

## The NT Local Administrator and Shared Passwords

*Daniel Marvin* 2001-04-02

### The NT Local Administrator and Shared Passwords

by *Daniel Marvin*

last updated April 2, 2001

---

There is a Local Administrator account on every NT machine currently deployed. This account can be renamed, but not removed. It is extremely common to find many NT machines in an enterprise sharing the same password for this Local Administrator account. This article will establish that this shared password constitutes a security vulnerability, discuss various steps to mitigate the risk arising from the shared password, and make a case for applying unique passwords to every Local Administrator account in your enterprise.

## The Security Issue of Shared Local Administrator Passwords

Workstations share the same Local Administrator password for a number of reasons. First and foremost, a shared password eases the daily burden of support personnel. No desktop support staff person wants to carry around a massive list of passwords or go through the cumbersome process of querying a centralized database of passwords. Secondly, automated build processes, which are very common, result in the deployment of a shared password. This is because disk imaging software ensures that each cloned machine has the same Local Administrator password as the original, and scripted installations reapply the same Local Administrator password each time the script is executed.

So how does a shared Local Administrator password constitute a security vulnerability? Very few of us have an enterprise in which the Local Administrator account cannot be cracked. This is usually by choice, we choose to give some users Administrator access so that the users can install his/her own software packages. We choose to leave bootable floppy drives in workstations and servers. Either of these choices may result in easy access to the Local Administrator account. Since this is a "shared" password, once the account name and password are obtained for one machine, this information can be used to access all the other NT machines that share the same password for the Local Administrator account. In short, successfully hacking a single machine results in access to multiple machines - and you don't have to be a CISSP to know that this is bad news!

Administrator access + common password = major security hole.

This is the vulnerability: access to one resource allows access to a second resource. Now, how does the access of the first machine lead to access of the second machine? Pay attention here, this material will get your manager's attention in a hurry! If your enterprise is normal and uses a common password for the Local Administrator account then any employee sitting at an NT workstation could own your CEO's access in less than a week. Let's imagine?

A contractor is hired to do some menial programming for your company. The programmer is immediately provided with a freshly-installed NT workstation built to enterprise standards. Using a DOS boot disk and an NTFS tool he copies the local SAM to a floppy disk. Next, using L0phtCrack software he cracks the Local Administrator account on his workstation. Now, posing as "Local Administrator", the contractor can gain access to any workstation because all the workstations have the same Local Administrator password. What is he going to do with this low level of access? Maybe he only wants Administrator access to his own machine so he can install his favorite screensaver. Then again, maybe not.

With Administrator access, he could install a keyboard sniffer on any target workstation and wait for the target to authenticate to the domain. In a short, he will soon know the victim's passwords for NT, Novell, Lotus Notes, mainframes, mail systems, file systems, etc. This points out the importance of securing the NT workstation. The victim of the keyboard sniffing could easily be the CEO of your company! All too often the focus is on the NT servers without sufficient regard to the workstation: securing workstations is as important as securing servers .

All right, so you think the workstation isn't important enough - you only want to worry about your servers? The same vulnerability exists if you have servers with a common Local Administrator password. Suppose your contractor is hired and given access to a single NT server. He may then crack the Local Administrator account on that single server thereby gaining access to all the servers that share the common password. Although your intention may have been for the contractor to have access to the lowly test boxes, he now has access to your production SQL servers. The contractor - who you do not know from Adam - has now bypassed your attempts to restrict his access to a single server!

So, just to recap: in many NT environments, most, if not all, of the NT workstations or servers share a common Local Administrator password. This implementation flaw allows Joe Customer Support to crack the Local Administrator password and use that access to escalate his privileges

all the way up to the CEO. Now, how do we fix it?

## **Solutions to Shared Passwords**

One obvious and straightforward solution would be to eliminate shared passwords altogether; however, in some situations this is not feasible. If management won't let you eliminate the shared passwords, the following mitigation steps will at least let you minimize the scope of your exposure.

### **Limit the Attacker to Machines to Which They Have Physical Access**

To accomplish this, deny the Local Administrator network access to each machine. With this restriction in place, the cracker cannot use one machine to access others over the network. This doesn't prevent an attacker from walking over to the manager's machine and logging in as the Local Administrator, it just forces him to physically access the machine. However, be warned, this mitigation only removes the network access from the attacker, they would still have the capacity for local logins. Essentially it limits only the speed and convenience of compromise.

You will need to point User Manager at each NT machine and remove the user right of "access this computer from the network" from the Administrators Group (this is the Local group) and add the same right to the Domain Administrators group. (You'll have to specifically add the domain group while in the User Rights menu). Be sure the Domain Administrators group is in the Local Administrator's group.

### **Minimize the Time Window for Potential Crackers**

With enough time, even the strongest of passwords can be broken by a brute force attack. Time is indeed of the essence in this regard because the stronger the passwords, the longer it takes to perform a successful brute force attack. Consequently, the stronger the passwords you apply, the less often you will need to change them. You can ensure that the window of opportunity for crackers is minimized by is accomplished by changing the passwords faster than they can be cracked.

While we're busy minimizing the time window, we must also maximize the time it takes for a brute force attack by using "strong" passwords. Passwords should be a least 12 characters in length and include some non-alphanumeric characters. You'll definitely want to develop an

automated mechanism for applying the new password on a scheduled basis. Pointing User Manager at every machine in a domain is just not an acceptable option. If you really must change passwords manually, then consider a commercial option for synchronizing the Local Administrator password across multiple machines such as [User Manager Pro](#). WARNING - this mitigation doesn't rule out the many fine alternatives to CPU cycles - such as cameras, hardware keystroke capture devices, shoulder surfing, etc.

### **Minimize the Scope of Exposure From Any Single Machine**

Even if we cannot eliminate the use of common passwords completely, we can at least use different passwords for different areas of the enterprise. Functional distinctions provide for some obvious logical groupings. For instance servers and workstations should absolutely not share the same Local Administrator password! Consider using a different Local Administrator password for each resource domain, or perhaps for logical geographical divisions such as campuses or buildings. At the very least, make sure your "mahogany row" executive desktops have a different Local Administrator password than the rank and file workstations!

### **Protect the Domain Account Used for Applying New Passwords**

If we are going to have a common Local Administrator password for all NT machines, then the interval between password changes must be shorter than the time required to brute force the password. We must use a Domain account to affect the Local Administrator password change on all the targeted hosts. The danger here is that now this Domain account NT hash will be exposed to any hostile target machine. As a result, it is crucial to protect this Domain account from compromise! For the same reason that we should frequently change the Local Administrator password, we should also change the Domain account we use when applying new Local Administrator passwords.

## **Removing the Shared Local Administrator Password**

Considering the weaknesses and dependencies of the steps outlined above, it is without doubt preferable to eliminate the shared passwords completely.

### **Creating Unique, Unpredictable and Strong Passwords**

First, let's consider what kind of passwords we want to apply in place of the shared password.

The security vulnerability we have been discussing arises from the commonality of a single password; however, the solution is not simply a matter of creating unique passwords, but unique passwords that are also unpredictable. Imagine that every machine in the enterprise had a unique local admin password, but that password was the same as the hostname of the machine! What's the problem with that? Predictability. An attacker must not be able to use the password for machine A to access machine B. Having passwords that are predictable is as troublesome as having passwords that are similar. Clearly, the more unpredictable each password is, the stronger our security posture becomes. Random passwords would be truly unpredictable and therefore are an excellent choice, but we must also consider the strength of a password.

It must be kept in mind that unpredictable doesn't always mean strong. A password such as "37a" might indeed be "random", and thus unpredictable, but is weak and therefore easy to brute force. In order to be truly effective, passwords must combine strength and unpredictability. The idea is to have both 0% predictability and serious strength in order to resist both brute force and logical attacks. Strength is achieved by utilizing a large character set and a sufficiently long password.

## **Recovery of Passwords**

Second, let's consider the administrative issues surrounding the recovery of those passwords. By recovery we mean the ability to determine the Local Administrator password for a particular NT machine in an environment in which the Local Administrator password is not shared. Access to the Local Administrator account on servers or workstations is a requirement for most enterprises. This means that after we apply unpredictable, unique and strong passwords to every NT machine we will need a recovery mechanism.

Why would we need to recover a Local Administrator password? Suppose the "Senior Executive of Irrelevant Paperwork" needs to print that huge Power Point presentation just minutes before the "big" meeting, but her NIC card decides to die at this most inopportune moment. Your support staff can save the day in minutes with the Local Administrator account, or they can lecture her about the value of storing important documentation on the file server instead of her local machine and start one of several time consuming processes to rectify the situation.

It might be a server rather than a workstation - perhaps it is the SQL server in Accounting that requires a new NIC card, or some other high demand machine like the employee internet access proxy server. Whatever the situation, there are inevitably times at which password

recovery will be required.

The recovery process must be simple for reasons of expedience. When support staff need to recover the Local Administrator password for a particular machine, they don't want to be given a paper form requiring multiple managerial signatures! You'll need a central database with careful access controls applied appropriately so that only your support staff can access it.

Alternatively, to avoid the central database you could utilize an algorithmic generation mechanism. This simply means that if you provide the same input to the generation algorithm, it will produce the same output. For instance, if you had a secret knowledge key such as "TrailBlazers" and the hostname of an NT machine, then you could run both character strings through the generation algorithm to create a unique password. Hopefully, you will choose a generation algorithm that provides unique, unpredictable and strong passwords. The uniqueness in this scenario comes from the use of the hostname, which must be unique in any NT domain. The recovery process uses the same technique to generate the password whenever you need it. This solution avoids the storage issues surrounding a central database, but introduces the need to manage the secret knowledge key.

In the event that recoverability is not necessary for your organization, you might consider simply applying random passwords to the Local Administrator accounts wherever possible. A good site for random password generators can be found at [CNET's WinFiles.com](http://www.cnet.com/WinFiles.com).

## Conclusion

A Local Administrator account password shared by many NT machines constitutes a security vulnerability and must be mitigated. If you cannot remove the shared password, then it is vital to minimize security risks by implementing frequent password changes and restricting network access for the Local Administrator account.

If it is possible to operate without a shared Local Administrator password, do so with the following precautions. If support staff does not require access to the Local Administrator account, then consider applying random passwords to this account on each machine. If necessary, make sure that steps are provided to allow for recoverability. I strongly encourage you to create your own solution (see example above) or pursue a commercial package to eliminate this vulnerability. Here's one of many "practical" solutions you can implement yourself:

Divide your enterprise machines into logical groups (Servers, Workstations, Mahogany Row, Sales, Bean Counters, etc..) Use a random password generator to generate passwords for each target machine. Store each hostname/password pair in a central database Secure the database such that support personnel responsible for each logical group can only access the stored passwords for machines in their logical group. Use PERL and the NetAdmin module to script the application of these passwords to each logical group.

Think carefully through the issues of passwords and recovery. Without unpredictable uniqueness you've gained nothing. Without strength you haven't improved your security position. Overlook recoverability and you might be unemployed. Charge blindly ahead without a plan for storage considerations, manual recovery procedures, generation algorithms, and a dissemination process to support staff - and you are definitely asking for a headache.

## References

[1] (The SANS Institute, Windows NT Security Step by Step, version 2.15, 7/30/1999, introduction page 2)

*Daniel Marvin is a senior security consultant for Foghorn Security, a security group based in Portland Oregon.*

### Relevant Links

[Local Account Password Manager](#)

*Foghorn Security*

[Choosing Strong Passwords](#)

*Eric Schultze, SecurityFocus*

[Enforce Strong Passwords in NT 4.0](#)

*Microsoft*

[Privacy Statement](#)

Copyright 2006, SecurityFocus