WHITE PAPER

FIM Isn't Just for Files Anymore

Comprehensive Integrity Management for Traditional IT, Cloud and DevSecOps Environments

FOUNDATIONAL CONTROLS FOR
SECURITY, COMPLIANCE & IT OPERATIONS

Integrity is Foundational

The CIA triad (confidentiality, integrity and availability) has been a fixture in information security for many years. It's widely accepted that protecting these three aspects of data and service are core best practices. Integrity, however, is often relegated to the realm of encryption, while confidentiality and availability get more attention. It's a mistake to overlook integrity, however. Broadening the scope of integrity beyond data and focusing on it as a leading principle for risk management can lead to extensive benefits. Not only is integrity foundational—Integrity Management just might be the way to make information security successful.

Every Incident Starts with a Change

Accepting this simple fact can dramatically change your perspective on preventive and detective controls in your environment. You may find some value in shiny, new technologies like machine learning, artificial intelligence or even active threat hunting. But a close examination of how incidents occur creates exponentially more value in the ability to detect and prevent changes, and to do so effectively across your entire environment. While your impression of file integrity monitoring may date back to running open-source Tripwire on a Unix server, the technology has changed and the value has dramatically improved. File Integrity Monitoring and change detection are inextricably linked, and detecting change is at the core of FIM.

Making the Shift: FIM to Integrity Management

File Integrity Monitoring may describe a very specific set of capabilities, often associated with meeting compliance requirements, but it's also become shorthand for a broader application of integrity; FIM isn't just for files anymore. Shifting language can be difficult, but it's more appropriate to talk about Integrity Management in regards to today's technology landscape. Integrity Management provides an umbrella approach to managing risk in an environment, and it can be used alongside compliance and security standards. There are four basic steps to ensure integrity.

1. Start with a Secure Deployment

The first place to apply the principles of integrity management is at deployment. Every organization should work to ensure they're deploying systems that meet risk acceptance criteria. That means you have to establish those criteria and be able to measure them for servers, images, containers and any other system that gets deployed, whether on-premise, virtual or in the cloud. Ask yourself which systems in your organization don't get this treatment.

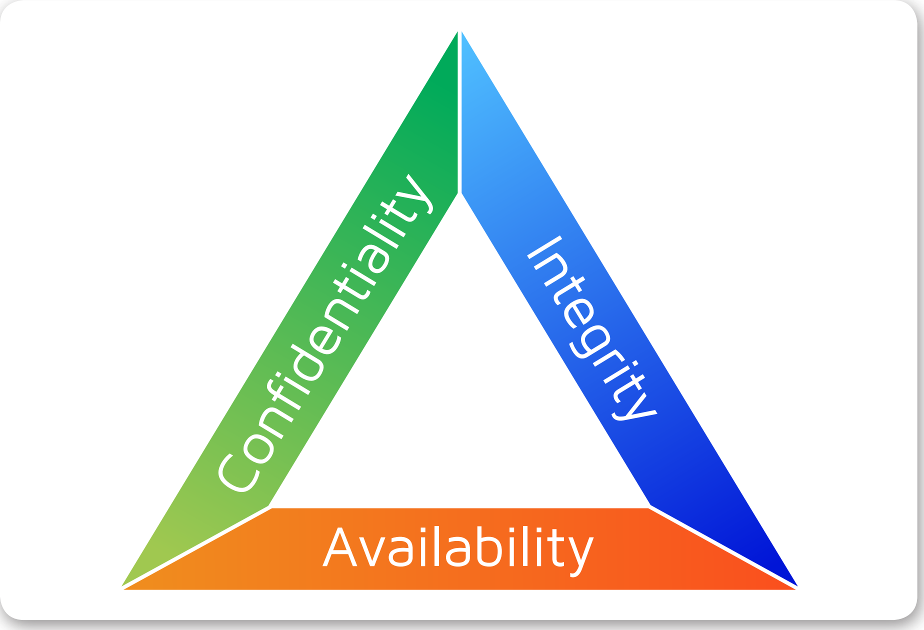

Fig. 1 The "CIA Triangle"

2. Baseline every system that's deployed

The time to establish a baseline for a system is when it's first deployed. That baseline is crucial for being able to identify changes and determine how they might affect the risk posture of that system. The baseline should be closely correlated with the standards for secure deployment of that type of system.

3. Monitor systems for change

Detecting change is at the heart of Integrity Management. Once you've deployed and baselined secure systems, you must be able to detect changes that compromise the integrity of that system. This process requires a close connection between change detection, baselines and the change process for the organization.

4. Investigate and remediate changes

Not every change requires action. Implementing a reconciliation process to separate the wheat from the chaff is crucial. Changes that are business as usual and associated with change orders or planned updates don't require response. Changes that can't be reconciled or changes that impact risk must be investigated and remediated. In order to do so, you must have sufficient detail about the changes to make decisions.

It may be implicit in the discussion so far, but the shift from File Integrity Monitoring to Integrity Management acknowledges that the technical capabilities have also expanded beyond monitoring files. Integrity monitoring tools can now monitor a plethora of other systems for changes, including network devices, databases, directory servers, cloud images, containers and cloud management accounts. The ability to monitor these systems provides the technical underpinning of an expansion from FIM to Integrity Management.

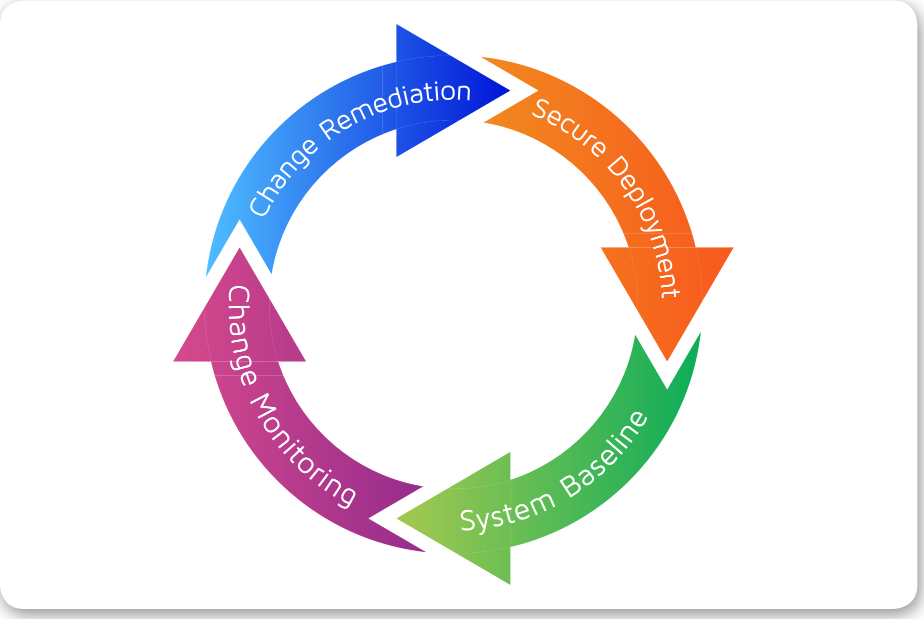

Fig. 2 The four basic steps to ensure integrity

Applying Integrity Management

Traditional IT Environments

File Integrity Monitoring was born out of traditional IT's need to identify changes on discrete systems. It's grown in that environment, so the broader Integrity Monitoring concept is well suited to data centers and other IT environments. The core principles of Integrity Management don't need much adaptation or explanation for implementation in a traditional IT environment.

Organizations looking to implement Integrity Management, or to shift from File Integrity Monitoring to Integrity Management, should start with an assessment of their current processes and capabilities following the cycle described above. All organizations deploy new assets, and understanding how those assets are assembled and configured prior to deployment is the place to begin. Just as it's more expensive to fix a defect once a product is released, it's exponentially harder to secure an asset once it's deployed. Understanding the current process is the place to start, but the objective is to identify a target process for deploying securely configured, compliant assets. In order to accomplish this goal, the deployment context of each asset needs to be identified, including the environment in which it will be deployed, the business objectives it will support and the classification of data it will store and process. These criteria lead to a risk profile and compliance requirements, which must be met in the pre-deployment assessment.

Successfully deployed assets are often the result of multiple teams working together on related objectives to accomplish a task. As such, there is room for error, even with a process for secure deployment. The final step of that process needs to include establishing a production baseline for that asset. Once the work is done and the proverbial switch has been flipped, a snapshot of that asset's risk profile and compliance state should be taken. This baseline is what allows you to identify changes in the integrity of that asset, and therefore of the systems it supports.

Monitoring for change requires that baseline to be known. There are important criteria for how and when to monitor for changes. Real-time detection of changes is important on highly valuable or extremely static systems. In other cases, a more periodic measurement is appropriate to the value of the asset or data. The ability to establish clear policies for monitoring is a key requirement for Integrity Management.

If you only implement the first three steps for Integrity Management, you are guaranteed to be overwhelmed by the sheer volume of changes detected. In fact, this very pattern of data overload is what generated the Security Incident and Event Management (SIEM) industry out of the log management market. Building change reconciliation into the process is the best way to ensure that you can take action on the data produced. In order to effectively reconcile changes, you must have some record or ledger against which you compare the changes detected. There are levels of process maturity here, and each of them yields different results. Organizations can reconcile changes purely based on change windows or based on the combination of a change window and identified asset. These are broad reconciliation processes that yield minimal, but important, value. If more detail is available in the ITSM or change management system, better reconciliation can be performed. Matching changes to specific work orders or tickets is ideal. Matching changes to file manifests in those work orders is the most advanced reconciliation, and results in the most accurate identification of suspicious changes.

Cloud Workloads

It's a common myth that adoption of public cloud infrastructure requires entirely new security controls. The reality is that, while the underlying technology may have changed, the same basic security controls are required for cloud workloads as in a more traditional IT environment. In fact, using Integrity Management as the framework for discussions on how to secure cloud workloads is an effective means to abstract the control requirements from the technology.

Of course, the technology is different for securing cloud workloads. An understanding of the common controls drives the discussion around the technology required to apply those controls consistently. The key technology changes to consider are:

- **Platform support**: virtualized operating systems and cloud-only operating systems (e.g. Amazon Linux)
- Platform-as-a-Service (e.g. Amazon S3, Azure Blobs)

- Multiple cloud providers (e.g. Amazon, Azure, Google)
- Cloud administration accounts

Each of these technology differences points to a product requirement for security controls for cloud workloads. They can be translated into requirements statements.

Does the product/tool:

- Support the platforms we're using for cloud workloads?
- Deliver its described value for the platform services we're using in the public cloud?
- Support the cloud providers we're using?
- Deliver its described value for the cloud management accounts for those providers?

While we might not think of public cloud as an emerging technology, it really is. Many organizations have only dipped their proverbial toes into public cloud, and there are certainly some that haven't. That means that the response to these requirements is likely to have some gaps.

DevSecOps

DevOps is not so much a technology as a methodology that implies certain technology choices. The addition of security into DevOps has produced the somewhat awkward moniker "DevSecOps." While DevOps doesn't require cloud, the two are often linked. Alternatively put, the destination of DevOps deployed services is most often the cloud.

The primary objective of DevOps is to remove friction from the process of delivering value to end-users. DevOps is, by its nature, continuous, flexible, and incremental. IT security has a long tradition of being the opposite. It's hardly surprising that the interjection of security into the DevOps lifecycle hasn't produced the smoothest of relationships.

Integrity Management can bridge the gap between DevOps and Security.

Trying to apply traditional security controls to deployed assets (containers or images) in a DevOps environment is likely to fail. While IT security may see those assets as potential points of compromise, DevOps often views them as disposable extensions of templates. Understanding the dynamic between, for example, container images and running containers, drives IT security to "shift left" and apply controls pre-deployment. That doesn't mean running assets don't get any attention. The objective shifts from runtime controls to validating the integrity of the deployed assets against the pre-deployment desired state. Deviations are destroyed in production and addressed in the template. Production assets are disposable.

Another key to bridging this gap is all in the toolset. The addition of poorly integrated security tools into the DevOps tool chain adds friction. Successful security tools should work with the toolchain, ideally adding process without friction, and value with minimal operational change. For example, scanning pre-deployment images for vulnerabilities is a good action to take. Requiring that each image is deployed in a staging environment, running a scan, and producing a report that DevOps engineers have to review to patch vulnerabilities is a process doomed to failure. Vulnerability scanning should be an inline process, and the results should be integrated into the tools in the continuous integration and deployment pipeline.

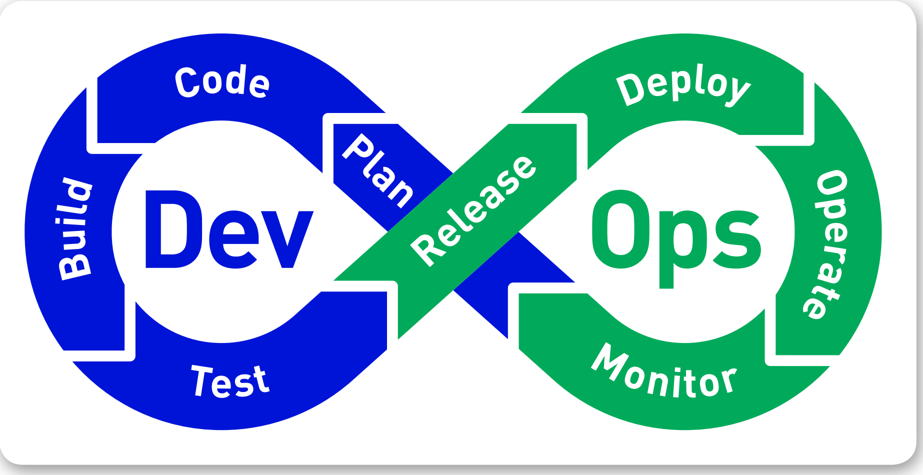

Fig. 3 The DevOps workflow

Security Benefits of Integrity Management

Every incident begins with a change. That change may be internal or external. It may be malicious or accidental. Regardless of the attributes, there's a change at the root of every incident. Integrity Management is, at its core, the identification, investigation and remediation of change. Focusing on managing the integrity of the systems on which your business relies provides a foundation upon which security programs and processes can be reliably constructed. That's all well and good, but what are the real, tangible benefits of using Integrity Management in this foundational capacity?

Reducing the Frequency and Severity of Incidents

This benefit alone is sufficient to merit a serious look at Integrity Management. Information security is about managing risk, and reducing the frequency and severity of incidents is, perhaps, the best set of metrics to determine the efficacy of a program. Using Integrity Management to detect and remediate changes reduces the risk surface in your environment. Keeping configurations secure, ensuring that unauthorized software is detected, and managing vulnerabilities are all components of Integrity Management that help reduce risk.

Faster Time to Recovery

The biggest barrier to recovery from an incident is the difficulty in determining a root cause. Imagine if you could start root cause analysis by examining what changes actually occurred between operational state and incident identification. Focusing on Integrity Management brings along that level of resolution on changes, even historical changes. If you need last month's changes in order to determine root cause, you have them. Faster root cause analysis results in faster recovery.

More Accurate and Complete Investigations

Root cause may lead to recovery, but it's rarely the end of the investigation. It's difficult, if not impossible, to produce forensic data that you've never collected in the first place. Log data is the most crucial tool for investigations, but it often fails to paint a complete picture. By establishing baselines and monitoring for changes, organizations can use the principles of Integrity Management to move from logged events to detailed changes easily.

Compliance Benefits of Integrity Management

The security benefits of Integrity Management are clear. You are significantly more likely to prevent, detect and effectively investigate incidents if you apply the principles of Integrity Management. Beyond those benefits, however, the core capabilities are also required by a variety of regulatory standards. Regulatory requirements can often be used to secure budget, so it's valuable to identify capabilities that can be funded by compliance and deliver additional security benefit.

The Payment Card Industry Data Security Standard (PCI DSS) has a long history of requiring File Integrity Monitoring specifically. More recent versions of the standard have shifted from the FIM language to a broader requirement for change detection. This shift is well in line with the growth of FIM from a point tool to a more comprehensive change detection capability. The current version of requirement 11.5 reads:

> Deploy a change-detection mechanism (for example, file-integrity monitoring tools) to alert personnel to unauthorized modification (including changes, additions, and deletions) of critical system files, configuration files, or content files; and configure the software to perform critical file comparisons at least weekly.

Another good example of regulatory requirements comes from the U.S. electric utility industry with the NERC Critical Infrastructure Protection standards. NERC CIP-10 doesn't specifically discuss integrity, but describes requirements for establishing baselines for systems and monitoring them for change. NERC CIP also includes requirements for vulnerability assessments. Integrity Management is embedded throughout the standards.

The standard that tends to matter most across U.S. government is NIST 800-53. Not only does the venerable 800-53 contain an entire family on System Information and Integrity Controls, but SI-07 explicitly calls for the use of "integrity verification tools to detect unauthorized changes" on a variety of organization defined objects. NIST 800-53 cascades down through multiple other standards, both commercial and government. While integrity verification is required for SI-07, the basic ability to detect and reconcile changes in an environment delivers validation of many other controls. You don't have to go too deep into 800-53 to discover this benefit. AC-02 focuses on account management, and while Integrity Management isn't going to "Create, enable, modify, disable, and remove information system accounts in accordance with policy," it will give you the ability to capture every single one of those changes for investigation and audit.

Outside the U.S., ISO 27001 is a broadly applied standard for information security. Not surprisingly, the ISO standard also includes requirements fulfilled by Integrity Management. In some cases, integrity is specifically addressed in the requirements. 10.5 requires that organizations "maintain the integrity and availability of information and information processing facilities" through the use of backups, which is a good example of how Integrity Management extends to specific controls. 10.9.3 is more specific: "The integrity of information being made available on a publicly available system shall be protected to prevent unauthorized modification." The specification of both "data" and "unauthorized modification" identifies this requirement as classic integrity monitoring. Cryptography is addressed in 12.3.

In order to understand how Integrity Management applies more broadly to ISO 27001, it's worth examining the change control requirements in 10.1. Here we have requirements for a change management process (10.1.2) and segregation of duties (10.1.3). Both of these requirements are more easily fulfilled with accurate baselines and change detection in place. 12.5.2, part of the "security in development and support processes" section, specifically calls out the need to review and test business critical applications after operating system changes. That requirement is really only possible with a means to detect and reconcile those changes.

Conclusion

Organizations that can shift their mindset from a piecemeal security approach to risk management to a holistic Integrity Management approach to risk management will start seeing benefits that span security, compliance and IT operations. The principles of Integrity Management are even more valuable in the fast-changing world of DevOps, driving increased automation of the four key steps throughout the continuous integration and deployment pipeline. Whether you call it File Integrity Monitoring or Integrity Management, this capability is both required and foundational.

Tripwire is a leading provider of security, compliance and IT operations solutions for enterprises, industrial organizations, service providers and government agencies. Tripwire solutions are based on high-fidelity asset visibility and deep endpoint intelligence combined with business context; together these solutions integrate and automate security and IT operations. Tripwire's portfolio of enterprise-class solutions includes configuration and policy management, file integrity monitoring, vulnerability management, log management, and reporting and analytics. **Learn more at** tripwire.com

The State of Security: Security News, Trends and Insights at tripwire.com/blog
Follow us on Twitter @TripwireInc » **Watch us at** youtube.com/TripwireInc