001*.

Workstations on a server-less network have a common identification that describes that they're part of the same organization; this is the WORKGROUP setting in the network setup. The default from Microsoft is usually just *WORKGROUP*; on some versions it may be something like *MSHOME*. In any case, you're going to want to change this to something that reflects your organization, e.g., *XYZCORPGROUP*. Set it to the same value on every workstation in your organization.

No matter the operating system, it must have accounts created for the user of the machine. Many Windows distributions attempt to make things "easier" for the end user by creating a generic user—often something like "Owner" or "User". These should **not** be used. Create a local account for every user who will be on that workstation. Moreover, create one for other users in the office or organization who may need access to the machine, either as a direct login or to share data.

If you have the choice, never use a "Home" version of Windows in a business environment—even without a server, the "Professional" version (whether it be Windows XP, Vista, or Windows 7) offers options that are much better suited to managing the machine in a business network.

Finally, if at *all* possible, don't use Windows Vista. This is a dead operating system—it wasn't accepted well at all in the marketplace, and Microsoft has effectively set it to "End Of Life". If you're still using Windows XP, you'll be OK until 2014, but be aware that it's also near its official "End Of Life", not to mention it is more vulnerable to malware ("malicious software") than Windows 7.

5. NETWORKING

All computers on a network must have a unique Internet Protocol, or *IP*, address. This can be explicitly assigned as a permanent value, and in fact must be if there's nothing on the network that can automatically assign such addresses; this is called a *static IP address*. This is the least desirable configuration—it makes management more difficult, and requires coordination and documentation to keep everything in order.

It can also be assigned by a *DHCP Server* (DHCP stands for Dynamic Host Configuration Protocol). On a network with a Windows Server, of course, that machine (or machines) fills this role; on standalone networks, it's often configured on the gateway DSL modem, cable modem, an intelligent switch, or a hardware firewall.

5.1 ADDRESSING

In all cases, use a numbering scheme that's in order with *RFC1918*. RFC (Request for Comment) documents are the underpinning of the standards that define the Internet; RFC1918 refers to a document that specifies private IP addresses for internal networks (If you really want to know the technical details and options available to you, see Appendix I).

While there is a range of such addresses, it's probably most sensible to use one from the so-called "Class C" range. If you've no other guidelines or requirements, I suggest you select 192.168.100 as the network; machines will be assigned from 192.168.100.1 through 192.168.100.254, with a Netmask of 255.255.255.0.

Conventional use is:

192.168.100.1 :	Gateway. Your firewall, DSL or cable modem, etc.
192.168.100.2-9:	Special networking devices (e.g., switches)
192.168.100.10-19:	Servers
192.168.100.20-99:	Printers and other network devices
192.168.100.100-199:	Computers on the network
192.168.100.200-254:	Special use. VPN addresses, temporary devices.

5.1.1 Static Assignment

Using the guidelines above, make sure to assign addresses to each device and record the address assigned.

5.1.2 DHCP Assignment

Find out what device on your network is capable of assigning DHCP addresses. If you control the network, configure the device to assign addresses from 192.168.100.100-199.

If you don't control the network, make sure whoever manages it knows about your systems, and is OK with them getting addresses from their DHCP server.

5.2 GATEWAY

Every network needs a *gateway* as part of the networking configuration. This refers to a device that can tell your computer how to find other networks—for instance, if you use DSL or Cable, it will be the network address of the modem. This will be assigned automatically if you use DHCP; it must be configured manually for static addresses.

5.2.1 DNS Servers

A *DNS Server* is a computer that knows how to translate names to IP addresses—for instance, when you enter *www.google.com* into your browser, the DNS Servers will figure out the real Internet address for Google. (DNS stands for Domain Name System). Again, if you're using a DHCP server, this will be assigned automatically; if you're doing static addresses, you'll have to know the address of at least one DNS server.

Often this will be the address of the Gateway; however, sometimes you may have to put in the two DNS servers specified by your Internet Service Provider.

6. DATA SHARING AND PROTECTION

Even without a server, it's possible to share data between cooperating Windows workstations on a LAN; it's just more tedious, since you, the users, are responsible for manually setting up the mechanisms that allow such sharing.

6.1 WORKGROUP IDENTIFICATION

As mentioned previously, workstations that are in the same organization on a LAN should all have the same WORKGROUP setting. Select a name that reflects your organization, but don't make it too long; no spaces or special characters. Make sure it's exactly the same on every workstation.

6.2 USER ACCOUNTS

Every user of a Windows computer has a *login* that uniquely identifies them to the computer. On a network with a Windows Server, this login is actually stored and managed on the server, and workstations ask the server if the login and password are valid.

Without a server, each computer is responsible for "knowing" the logins of individuals who can access that computer. How you create and manage these accounts depends on the version of Windows you're running; in all cases, start looking in the Control Panel. If you've a Professional version of Windows XP, Vista, or Windows 7, look in *Computer Management* for "Local Users and Groups". On home versions of the various operating systems, you may only be offered a *Users* applet in the Control Panel.

As mentioned previously, many Windows versions from vendors may have a single default account created, often something like "Owner". Don't use this. It doesn't describe who the user is, and doesn't allow separate identities that are important in a business environment.

Always assign a password to an account; if you want to do any data sharing across the network, it's required, and it's a bad idea to allow anyone to be able to just log into your account by knowing your login name (or, worse, by clicking on it if you've selected to show user names on the login screen.)

6.2.1 Administrator Permissions

This is a difficult issue on standalone workstations. It is possible to create accounts that have only user-level permissions, in order to protect the system configuration, and/or to protect data. Running as a normal user also provides more protection against malware—if you don't have permissions to modify the operating system, or install programs, neither will the malware.

However, there are very many normal tasks on a Windows workstation that require administrative permissions, such as adding printers, or installing legitimate programs. You essentially have two choices in how to manage this:

- Create two accounts for people who have the right to administer the computer—one that has Administrative permissions, and one that's the day-to-day login for normal work. This allow you to grant and restrict rights to individuals, but "costs" you in terms of having to manage two accounts for the privileged users, and in the annoyance to users of having to log out of their "normal" account and into the privileged account to make system changes.
- Give everyone administrative rights. This is, unfortunately, the model that most standalone computer networks use, simply because it's easier to work with on a day-to-

day basis. It also exposes the workstation to more types of attacks by malware, as well as simple errors by the users.

Clearly, it's preferable to use the first configuration; equally clearly, many existing workstations on LANs will be set up using the second model.

6.3 USER GROUPS

On professional versions of Windows, it's possible to create what are known as *groups* in "Local Users and Groups". There will be a number of groups already defined for system use; a good convention is to make any group names you create for your organization all capitals, e.g., ADMINGROUP. You can then add individual user logins to the group membership.

These groups are particularly useful when setting permissions to shared data, since you then don't have to explicitly define every user's name when setting access rights—just the group name.

If you decide to use groups, create the same group name on every workstation.

6.4 SETTING PERMISSIONS

It's possible to control permissions for information you share from a workstation, but it's both more tedious than doing it from a server, and less secure.

There are specific steps you must take to share information on the network:

1. *Share Permissions*. You share directories from your hard drive. Any such shared directory has a "share name"—which doesn't have to be the same as the directory name—and "share permissions", in which you define the user login or group that is permitted to access that share on the network.

   Share permissions only define what users or groups on the computer can even see or get to a shared resource; what they can do to the data is controlled by filesystem permissions.

2. *Filesystem Permissions*. Filesystem Permissions define in detail what can be done with the files and folders in the directory.

   On the same computer, you can add or remove permissions for accounts and groups on that computer.

   When allowing users or groups from other computers to access data, they have to specify their login and password on *this* computer to gain access to the share.

Let's show this using an example. Assume there are three computers in the LAN, and we're going to control access by using groups:

Computer	Groups	Logins
WKSTN01	ADMINGROUP USERGROUP	joe (in both groups) sherry (USERGROUP only)
WKSTN02	ADMINGROUP USERGROUP	joe (in both groups) sherry (USERGROUP only)
WKSTN03	ADMINGROUP	joe (in both groups)

Computer	Groups	Logins
	USERGROUP	

Table 1- Example LAN

We're assuming that WKSTN03 is only used by office administrators—only *joe* in this simple network—so it's OK in this case not to give *sherry* a login on that machine.

Further, let's assume there are two directories on each computer called *C:\Data\CorpData*, and *C:\Data\AdminData*; we only want people in the USERGROUP to get to the first, and only people in the ADMINGROUP to get to the second. Finally, we want every computer to be able to get to every other computer's shared data.

On **each computer**, you would create the directories, and define two shares:

Directory	Share Name	Permissions
C:\Data\CorpData	CorpData	USERGROUP: Change,Read ADMINGROUP: Full Control
C:\Data\AdminData	AdminData	ADMINGROUP: Full Control

Table 2 - Share Permissions

So what have we said here? Simply that everyone in USERGROUP can read or modify information in CorpData, but they don't have control over permissions. ADMINGROUP members can do anything to both shared resources.

Now how about filesystem permissions? On **each computer**, you would add both groups under the "Security" tab for the directory's "Properties". You would then give ADMINGROUP "Full Control" here as well; for USERGROUP, you would only allow "Read & Execute", "List Folder Contents", "Read", and "Write" from the checkboxes in the security dialog.

6.5 ACCESSING SHARED DATA

Once all the above is set up, how do you get to the data on another computer? "Map Network Drive" is the most commonly used method. When mapping the drive, however, you'll have to give *your* login and password on the machine you're connecting to. (On Windows XP, this is a highlighted link called "Connect using a different user name". On Vista and Windows 7, there's a checkbox for "Connect using different credentials".)

As an example, *joe* wants to connect to the *UserData* shared from WKSTN02 while using WKSTN01. Select the machine and share by browsing, but also remember to change your credentials (this is slightly different between Windows XP and Windows Vista or 7, but it's a choice in the map drive dialog). Joe must make sure to use the name "WKSTN02\joe", not just his login name, and his password for his *joe* account on WKSTN02.

On all operating systems, there's also a choice to "reconnect

7. MAINTENANCE

Once you have your system and shared data set up, you'll want to carry out regular maintenance tasks. Some can be automated; others need to be carried out by the workstation user on a regular basis.

7.1 WINDOWS UPDATES

Windows updates are critical. There was once a time it was recommended not to apply all the updates from Microsoft, since they often "broke" the system—let someone else try them for a while, then update. Maybe.

This is no longer the case—the attacks by malware have gotten so fast and furious, and Microsoft *has* gotten so much better at issuing updates that don't break the system, that the recommendation now is to allow daily updates.

On a Server-based network, workstations commonly get their updates from the server, which is the only machine that actually collects the updates from Microsoft.

Clearly, this isn't available on a standalone network, so you should configure your machine to update daily and apply the updates automatically. This is done from "Automatic Updates" for Windows XP, and "Windows Update" for Vista/Windows 7, both in the *Control Panel*. You might want to move the time away from the default of 03:00 AM—after all, every computer that stays with the default (at least from your time zone) will be going out at the same time.

The process, once configured, is automatic; you may sometimes find your workstation has restarted overnight, if the update requires it.

However, you should run a manual update session at least once or twice a month just to make sure everything is being updated properly.

7.2 DATA BACKUP

Again, on a network with a Windows Server this is usually taken care of automatically by a regular backup of the data on the server to tape or (more commonly today) disk, or even to an Internet backup. (In such an environment, no data is ever saved on a workstation—*everything* is saved on the server.)

Standalone workstations don't have this luxury, so some provision must be made for each workstation that has data that should be saved.

7.2.1 NAS Data Storage

Perhaps the best solution is to get a Network Attached Storage device (NAS). These are boxes that are "mini-servers"—they look like another computer on the network with shares, but that's all they do; a good example is *Buffalo Technologies' TeraStation* line of NAS devices. These are far cheaper than a server—you can get a NAS unit with 2GB of storage for a few hundred dollars.

If you can afford it, it's best to have two of these, with one accepting all the data you want to save and automatically "mirroring" it to the second unit as a backup However, if you can't afford two units, at least you do have two copies of everything—one on the original workstation(s), and the other on the NAS unit.

The major disadvantage of this approach is that all the data is in the office; should some disaster befall the office—fire or flood, for instance—the NAS unit(s) may well be destroyed along with the workstations.

7.2.2 Removable Hard Drives

Very large removable hard drives are available for very little money, and have the added advantage that they can be carried off-site for backup protection. The disadvantage, of course, is that each workstation user—or an office administrator—has to remember to make sure the drive is connected and a backup is actually carried out to it on a regular basis. Again, it's a very good

idea to have at least two such drives—one is kept in the office, one taken off-site, and the two are rotated regularly.

It's a good idea to get a fire-resistant safe for the office, and keep the hard drives in the safe when not actually in use.

Most of these drives come with their own software—almost invariably, it's garbage. Get a copy of the free utility *Easeus ToDo Backup Free*, and schedule it to do automatic backups.

How often do you need to backup your data? That's totally dependent on your needs. The best question to ask yourself is, "If I lost all the work I did for one day, would I be extremely upset?" If the answer is yes, you need daily backups. If not, start extending the time—2 days, a week, two weeks. When you hit a time period that would be unacceptable, you've defined your backup frequency.

7.3 MALWARE PROTECTION

It's an unfortunate fact of life that there is a *lot* of malware out there, and there's only more every day. Worse, it's getting extremely sophisticated, and the authors are getting very good at figuring out how to get it onto your computers.

Even worse, no single program or package will provide full, complete, comprehensive protection. The kinds of malware are too diverse, and too well crafted. Instead, the goal is to provide a *layered defense*, and to prepare the system for cleanup in the case something does get through the defenses.

For individuals and charitable organizations, much if not all of the software available for protection is free. For small businesses or in cases where it's not free for charitable organizations, licenses are quite affordable.

Again, in a Server-based environment, there are centralized packages that allow the administrator to monitor and take care of machines from the server; and, again, in a standalone environment the workstation user is responsible for this.

7.3.1 Categories

These categories define anti-malware software by the kind of protection they offer. In each case, I'm going to recommend a particular package—and it's almost certain that if you were to ask someone else, they might suggest a different solution. There are a huge number of such software packages available; my suggestions come from my own experiences in supporting dozens of small business and charitable clients over the years.

7.3.1.1 Anti-Virus

An anti-virus package provides the core of your defense. A good package performs dynamic scanning of files and E-Mail while you're using the machine in an attempt to identify malware as it comes into the system. It will also provide the option to run deep scans of each hard drive on the system on a scheduled basis.

Many of the commercial anti-virus packages have become entire "suites", purporting to offer firewall, mail scanning, anti-spam, and a host of other "features". Unfortunately, many of these packages, as shown in comparative studies, have fallen behind in the main job of anti-virus protection software—detection of malware. For this reason, I've come to avoid such standard names as Symantec and McAfee.

One of the best anti-virus packages for free use by individuals is *Antivir*. Unfortunately, it does require a license for business use. Other good choices are *AVG* and *Nod32*; both have similar requirements.

For small organizations, *Sophos Anti-Virus* is one of the best choices for licensed anti-virus software. It's quite inexpensive, and is "lightweight" in its effect on machine performance. *Kaspersky* is a perennial top-rated package, but it tends to require relatively new workstations and can have a noticeable effect on performance.

No matter what package is selected, it should be configured to update at least daily, and to carry out a full scan at least every night. The user should open and check it at least once a week to see what it may have found, and to resolve any issues, empty the Quarantine, etc.

7.3.1.2 Inoculator

Malware authors don't want to waste their time re-infecting machine they've already compromised. For this reason, much of the malware circulating will drop an indicator on the infected system—it may be a Registry entry, or a small file hidden somewhere. These indicators, themselves, are harmless. When the malware first gets a chance to run on the system, it'll check to see if it's already there.

An inoculator tries to "game" the malware by creating its own indicators. A very good free inoculator is *SpywareBlaster*. The free version will require the user to manually update the software; this should be done every couple of weeks.

7.3.1.3 Registry Monitor

The Registry is a special database used by Windows to keep track of, well, everything. Installed software, program and system settings—literally everything defining the Windows system and its configuration is stored in the Registry. And, of course, it's one of the first things a piece of malware has to try to modify to get into the system.

A Registry monitor watches for requests to change the Registry, and pops up an alert for the user, who can then decide whether or not to allow the change. This can be confusing to users who aren't familiar with the terms used, so a simple rule can be followed: If you're not adding, updating, or removing programs, and not changing settings in a program, suspect the request and deny it.

A program called *Spybot Search & Destroy* has a component called the *TeaTimer* which serves as a very decent Registry Monitor. (It also is a *Scanner*—see the next section.)

Spybot can be configured to automatically update itself on a regular basis.

7.3.1.4 Scanners

Scanners are, generally, offer a combination of an anti-virus scan and an on-demand scan when you suspect you have a problem. These should be installed and kept up-to-date.

I recommend that at least two be installed—both are free. *Spybot Search & Destroy* can be configured to automatically scan; it should be scheduled to run at least weekly, if not nightly.

MalwareBytes is one of the best scanners available, also free for personal use—I believe it's also free for charitable use, but this should be checked as it can change. If not, licenses are quite inexpensive, and it's worth having on a system in case problems arise.

It must be updated manually, at least the free version; this should be done weekly.

8. CONCLUSION

There is no doubt that a Server-based Windows network is the best solution to implementing and managing a Windows-based LAN. However, in the case that it's not possible to support a server, the configurations and practices described in this document should allow reasonable management of a serverless LAN.

==EOD

Appendix I
RFC1918 Addresses

Every computer connected to a TCP/IP network must have an *Internet Protocol* address. Computers that are "known" to the Internet at large must have public addresses, assigned by IANA (Internet Assigned Numbers Authority). Usually, these are assigned in blocks; most people don't get their addresses directly from IANA, but rather have an Internet Service Provider (ISP) who's been assigned large blocks of addresses, and who assigns one or more addresses to its customers.

Since an IP address (technically, IPv4, or "Internet Protocol Version 4") fits in a 32-bit computer word, there are a large number of possible addresses—4,294,967,296 addresses, if all could be assigned. That sounds like a lot—but it's not.

Early on it was realized that the burgeoning number of private devices on local area networks (LANs) would quickly deplete even this large pool of addresses, so computer scientists devised two mechanisms to help prevent this—Private Internet Addresses, and Network Address Translation (NAT) to help use these private addresses.

Simply put, the document RFC1918 (see http://tools.ietf.org/html/rfc1918) defines reserved addresses that are only to be used on private networks. Networking equipment will recognize these addresses, and never let them be seen outside the local area networks. These addresses are in the following ranges:

Start Address	End Address
10.0.0.0	10.255.255.255
172.16.0.0	172.31.255.255
192.168.0.0	192.168.255.255

Anyone can select an internal network addressing scheme that uses these addresses without having to acquire approval from any governing body. All those local workstations, printers, and other gear on your local network that needs an IP address can now have one, and it won't use up precious external addresses, and won't expose your equipment to the outside world.

But this brings up another problem—these machines almost always need to get to the Internet for browsing, E-Mail, and the myriad other uses that have sprung up as the Internet has evolved. How to do this with these private addresses?

Enter Network Address Translation (NAT). The idea is simple in concept—if a computer with a private address wants to access the Internet, it does so through a *gateway* that has a "real" external address. The gateway will package all requests from computers on the LAN as if they're coming from that single external address, and when any response comes back from the Internet, knows how to direct it to the local computer that sent the request. Another document, RFC1631 (see http://www.ietf.org/rfc/rfc1631.txt) defines how to do this.

Manufacturers of DSL and cable modems, firewall appliances, and other devices that act as gateways between the LAN and the Internet "know" about LANs, and carry out this address translation automatically.

It should be noted that we're *still* running out of "real" addresses—in fact, as of the summer of 2011, technically have run out, as all the unassigned addresses possible for IPv4 have been allocated. A new addressing scheme called *IPv6* has been designed and implemented, but has not yet gained widespread acceptance as of the time of authorship of this paper. The biggest problem to acceptance of the new scheme is that literally—not virtually—every device directly connected to the Internet must "understand" the new address model. This is a tremendous expense in replacing existing devices, reprogramming computer operating systems, verifying security of software and firmware, etc. It must happen, but will be a long process. Consideration of IPv6 is far beyond the scope of this paper.