

Starting from Scratch: Formatting and Reinstalling after a Security Incident

Matthew Tanase 2003-05-07

Missing files, corrupt data, sluggish performance, programs not working - any of these things could indicate a breach in network security. Once the breach has been identified and mitigated, the painful process of rebuilding and recovery begins. There is a point you reach in the recovery process, after you have done a little digging, put a finger on what might have gone wrong, where you come to the proverbial "fork in the road". Every security professional or systems administrator has faced the decision at some point in his or her career: is it better to try to repair the damage, or just reinstall the system and start from scratch?

This IT dilemma will plague us all at some point. In this article, we will examine the process of starting over, and more specifically, reinstalling as the result of a security incident. We will focus on the steps necessary to prevent a repeat intrusion, get your system back online and ensure a rapid response in the future should this happen again. Needless to say, these steps should be planned in advance of any security incident and should be included in the organization's [incident response policy](#).

Why me?

Before we get into the specifics, let's consider how we have reached this unfortunate point. Obviously, there has been a security incident. An intruder has likely breached and manipulated your machine in some manner. So why not fix the problem? Patch the system, clean up the changes and put it back out there. For any particular exploit, even if a well-documented clean-up procedure is in place, it's hard to ensure that modifications outside of the known scope weren't made. Worms, viruses and rootkits can wreak havoc on any system. They often remove crucial files, embed themselves in other parts of the system and sometimes remain silent. And they can be modified to do other nasty things? making the documented clean-up routines, released after a major incident, obsolete.

The reality, as any incident response expert can attest to, is that discovering all of the changes made to a cracked system is extremely difficult. Once inside a system, an attacker can implement several backdoors, modify standard system operations (such as logging) and hide files. Unless there is a file integrity checker such as [Tripwire](#) in place, it's virtually impossible to guarantee a clean system. And there's no worse feeling then spending hours rebuilding a

system, only to have it cracked shortly after putting it back up.

Repairing a compromised system is, without a doubt, one of the most challenging aspects of a security professional's job. While it might seem like the easy way out, wiping a system and installing clean versions of the original software is often the smart choice.

Preparation

Before beginning the rebuilding task, there are a few steps to take that will ease the process. First, consider your response to the cracked system. Obviously, the immediate concern is getting the system back to normal. But in the near future, you might need to investigate the box further, learn how the cracker got in, or perhaps turn the evidence over to law enforcement. With that in mind, consider using a ghosting (disk image copier) or disk duplicating program to dump the contents of the system to another hard drive or storage medium. With the duplicate image set aside, you can immediately get to work on restoring the machine without tainting evidence, and focus on the incident analysis later. A raw system image is a requirement for any type of official incident analysis, so this important step is recommended. If you modify or examine the victimized machine in any way, the data will likely be considered invalid by authorities due to the numerous aspects of the system which can be compromised. Much like a physical crime scene, this digital evidence needs to be documented, preserved and protected from contamination. Disk imaging software, such as Ghost, provide incident handlers and forensics experts with the clean slate they need to begin an investigation. (For more information on dealing with law enforcement agencies in forensic investigations, please see the SecurityFocus article [Incident Management with Law Enforcement](#) by Ron Mendel.)

Next, you need to audit the system. Take note of the servers and services running, important configuration files, patches, the third party applications in place, the users, the directory structures, and so on. Additionally, consider saving especially critical files, but be warned that they could have been manipulated or corrupted at some point. Obviously, you'll want to avoid capturing the malicious changes to the system, but you should be able to cull the basics with a general review.

Lastly, have all of the original installation disks, registration codes and support numbers at hand. It's best to have these in place before the process begins, so you aren't frantically digging around for a disk or number in the middle of the setup. In addition to this software

repository, keep a journal of each step you take. This record will help track the rebuilding process. Additionally, it might prove to be a handy reference should you need to rebuild the system in the future.

Formatting the Drive

The big step in rebuilding a system, the point of no return, is wiping or formatting the system drives. This will destroy all of the data on the disk and make it possible to reinstall clean, system software.

You might wonder if it's possible to repair or upgrade a system, a choice available for many operating systems. If you repair a system, a process which normally requires an emergency or repair disk, the OS cleanses itself by replacing or reinstalling critical system files or missing applications. The problem with this lies in the fact that while the repair option might catch some modified or missing files, it likely will not recognize what was added to the system. Therefore any backdoors, extra applications or otherwise malicious code will remain in place, undetected. So a complete reinstall including a disk format is the safer choice when dealing with a compromised machine.

Formatting the drive is, today, a relatively simple process. Most modern operating systems simply require you to insert the installation boot disk. Shortly into the process, you are presented with a list of drives and installed OS's. You'll likely want to select all of the drives for formatting. The partitioning (disk spacing) can be handled automatically, but if you have specific requirements, mimic the previous configuration. In the past, formatting a drive was somewhat more tedious and a mysterious process left up to the user. It required a bootable system disk and a program such as 'fdisk'. If you must use this method, boot from the necessary disk and use the utility to wipe the drive clean.

Another option, if you want more control or assistance with this process, is a third party utility such as 'Partition Magic'. Such software makes it easy to resize existing partitions and format drives in a number of different formats. Consider similar utilities if you encounter problems with the OS formatting process described above.

Rebuilding the Systems

With an empty system in front of you, the next step is to reinstall the OS software. This

straightforward process will vary depending on your software. Follow the installation guidelines provided to build the bare-bones system. After the OS, move onto installing specific applications, such as servers, utilities and other programs you require. Again, the process is different for each application, but there shouldn't be any unexpected challenges. If possible, and ONLY if you know they were untouched, reinstall the critical configuration and system files copied from the compromised machine. Or, at the very least, review them while configuring the current setup.

By this point, you should have a decent replication of the original system, but keep in mind - it is still offline. Before reconnecting to the network, security needs to be tightened, or we will end up back where we started. Begin by removing any unnecessary open ports or network services. Use a portscanner such as [Nmap](#) to determine what servers are listening. Turn off everything but the absolute essentials. Next, review the running applications. Again, if something seems unnecessary, remove it. We want this system to be spartan in terms of processes - each one is a potential vulnerability. Bring the OS and application level patches up to date. These patches, often security related, are available from the vendor sites. It's a good idea to group the patches onto a disk before beginning the rebuild. Therefore, you won't have to put the system online while it's still insecure. Additionally, take note of every patch applied, for future reference.

A vulnerability scanner, such as [Nessus](#), is a good utility to employ during this process. These tools check the system against a known database of vulnerabilities and generate a report of potential threats. Make sure all aspects of the report are addressed before bringing the system up.

Lastly, consider installing an integrity checker (such as Tripwire), which can help in both the short and long terms. Immediately, you'll be concerned with a repeat incident. If you missed the original vulnerability in the rebuilding process and the system is compromised again, an integrity checker will alert you of changes. Long term, the benefits are similar. If the machine is hit again, a quick list of changes will be available. An intrusion detection system, such as [Snort](#), can also help you monitor the network for the attacker's return. Monitoring is a crucial component once a system is back online and both of these utilities can help immensely.

An important point, which deserves repeating, is that the system should, if possible, remain unconnected to any network during the OS reinstallation and patching process. This means that you need to compile all of the necessary software: the OS, specific application and patches, before hand. Rebuilding a machine without network connectivity can be done on some

operating systems, but is somewhat difficult on others. If circumstances demand network connectivity, proceed with caution. Make sure ALL listening services are shutdown prior to connection. Additionally, the machine should be placed behind a firewall which blocks inbound traffic requests. Lastly, it should be the only machine on the particular network segment, to prevent an internal virus or worm from reaching the machine. An unpatched machine is extremely vulnerable to multiple threats, so make sure the proper defense techniques are in place before putting the machine on a network for updates.

Going Back Online

Before bringing the system up, you need to create a system backup. Since you just rebuilt the machine from scratch, it's fair to say that a backup was not in place prior to the compromise. Backups are a fundamental aspect of system administration and security. At some point, they will be needed. In addition to the full backup, you need to create a regular schedule for incremental backups. This will help ensure that frequently modified files are saved to a secure medium.

Finally, we can bring the system back online. The fresh build, newly applied patches and security review should prevent an attacker from returning. If the machine is compromised again, it's safe to say you missed the original vulnerability or have fallen prey to an insider attack.

For a while, monitor the system with increased frequency. Review logs, security mechanisms such as filecheckers and intrusion detection systems, and general system activity on a regular and frequent basis. You need to ensure that the machine is no longer vulnerable. Unfortunately, this is invariably a wait-and-see process.

Conclusion

Rebuilding a system is never a pleasant task. It is, however, often the proper choice, when dealing with compromised machines. Sometimes, it's the fastest route to restoring the status quo. The process demonstrates how important regular backups and strict security procedures are for networks. When you do need to start over, the basic steps outlined in this article can ensure a rapid return to action and prevention of further incidents.

[Matt Tanase](#) is President of [Qaddisin](#). He and his company provide nationwide security consulting services.

Additionally, he maintains *The [Security Blog](#) and the [Wifi Security Project](#), Web logs dedicated to network security.*

Relevant Links

[How to Design a Useful Incident Response Policy](#)

Timothy E. Wright, SecurityFocus

[Detecting and Removing Malicious Code](#)

Matt Tanase, SecurityFocus

[Introduction to Autorooters: Crackers Working Smarter, not Harder](#)

Matt Tanase, SecurityFocus

[Nmap](#)

[Nessus](#)

[Tripwire](#)

[Snort](#)

[Privacy Statement](#)

Copyright 2006, SecurityFocus