Final Project Part 3

Stewart Johnston
CIS 127 – Intro to Information Security
NCMC
`johnstons1@student.ncmich.edu`

System hardening and auditing

Microsoft provides a handful of mechanisms to address auditing and system hardening needs in Windows in their own particular philosophy. Some of these are of their own doing, and some are inherited from industry standards set decades ago. Among these are:

1. Access controls and permissions for their filesystem. Sequestering users and groups and controlling who can access files can help keep both accidental and intentional unauthorized access from occurring.
2. Encryption in both a granular sense and for full-disk security. This helps to keep sensitive data safe when it is at rest. That is to say, when not in use. I know from first hand experience how trivial it is to break permissions if I can physically access a machine, since I've had to do so to repair machines when the user forgets their credentials. Sensitive information should only be stored locally on machines if that information is encrypted at rest.
3. regedit – The registry editing tools allow for administrators to audit the registry themselves and fix issues they encounter.
4. gpedit – The group policy editing tools help to make managing large numbers of computers comparatively easy. This also allows administrators to set logging and auditing policies on an entire domain at once.
5. Event viewer – What it says on the tin: allows for detailed auditing of the binary log data reported from multiple sources within the system. Once auditing policy is set, this is where logon events can be checked.
6. Others outside the scope of this document which help to flesh out the toolset.

Antivirus updates and scans

AVG is a trusted antivirus company which provides several conveniences for administrators. Among those is a management console, a tool which requires that machines be on the same domain as the AVG administration server. This allows, among other things:

- On-demand and scheduled scans of arbitrarily large groups of machines.
- A centralized destination for warnings about threats found during scans, with the option to respond as appropriate.
- A centralized update and upgrade distribution cycle. Instead of every machine reaching out to AVG for updates or upgrades, the administration server does this as needed, and machines on the domain can be served virus definition updates and software upgrades remotely.

These things can be done manually on each machine, and for the sake of argument I'll explain how, but it is tedious. To check for updates in AVG, one would open the AVG application, locate the help/about section, and check for virus definition or software updates. It will not automatically upgrade the software without asking, because this often requires a computer restart.

Scans can be initiated manually by pressing the big green Scan button, or they can be scheduled. This can be done in the GUI using Scan Options -> Manage Scheduled Scans. They can be set for a regular basis and enabled.

Logs, as far as I can tell, are not exported to be viewed by Windows Event Viewer unless through the use of AVG CloudCare. Otherwise, their logs are locally stored in several possible locations. It is not uncommon (though mildly annoying) for antivirus vendors to require the use of their own tools to view logs.

Logging and auditing logon events

gpedit, mentioned earlier, comes into play here. This makes it possible to log on both the domain controller and the workstation both successful and failed logon events. In gpedit, the details can be found in Computer Configuration -> Windows Settings -> Security Settings -> Local Policies -> Audit Policy.

There are two kinds: "Account Logon" events and "Logon" events. Microsoft's naming scheme is unfortunate, and it's easy to confuse the two. There isn't a significant reason not to use both. In short, the difference is:

Account Logon Events	Logon Events
Logons for which a machine is the authority. DCs are the authority for domain accounts.	Logon/off events on any given machine. Logged locally to that machine.

How these are viewed, ultimately, is in the event viewer. I have a tendency to type faster than I think on occasion, so it is not unusual for me to have two failed logons before a proper success. Using both Account Logon and Logon events, a success will likely show as two successes, which is fine. These events can be found under Windows Logs -> Security.

To filter for only this kind of event, we can handle it in a couple of ways. Using “Filter Current Log” or “Create Custom View” on the right side, we can filter by event ID number code using the list of event codes that Microsoft publishes. To filter by kind, Logon, rather than the granular numbered ranges, we need to choose Event Source -> “Microsoft Windows security auditing”, and *then* choose Task Category -> Logon, or any other task categories desired.

Potentially problematic events

I tried this with two machines. On each were different issues. The first is a workstation, where I noticed that the allowed log file size was quite small. This is a problem because within 30 minutes, enough noise was generated in the log files that it pushed other, useful events out of the log. In this case, Windows was logging noisy machines on the network which were using multicast DNS every second in an attempt to find other machines to network with. This was just a handful of misconfigured machines doing something mostly harmless, but their sheer volume of chatter pushed out the useful logs I rather wanted to keep.

The other was significantly more worrying. I had a cloud-hosted server with Windows Server running for several days, which I had forgotten to shut down. This was hosted through Amazon Web Services, and for servers of that kind, they recycle IP addresses not in use, so what I was seeing was not a targeted attack, but likely the result of port sniffing. Several hundred, if not several thousand, failed logon attempts were logged, sometimes multiple times a second. The originating IPs were from a huge range, and each IP would wait a handful of seconds before retrying. This was likely a botnet which happened to spot my server online and started hammering away at it. The default login of Administrator was the target, including other guesses like “KATIE”, “NEWUSER”, “TEACHER”, etc. Thankfully, Amazon’s Administrator passwords on configuration are both quite random and strong.

What to do about these events

The first case is fairly straightforward. Increasing the size allowed for logging will help prevent spammed errors from disrupting the ability to audit. The other solutions include reconfiguring the chatty machines. They were Macintoshes using the Bonjour service, shouting for friends, which is useful in small domains with no DNS server, but not so useful in large domains with a properly configured DNS. That’s a prime example of well-meaning default configurations getting in the way.

Another example yet is to use regedit, mentioned earlier, to add an exception for failures of that kind from those specific IPs. Fixing the misconfigured machines is the saner solution in the long run, but almost certainly takes more time than just ignoring those logs or increasing the log size until that can be achieved.

For the server… that's tricky. Because it was a botnet distributing its attempts over many IPs and many different user names, that's difficult to guard against without also making it difficult for myself to log in. The only obvious solutions are also tricky to execute correctly. Configuring the server and firewall rules to only allow in a handful of static IPs, such as the static IP that Fullsoft pays for with business class internet, prevents botnets like this from outside the network from succeeding. It is wishful thinking that such software as Fail2Ban would be useful here; since the IPs were so distributed, the effect would be limited. Disabling the Admin builtin account would be useful, until such a time as a custom admin account is forgotten or accidentally locked. The easiest solution which comes to mind immediately is to set up either private key/passwordless login (fairly trivial for Linux boxes with SSH), or some variation on using cryptographic keys resulting in very long, borderline impossible to guess passwords, and changing those on a regular basis. Besides automated solutions preventing or avoiding this, configuring the server to email me when several logon failures are noticed in a short period would not be unwise.

Hardening and auditing and towards Fullsoft's goals

Mentioned in previous reports in the fallout of Fullsoft suffering from malware induced data theft, account and access controls are literally the least we could do in pursuit of hardening our machines. A principle of least-privilege applied to user accounts, and logging events generated is a step up from that. These and a number of other hardening techniques are relatively trivial to implement, without sacrificing much in the way of convenience, and providing the IT team more to work with.

We will almost certainly be the target of attacks in the future, whether these are random from passerby sniffing for vulnerabilities, or targeted. Logging and auditing processes will allow us to understand when and how this occurs, and thereby also the frequency.

Some hardening operations require support from the DevOps team, in particular how proprietary source code is handled. If we use a distributed version control system such as git, then such techniques as code signing wouldn't be unwarranted. Forbidding the git server from serving clone requests from IPs outside our network may be wise. If we don't already, acquiring business internet service for telecommuting workers wouldn't be unwise, so as to ensure static IP addresses for them as well.

Worrying trends and how they affect Fullsoft

From both a social engineering and a digital security standpoint, social media is quite often a dangerous thing. Oversharing information on social media can disclose secrets, sensitive details about operations, and details about oneself that are used for security purposes. Security questions, for instance, should not be used with answer that can be found on social media. Two suggestions come to mind:

1. Don't post this information, or don't use security questions which are based on this information. Use custom security questions, rather than your mother's maiden name.
2. *LIE!* Nobody but you should know the answer. Telling the truth makes it easier to remember, sure, but there's no restriction on you lying about your first pet or model of car. As long as you can remember it, that's what matters.

How this independent research helps Fullsoft

First thing that comes to mind in how my projects demonstrated here can help Fullsoft is in demonstrating how easy it is to configure for logging, once one knows how. Many if not most or all of the hardening techniques discussed here can be distributed from the group policy as set by the domain controller. Techniques like full-disk encryption will require more labor by the IT team, but are also not out of reach or impractical. The suggestions explored in this document can and should all be applied in short order.