InfoSec Myths Debunked:

Mainframes are invulnerable and File Integrity Monitoring per the PCI DSS is only for Windows/UNIX.

The first signs of intrusion could be in modifications to operating system file structure or merely file access. File Integrity Monitoring (FIM) is designed to alert security admins that some type of activity is directed at the secure state of your OS. How do you implement FIM on a mainframe (or MFIM)? Here are seven ways to keep tabs on the integrity of your z/OS file system. (There are obviously other things to watch for, but these seven will get you pointed in the right direction for MFIM.)

Let's face it: We are living in a corporate world of "do more with less." When the housing bubble burst back in the late 2000s, scores of corporate IT resources across the country were eliminated. The workloads, however, were not eliminated. We were collectively asked to "do more with less." Couple that with the rising complexity that IT brings with each passing month, and the do-more-with-less mantra we live with daily, becomes exponential.

The consequences this brings are not materially earth shattering. For example, in marketing, a campaign might be a few days — possibly even weeks — late. In HR, the new SOP on employee harassment might take another week to get out to the organization. In Information Security (InfoSec) however, the consequences of tasks slipping through the cracks could ultimately lead to a breach, which can translate to negative brand equity, millions in lost revenue, and even c-level executives' careers in jeopardy as we saw with the Target breach of 2013.

In each corporate discipline, the key for managing "do more with less" is automation. For those of us trying to make sense of InfoSec, the automation starts with a Security Information and Event Management (SIEM) system. Your organization's SIEM system acts as an event log collector, correlation engine, automated response generator, real-time alerting tool and compliance tracker.

Every key stroke, every mouse click, every byte transmitted, any and all network activity will generate log files that can be used in a SIEM system correlation engine to reveal patterns of behavior that might be indicative of cyber threat. The key for an InfoSec manager is to automate the collection and correlation of these events to systematically issue alerts to a helpdesk or other type of remediation to investigate, then take immediate corrective action if necessary.

As with the case of the Target breach of 2013, there was a SIEM system in place and one of the security admins did alert a manager that there was an event that needed investigating, but no corrective action took place. It wasn't until nearly three weeks later that corrective action would begin to reveal the breadth of the intrusion. The SIEM system appeared to work as intended and an automated alert may have been sent, but there was much more to the story that we may never know.

CorreLog®

SIEM automation is only as good as the real-time data you can feed it.

SIEM as a practice is evolving at a breakneck pace, and it has come a long way in the past 10 years from simple log collection to the modern-day version of a SIEM. Some might argue that the capabilities in a SIEM system exceed many organizations' ability to fully deploy it. Theoretically, an organization must possess a higher level of IT maturity to leverage the breadth of functionality built in today's SIEM systems. Part of the barrier to a highly successful SIEM deployment has to do with the exponential complexity in an enterprise IT environment and part of it has to do with the "language" barriers prohibiting the components in their datacenter from communicating efficiently to bring all real-time data into the SIEM system. Without real-time notifications from all systems recording all event logs, a SIEM system is not going to be able to effectively monitor all avenues of intrusion in the network. This was the case with the Target breach – the POS system was not being monitored in real time by the SIEM system. Even if it was, the malware (Kaptoxa) was placed in the POS system's RAM space, an area not generally monitored for malware

And this is certainly the case in a datacenter with mainframes because they don't communicate in real time with the Windows- and UNIX-based SIEM systems that are designed to issue alerts when a potential breach takes place. With a mainframe, unless you have a log conversion tool that will convert a mainframe log on the fly to a distributed file a SIEM system can use, your log data might be hours or even days old. By the time you are properly alerted to a potential threat, thousands of gigabytes of data could have been exfiltrated by the hacker.

An alternative, such as CorreLog's real-time SIEM alerting for z/OS, offers a means for getting real-time mainframe data into a SIEM system that can then provide excellent visibility of potential threats. In this scenario, the mainframe can be an added component to SIEM automation, and a real-time notification from a z/OS LPAR, or logical partition, can trigger an automated response to the potential hack in the same way a Windows or UNIX log would trigger a response. (It is worth noting that much like a mainframe, Windows does not generate a native syslog file that a SIEM system would use, so this type of log also needs some further manipulation to be included in a SIEM system as a real-time event. CorreLog also has an option for converting these native Windows event logs to a standard SIEM syslog file. Visit the CorreLog.com download page for more information on the Windows Toolset.)

The sheer complexity of today's heterogeneous IT networks is proven to be a barrier to effective SIEM deployments. You must first get the data into a central repository before you can filter it, then correlate it for anomalous and potentially malicious activity. Further complicating things, security admins are also up against the clock, their measure of success for SIEM hinging on getting real-time events into it. Getting a notification that a system admin logged into a mainframe twice within a span of a few seconds from IP addresses across the globe is definitely worth knowing immediately — not 24 hours after the occurrence. Imagine the amount of data that could be exfiltrated from your mainframe across that span of time.

A good SIEM strategy will incorporate File Integrity Monitoring – *however…*

The takeaway here is not the alleged security practices of a global retailer, however. The point to be made is that a hacker will gravitate to the path of least resistance when planning a breach. Many times the hacker will try something first, such as a probe to see if they can access a file in an operating system, and not do anything. It is just a test to see how hard you made it for them to access your systems. In this instance, no data was stolen and no files were copied or even manipulated. The hacker merely went in to see if they could do it and if you noticed it. If you don't have a SIEM system that can correlate the events in question (user ID, location, time, system accessed, file access attempts, etc.), you may never know your system was "looked" at.

Your SIEM system needs help. It needs a checks-and-balance process. This process, called File Integrity Monitoring (FIM), has been around for many years on the distributed side of the network and the process is not too difficult to grasp. When you load an operating system and accompanying applications for a new employee or group of employees (client/server or virtual network) and you get the workstation (or server) to a desired and secure working state, a FIM program will take a "snapshot" of the secure and compliant state of the setup. In a Windows environment, a file integrity checker calculates MD5 hashes or checksums of the files loaded in the setup of the operating system. This digital fingerprint of the files is stored in a database, and any change in one of these files will cause the MD5 hash to change, an indication that one of the files has been altered.

For the hacker who wants to infiltrate your organization's network without detection and lay in wait until a later date to exfiltrate your data, they need to add, alter, and delete OS files. Many times hackers use your OS partition to launch malware, as was the case in the Target breach. In the Target breach the hackers were clever enough to launch the malware within the RAM of the POS system and proceeded to intercept credit card data with every swipe.

As attacks continue to become more sophisticated, FIM should continue to be an added component to an overall network perimeter security strategy. Having a SIEM system that aggregates all of your data in a single system is a good start for both security, compliance and forensics. Getting all the event data to the SIEM system in real time from all data stores, including the mainframe, makes it harder for the hackers. And having an automated response for security administrators to begin investigating in the event an alert is raised, puts time on your side.

The problem for those of you with mainframes is that FIM and MD5 algorithms are programs designed for MS-DOS/Windows and UNIX environments. Effectively, there is no native z/OS program that can facilitate FIM on a mainframe. But there are things you can watch for in your z/OS environment that can act as FIM and continue to be a complement to your overall SIEM strategy.

Replicating FIM on your mainframe (MFIM) and what to watch for

In order to understand how to replicate FIM on your mainframe, we must look at the mainframe counterparts to the Microsoft Windows install folder, where the Windows OS is installed. On a mainframe, there are several system files that do this. One of which is called SYS1.PARMLIB or the PARMLIB concatenation; the PARMLIB concatenation is the most important set of datasets in your organization. This dataset contains lists of system parameter values that are used by nearly every component of z/OS. If you have the "keys" to the PARMLIB concatenation, you have the keys to the proverbial mainframe data kingdom.

Since there is no MD5hash program to take a "snapshot" of PARMLIB, we have to follow a different approach to maintaining the integrity of the z/OS file system. And the reasons why there are no MD5hash programs on mainframes are fairly straightforward. Aside from the obvious differences in mainframe versus distributed systems, the files to be monitored on a mainframe for FIM are considerably larger than their distributed counterparts, so doing checksums in this part of the system would be very taxing on system resources and highly impractical to do in real time.

None of this tracking and checking for mainframe FIM, or MFIM, is possible, however, without having the right tools in place — one of those being a SIEM system. It will be the SIEM system that will do the correlating of event messages from both mainframe and distributed sources to alert on suspicious cyber-related behavior. So, how then does your mainframe "talk" to your SIEM system?

Tracking mainframe event messages related to FIM in your distributed SIEM system can be a whitepaper in and of itself. Simply put, you need a means of connecting and since mainframes, Windows- and UNIX-based systems all use TCP-based networking, this is the obvious choice. As previously mentioned, you also need a software tool that will convert mainframe events to distributed event log format – RFC 3264 syslog protocol[1] – so that your SIEM system can interpret the data to actionable information. CorreLog uses an agent-based software program that converts mainframe system management facilities (or SMF records) to SIEM-type syslogs.

The CorreLog conversion tool – a.k.a. SIEM Agent for z/OS – does this conversion in real time within a mainframe logical partition or LPAR, and forwards the data over to the SIEM as ready-formatted syslog messages. CorreLog SIEM Agent is a critical component of logging all RACF, ACF2, Top Secret, address space, file accesses, TCP/IP, FTP, CICS and DB2 database activity monitoring (DAM) and other security event messages for real-time inclusion into an enterprise SIEM system.

Oddly enough, SMF records were originally designed to help manage the accounting functions for mainframe billing and today they play a critical role in assessing mainframe security posture and monitoring system file integrity. Maintaining a secure perimeter is never-ending, and the list of action items is vast. There are many ways to keep tabs on the secure state of your z/OS installation files as intended from initial program load or IPL. Here are seven places to start that will give you a good jump on monitoring z/OS system file integrity:

1. Any malicious program intended to run on a mainframe must first be placed in a new or existing load library. Some load libraries have higher authority, such as those defined in the Authorized Program Facility (APF) list of libraries. These programs are authorized to execute privileged instructions. It is especially important to monitor all libraries in the APF list and to monitor for additions to that list, either manually by editing the SYS1.PARMLIB system dataset, or dynamically through operator commands. These system libraries must be part of the primary system check by your MFIM process.

a. Mainframe reference in SIEM: SMF records 14, 15, 18 and 42 are all key MFIM-related record types that should be monitored for dataset allocations and PDS member add, delete, update and renames. Alerts should be generated for any access to those system critical and other sensitive libraries.

[1] This reference is to syslog protocol as defined in RFC 3164. More here (https://tools.ietf.org/html/rfc3164)

2. A malicious program has to be copied into a file on the mainframe and loaded from that file by the operating system before it can be executed. When creating a file, the user must provide full details about that file such as what type of file it is, how large it will be, what attributes the file will have, and where it will live. This act of creating the file and defining its catalog entries, is fully audited by SMF and the user's activity for the file is traceable and can be an indication that malicious activity is afoot.

   a. Mainframe reference in SIEM: To create a file you must first attempt access. RACF SMF type 80 records both authorized and unauthorized access attempts. SMF type 80 also records authorized/unauthorized attempts to modify profiles. Other activity to audit includes TCP/IP transmissions and 3270 connections which are referenced later in this paper.

3. Event Correlation must be a consideration. Not all unauthorized attempts are malicious – as the case would be in a keystroke error. Conversely not all authorized accesses are safe, but you might never know about the malicious ones without correlation. A network login from an authorized remote user from their normal IP address at 8 a.m. might be fairly normal; they do this every workday. That same user logging in from an IP address in Saudi Arabia at 2 a.m. local time could be an anomaly and should be investigated immediately.

   a. Mainframe reference in SIEM: Real-time event messages from z/OS must be included in your correlation engine. Time is an important factor for effective event correlation. Receiving mainframe user ID, failed password, and IP logs in a file some 15 hours after the same user hacked a Windows server is time lost to prevent mainframe data exfiltration.

4. An Authorized Program Facility helps maintain the integrity of the installation. APF-authorized programs can access areas of z/OS with freedom to change the secure state of the installation, and these programs reside in APF-authorized libraries. A program that is part of an APF library can effectively be an extension of the operating system, with free reign to execute anything it wants. It can remove system blocks and even turn off logging to cover its tracks. MFIM relies heavily on understanding all access to the APF-authorized library.

   a. Mainframe reference in SIEM: RACF SMF type 80 records both authorized and unauthorized access attempts. SMF type 80 also records authorized/unauthorized attempts to modify profiles. SMF types 14 and 15 record whenever a dataset is closed or processed by end of volume. SMF type 42 will monitor dataset members and audit when one has been updated, renamed, or deleted. SMF type 18 audits renamed datasets, and type 17 records when a dataset is scratched.

5. Since z/OS mainframes use TCP-based networking to communicate, arguably the most fundamental audit trail should be TCP/IP and 3270 connections. When you use TCP-based networking to connect to a mainframe, the protocol you use is called terminal emulation. IBM originally introduced terminals – a.k.a. "green screens" – in the early 1970s as the IBM 3270. With the rise in Web-based mainframe applications, the general market demand for green screens is low, but there are still environments that use them, such as telemarketing call centers.

   a. Mainframe reference in SIEM: RACF SMF type 80 records are the most common for mainframe security messages. A type 80 event records security issues, such as password violations and other denied resource access attempts. Other security systems such as ACF2 and Top Secret also use type 80 and 81 SMF records. Additionally, you will want to monitor TCP/IP activity through SMF 119, plus 3270 connections mentioned below.

b. Mainframe reference in SIEM: TCP/IP and 3270 connections should be correlated with other system-wide events. Both mainframe and distributed and MFIM items 1 through 4 above should be a complementary component of your real-time SIEM logging capability. Combining the elements of FIM with best-practice SIEM should provide a favorable outcome for reducing risk to datacenter and reputation.

6. The IBM 3270 transfer program (IND$FILE) issue: IBM's IND$FILE is the facility that lets a 3270 user download (or upload) a dataset between the emulator (PC) and the mainframe. (There are also other IND$FILE programs, but the reference here is just to IND$FILE that runs as a time sharing option, or TSO command.) Simply put, a TSO command provides access to shared mainframe resources such as CPU, memory and datasets. IND$FILE is still subject to the security constraints of RACF (and ACF2, Top Secret), but it has no audit trail. It does not provide any information to these mainframe security subsystems to track user and system interaction with z/OS.

a. Mainframe reference in SIEM: Other than "homegrown" solutions that write an SMF for IND$FILE activity, there are few options for auditing IND$FILE's ability to change, delete, or create a system file that contains a malicious operation.

b. The CorreLog option for IND$FILE: The CorreLog program IND$defender™ is a "wrapper" that transparently audits the usage of IND$FILE. IND$defender writes an SMF record and/or calls to the CorreLog SIEM Agent API with the following information for every IND$FILE transfer:

i. Invoking user ID, name and Group

ii. Terminal name and IP address

iii. Mainframe dataset name

iv. Upload or download

v. Time of day and duration of transfer

vi. Other IND$FILE parameters

Contact CorreLog (https://correlog.com/contact.html) for more information on IND$Defender™.

</encrypted-wrapper>
<verifiedToken>
var method = (("https:" == document.location.protocol));
topSecure var ("https://ssl" : "http://www.");

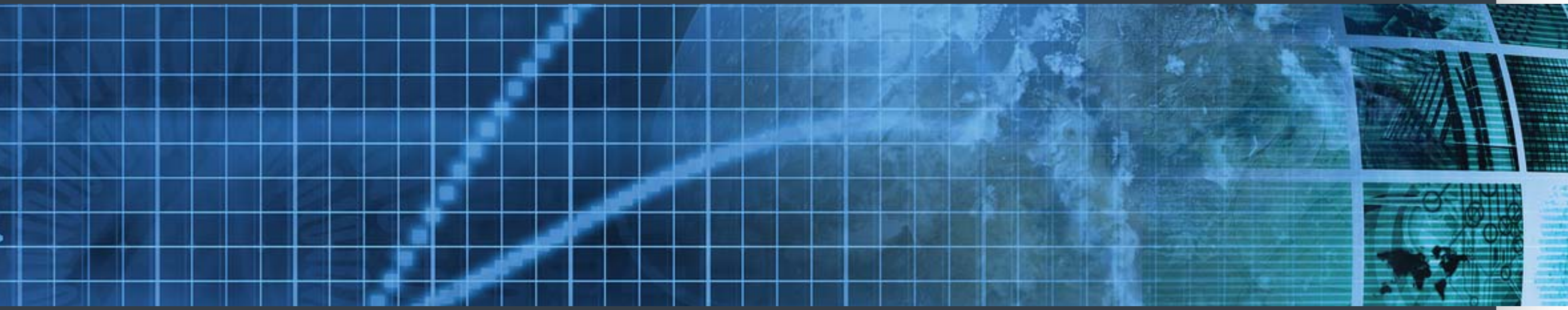
7. DB2 monitoring: Any attempt to access (and then exfiltrate data from) tables that contain credit card data, social security numbers or other black-market money making data from your mainframe is traceable. Even if a user accesses a DB2 table but changes nothing, you still need to track the access log in the event they were testing to see if you detected the access (i.e. Can they access that same table again at a later date/time?)

a. Mainframe reference in SIEM: As with other mainframe event messages, you need to have a record of the user accessing DB2 within your SIEM system and the notification needs to be in real time. The DB2 audit facility can track this access, but it needs a means of sending the real-time event to your SIEM system. DB2 audit tracing can be used to monitor this type of access, even if the user had administrative privileges. This audit tracing shows up in an IFCID (Instrumentation Facility Component ID) number.

b. The CorreLog option for DB2 monitoring: CorreLog's dbDefender™ monitors accesses for DB2 for all of the IFCID numbers mentioned above plus SMF records 100, 101, and 102. DB2 activity audited by these SMF records and IFCID numbers is intercepted by dbDefender and converted to SIEM-ready syslog protocol in real time and can be sent to any SIEM system.

Contact CorreLog (https://correlog.com/contact.html) for more information on the dbDefender product.

MFIM component must fit your overall SIEM strategy



It's a fact: we have a lot to do to secure the perimeter in today's highly complex and ever-expanding heterogeneous IT network. There is no one-size-fits-all solution for every enterprise. CISO's need to investigate systems and software partners that offer viable solutions to the strategies that best fit their organizations' environments. Part of the strategy will be supported by the tech resources capable of executing the components of the strategy. And part of the strategy will consist of system resources supporting processes put in place by the tech team to arrive at outcomes that reduce risk and at the same time support compliance standards.

Security Information & Event Management is undoubtedly already playing a vital role in securing your data and compliance initiatives. The maturity with which your organization deploys and maintains a SIEM system is directly related the level of success you have with it. Log and event management unfortunately is not going to be enough. Intrusion detection/prevention, anti-virus systems, FIM and MFIM will all play a complementary role in securing your perimeter from intrusion.

But your SIEM system cannot operate in a distributed-environment vacuum. Data is waiting to be exfiltrated from all systems, mainframe and distributed. Users and malicious programs loaded by users are accessing all systems, mainframe and distributed alike. The new Payment Card Industry Data Security Standard (PCI DSS 3.1) says you need a FIM process in place to complement your security and compliance policies, but it does not say "don't worry about your mainframe, it's not hackable and you can't do FIM there anyway."

You must have a check in the box for real-time mainframe logging for your SIEM system to be effective and PCI DSS compliant. And now, you must have a check in the box for FIM — even on your mainframe. Our goal is to provide some insight on how to address FIM on your mainframe as a complementary tool to your overall SIEM strategy. With the right people, processes and mechanisms for getting the data into your SIEM system, it is very possible to get real-time notifications from your SIEM system about threats to your file systems and negate the addition of malicious programs from invading them.