Realtime publishers

Monitoring, Detecting and Preventing Insider Fraud and Abuse

Dan Sullivan

sponsored by

Attachmate

Copyright Statement

Chapter 3: Effective Techniques for Preventing Fraud and Proving Compliance

Virtually every business must confront the risk of fraud and abuse. Insiders with detailed knowledge of business processes are in positions to exploit that knowledge to commit fraud. Chapter 1 detailed types of abuse and their costs to the business, including financial fraud, sabotage, and loss of privacy. Chapter 2 examined the technical barriers to monitoring, detecting, and investigating insider abuse. In this chapter, we turn our attention to techniques that can counter the advantages business insiders possess.

The techniques described in this chapter will work even if insiders are able to circumvent the access or logging controls of any single system. That's because their activities will eventually leave traces that can be collected and analyzed. For example, an insider committing financial fraud can avoid triggering database alerts designed to detect and log high-value transactions by using a program to issue multiple low-value transactions that together equal the value of a single high-value transaction. Each of those database transactions by itself is not suspicious, but the entire set of multiple transactions becomes increasingly noticeable as the number of such transactions grows over time. Although existing database monitoring methods may not detect this pattern, application-level monitoring can (see Figure 3.1).

This chapter considers four key areas for mitigating the risk of insider fraud and abuse:

- Multi-channel monitoring
- Application activity analysis
- Information security response
- Demonstration of compliance

As their names imply, the first three areas constitute the monitoring, analysis, and response phases one would expect in any ongoing process to detect and prevent security breaches. Demonstrating compliance is not strictly required to control the threat of abuse itself. Rather, it is a governance requirement that is equally well served when subjected to the same techniques used to control insider abuse.

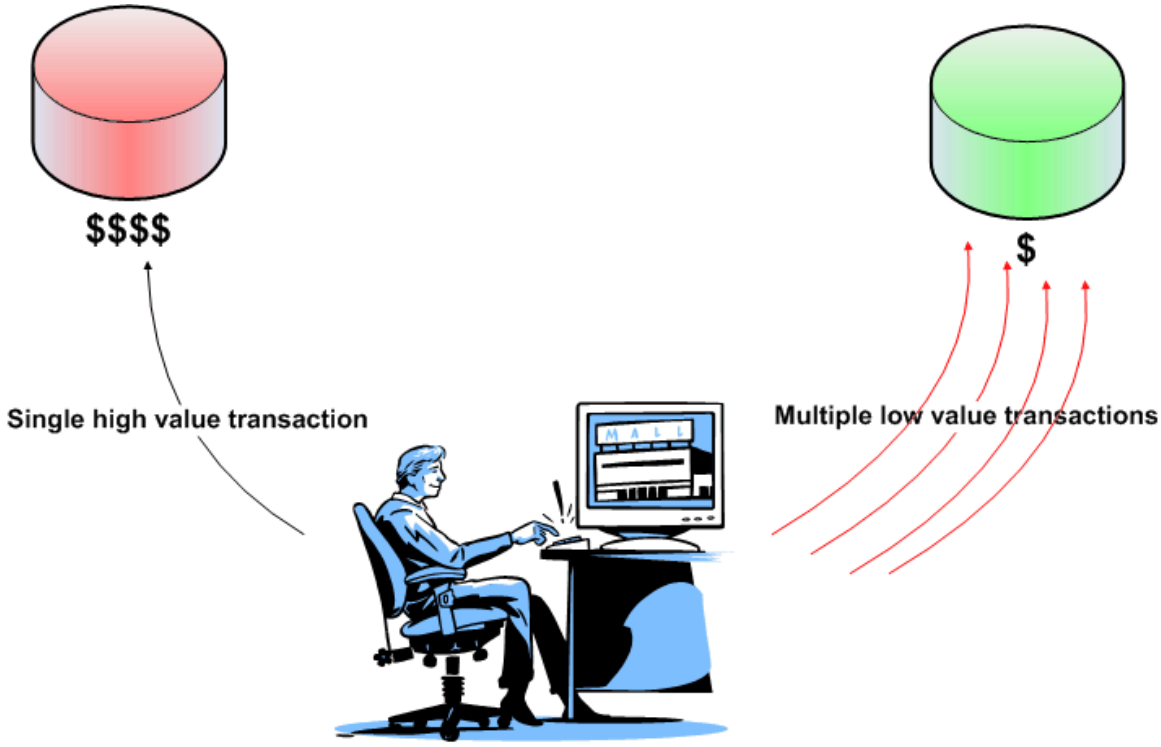

Figure 3.1: Insiders can avoid single application component controls, such as database logging, but they still leave detectable patterns of activity at the application level.

Multi-Channel Monitoring

One of the things security professionals do is to figure out as many different ways of breaching security as possible. This applies to those involved with both physical and electronic security.

Identifying Target Systems of Insider Abuse

Imagine a security manager for a chain of retail stores who is charged with minimizing the cost of stolen goods. There are several obvious ways thieves could commit their crimes:

- Shoplifters could hide small goods in their pockets or bags and walk out the front door
- Employees working the stock room could hide goods until the end of their shift and leave through an employee entrance
- Someone making a delivery could simply fail to unload the full delivery and drive away with the missing goods
- Burglars could break in through a skylight, air duct, or some other illicit passageway

The front door, the employee entrance, the delivery dock, and the unintended passageways are all different channels through which goods can be stolen. If we put all of our efforts into controlling one or two of these, say with anti-shoplifting security and surveillance cameras at the employee entrances, we risk missing theft that occurs through the other channels. Similar weaknesses can hamper application security if we focus only on individual components rather than on multi-channel activities.

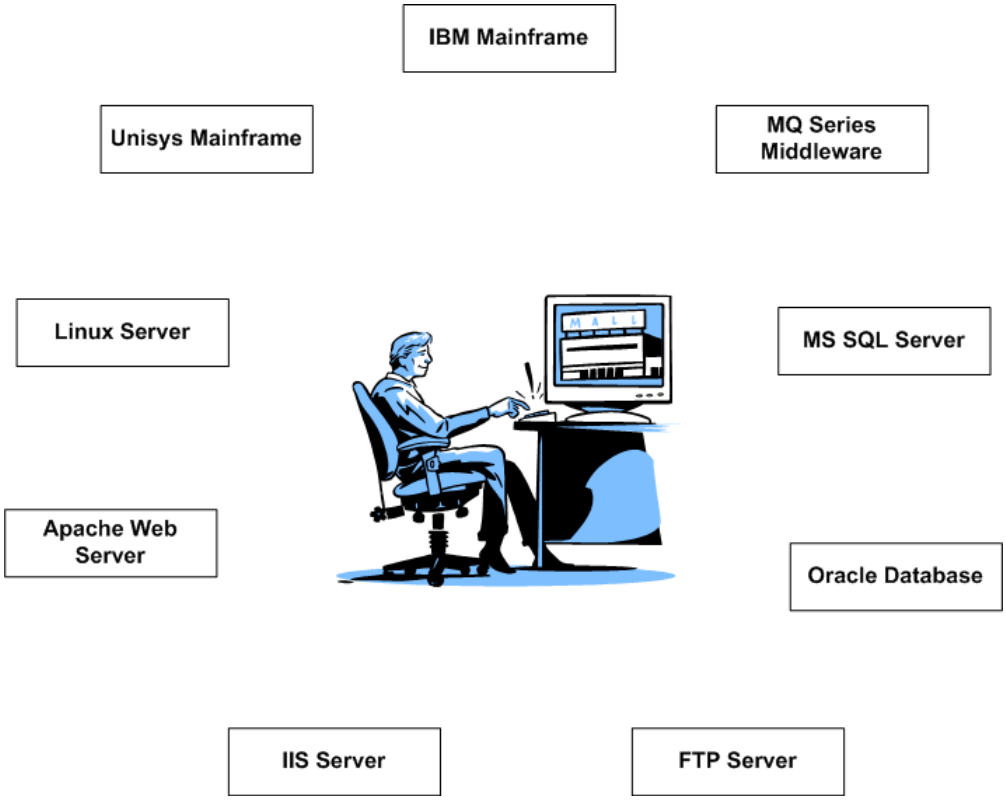

Figure 3.2: Insiders can take advantage of multiple channels to commit fraud using business applications.

Just as thieves do not have to walk through the front door to rob a store, insiders do not have to use the obvious access method to commit fraud. Also, unlike crimes committed on tangible goods, fraud can be perpetrated over multiple channels. An insider, for example, could issue transactions through an application, insert transactions into a middleware messaging service, and directly update a database through a malicious script all in service to a single act of fraud. The beauty of this type of approach, from the criminal's perspective, is that these individual actions are designed to avoid triggering alerts at the application, middleware, or database level.

The first step to effectively monitoring for insider fraud and abuse is to identify systems that are likely targets of insider abuse. These can fall into multiple categories:

- Databases that maintain information on financial and physical assets, such as accounts receivable, accounts payable, and inventory management
- Messaging systems used to communicate transaction information between distributed systems
- User interface (UI) applications, such as Web applications, that are provided for interactive activities
- Application servers that host Web services or other programs that provide specialized functionality to multiple business processes

It is important to remember that insiders can take advantage of various types of software infrastructure, not just the obvious candidates like end user applications. For example, an insider with sufficient knowledge of application design could inject transactions in the middle of a workflow rather than at the beginning. For this reason we need to monitor activities across multiple channels.

Monitoring Application Sessions

Once we have clearly identified the systems that are potential targets for abuse, we should monitor and record information on all activities on those systems. This may sound like a daunting task, but it is necessary to gather sufficient raw data about events within business systems.

The Limits of Logs

Application activity data is more detailed than data collected in basic logging because application activity logging can be done at the network level. Basic logging is limited to well-defined events, such as updating an account balance. This is certainly useful information in many situations, but it is insufficient for fraud monitoring. It does not, for instance, provide very detailed information on the context of the logged event. A log record might include:

- Previous account balance
- New account balance
- Username of the person updating the record
- Time of the update

Many pieces of potentially useful information are unavailable to us. These include:

- What was done from that IP address prior to and after that update
- What other transactions were performed by that user with another application
- If that user performed activities not normally associated with his or her role, such as exporting large amounts of data

Furthermore, we should assume that users have the knowledge and ability to avoid detection based on information provided by application logs.

So we are at the point where we have a limited mechanism for monitoring suspicious activity and an insider with the knowledge necessary to avoid triggering alerts. What is to be done? We cannot alter the insider's knowledge about logging, so we will have to change how we perform monitoring.

Beyond Logging: Multi-Channel Application Activity Monitoring

An effective response to the limits of logging and the ability of insiders to avoid some forms of detection requires a monitoring strategy that is based on three principles:

- Monitor all application components and all activity that makes use of those components
- Collect a rich set of attributes about each activity
- Monitor at the lowest level possible—that is, at the network level—to mitigate the risk of tampering at higher levels

We have already discussed the first principle, so we will turn our attention to the other two.

Collecting Activity Attributes

Data is often useless without information about the context. For example, if a sales report simply listed 10,000 units sold, it would be virtually worthless. We would need to know 10,000 units of which product, over what specific period of time, in what geographic locations, and through which channels. Similarly, knowing that a record containing private financial information was read is insufficient to determine whether a privacy breach has occurred. In such as case, we would want to know:

- Who read the record?—This would be used in conjunction with information about their role in the business, which may require them to view such data.
- When was the record read?—Reading such a record outside of normal business hours is somewhat suspicious but perhaps insufficient on its own to warrant concern.
- What application was used to read the record?—Reading it with an application other than the end user application commonly used for this type of operation would be somewhat suspicious.

If two or more of the attributes are suspicious, further investigation might be warranted. However, if we are assuming the insider has knowledge of alert triggers, the insider might intentionally avoid two suspicious attributes. For example, the insider might view the private data using a non-standard application but do so during normal business hours and using an account that is authorized for that operation.

These attributes are also useful in aggregate. A single event or activity may not be suspicious, but a number of similar events may be. If the insider has viewed private data at rates above average for her role in the company, it may be cause for further investigation.

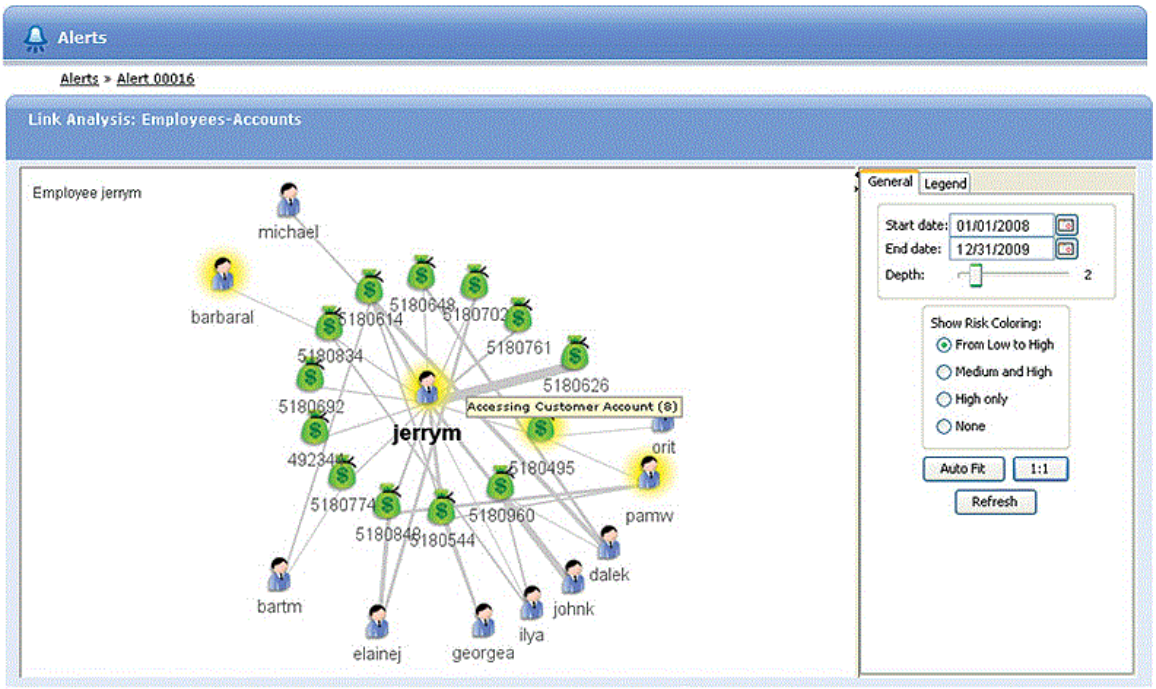

Figure 3.3: Users with above-average levels of activity in monitored transactions may require further investigation.

Monitor at the Network Level

The popular programming language Perl is known for flexibility; in fact, one its best known slogans is "There is more than one way to do it." The same could be said for committing insider fraud and abuse. A malicious insider can use end user applications, database utilities, development tools, or custom programs to manipulate applications and data in ways that were not intended by their designers. At some point, though, the malicious activity will have to transmit a command from a point of origin to the target system, and that means using the network.

Direct Connections

It is also possible to manipulate a data source or application via a direct connection. For example, a mainframe administrator may have access to a direct connect console that does not require transmission over the network. In such cases, the keystrokes or commands can be captured on the mainframe for later analysis and playback. The mechanism is different, but the principle is analogous to monitoring at the network level.

Capturing data at the network level presents several advantages from a monitoring perspective:

- Applications do not have to be modified to record user activity
- There is reduced risk that malicious insiders will avoid detection by using an application that is not monitored
- If an attacker uses multiple applications, utilities, or development tools, data is captured and recorded in the order in which those programs were used
- Virtually all communications between a malicious insider and target applications will have to use the network or a dedicated communication channel, such as a direct connection console, making it difficult to avoid monitoring

Multi-channel monitoring is an essential first step to effectively detecting and preventing fraud and insider abuse. We are assuming that insiders have specialized knowledge about logging and auditing methods and will work to avoid detection. By monitoring all potential targets and capturing activity data at the network level, we can collect a sufficiently descriptive set of data to detect patterns of abuse.

Collecting data with multi-channel monitoring is just the first step in the process. The next step is to mine potentially large volumes of data for indications of fraud and abuse.

Application Activity Analysis

Data is not information. Multi-channel monitoring provides us with raw data but not information we can act upon. The goal of application activity analysis is to derive such information from monitoring data. The process entails three essential steps:

1. Specifying patterns of abuse
2. Detecting potential abuse patterns in data
3. Analyzing findings to determine actual cases of fraud and abuse

The first step creates filters for detecting fraud and abuse, the second step applies those filters to raw data, and the final step applies more in-depth analysis to the most likely candidates of fraudulent or abusive activity.

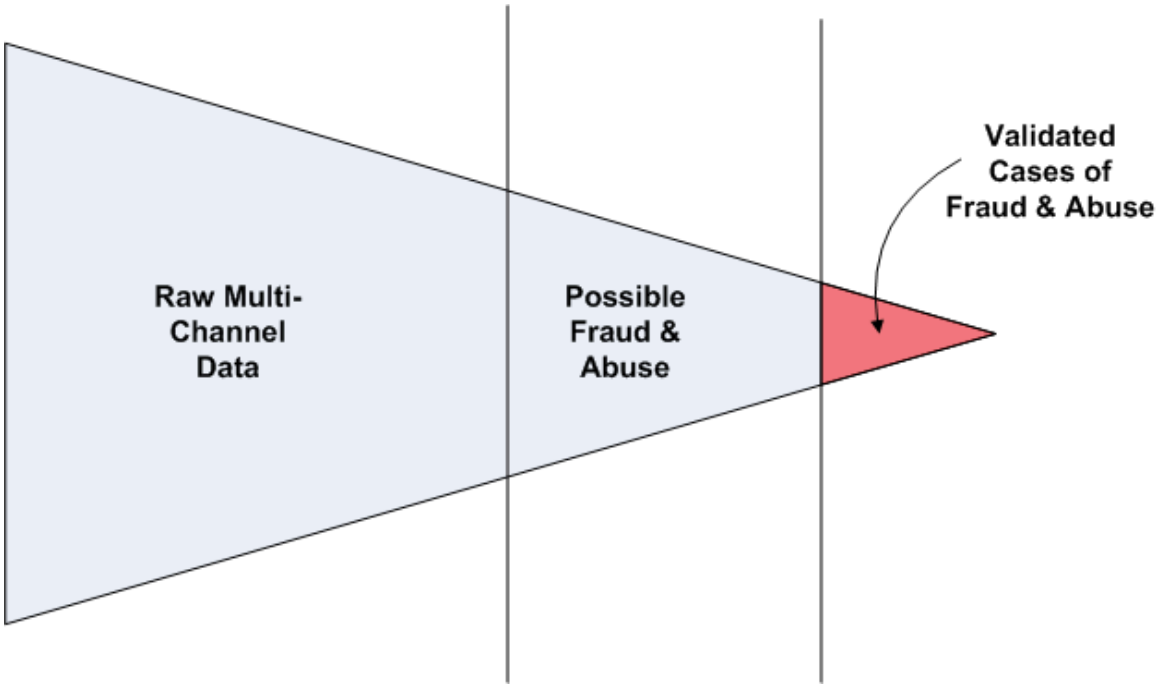

Figure 3.4: The three-step process of analyzing multi-channel data narrows the investigative focus to the most likely candidate of fraud and abuse.

Although malicious insiders may be able to avoid detection by single system logging and monitoring, they do leave traces in the activity record when there is monitoring occurring on multiple channels at the network level. An added benefit of this monitoring is that it provides a wealth of data on what patterns appear when normal, routine business processes are executed. This is a valuable set of data for defining normal and suspicious activities. Patterns of abuse can be defined in two ways: as patterns of known abuse and as variations from normal patterns of activity.

Known Patterns of Abuse

Businesses have always had to protect financial assets from insiders looking to commit fraud. In the past, this may have meant locking up cash in a safe at the end of the day. Today, we have to protect the electronic realizations of those assets as well. Money is not the only target, either. Financial, medical, or other personal information about celebrities, politicians, and well-known business executives are targets of prying eyes.

> **Celebrity Snooping**
>
> Private information about well-known personalities is grist for the tabloid mill. Employees of businesses and institutions with access to such information can satisfy their own curiosity as well as become the source for media outlets that thrive on such details. For example, the *Los Angeles Times* reports Ronald Reagan UCLA Medical Center was fined $95,000 for not preventing two employees and two contractors from accessing the private information of a patient. The UCLA hospital had previously fired or suspended 13 employees for accessing a pop singer's patient records without authorization, according to the *New York Times*.

Unauthorized access to private information and excessive activity in financial systems are just two examples of suspicious activity. One way to detect these types of suspicious activity is to look for patterns that are well outside the average range of activity. Consider the data depicted in Figure 3.5.

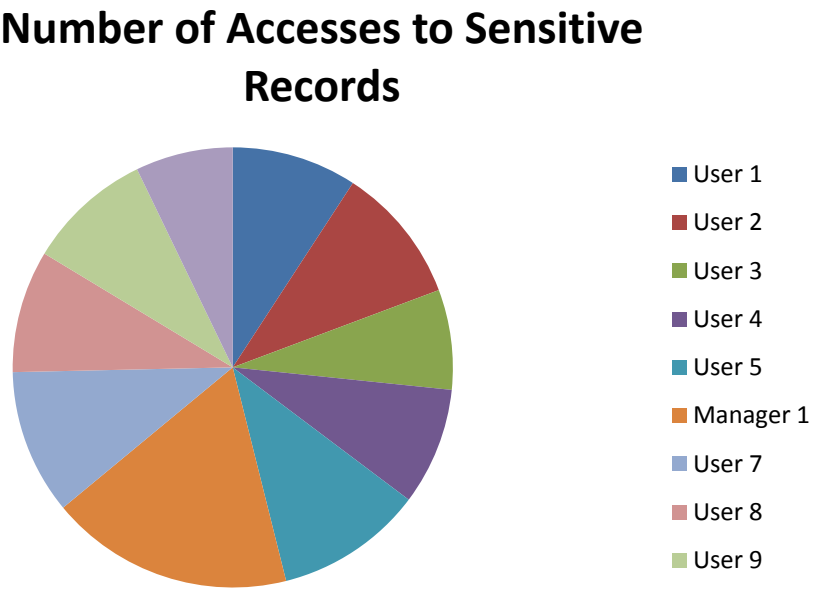

Figure 3.5: Well above-average actions on sensitive or protected data may be indicative of malicious activity.

In Figure 3.5, Manager1 has significantly more access attempts to sensitive data than do other users. One possible explanation is that Manager1 is also a manager with responsibilities for more sensitive information than other types of employees. In that case, a better measure is to compare users in that role to each other (see Figure 3.6).

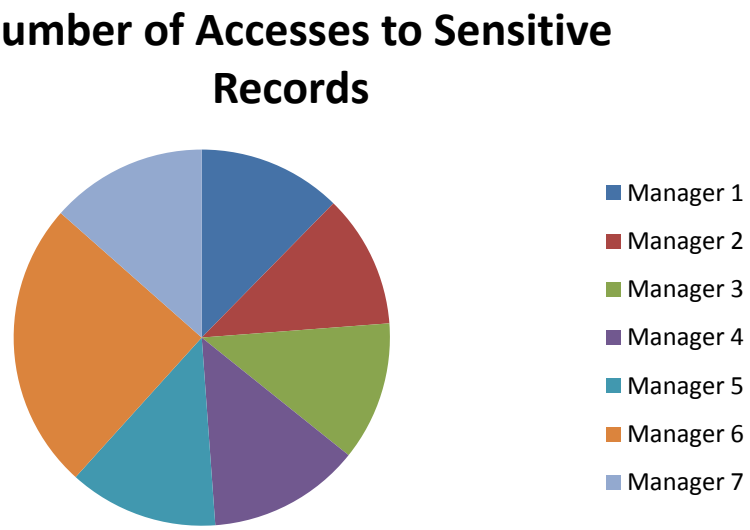

Figure 3.6: When comparing users, activities, events, and so on, it is important to consider populations with common characteristics, such as access control privileges and job responsibilities.

In this example, even when comparing Manager6 to a group of peers, it appears that this person has an unusual volume of activity with sensitive data.

It is not always obvious when fraud and abuse occur, especially when someone is trying to hide their malicious activity.

Variations in Patterns of Normal Activity

Many business processes entail a common sequence of activities. This is especially true for automated procedures. The fact that patterns of activity are repeated is actually an advantage when analyzing monitoring data. Take this workflow, for example:

- User authenticates to a Web application
- User searches for a customer record
- User issues a transaction against that record in the Web application
- Web application submits a message to a messaging queue
- Messaging service delivers message from the queue to a back-end application
- The back-end application updates the database
- The back-end application returns a result status code through the messaging service back to the Web application

There is a fixed set of steps that involves a user application, a middleware message service, and a back-end database. Now imagine someone trying to commit fraud by exploiting their knowledge of how the messaging system works. Rather than enter fraudulent transactions through the user application, the insider develops a program to write them directly to the messaging queue. From there, the transactions are processed by the database. The status response message is removed from the queue by the fraudulent application as well.

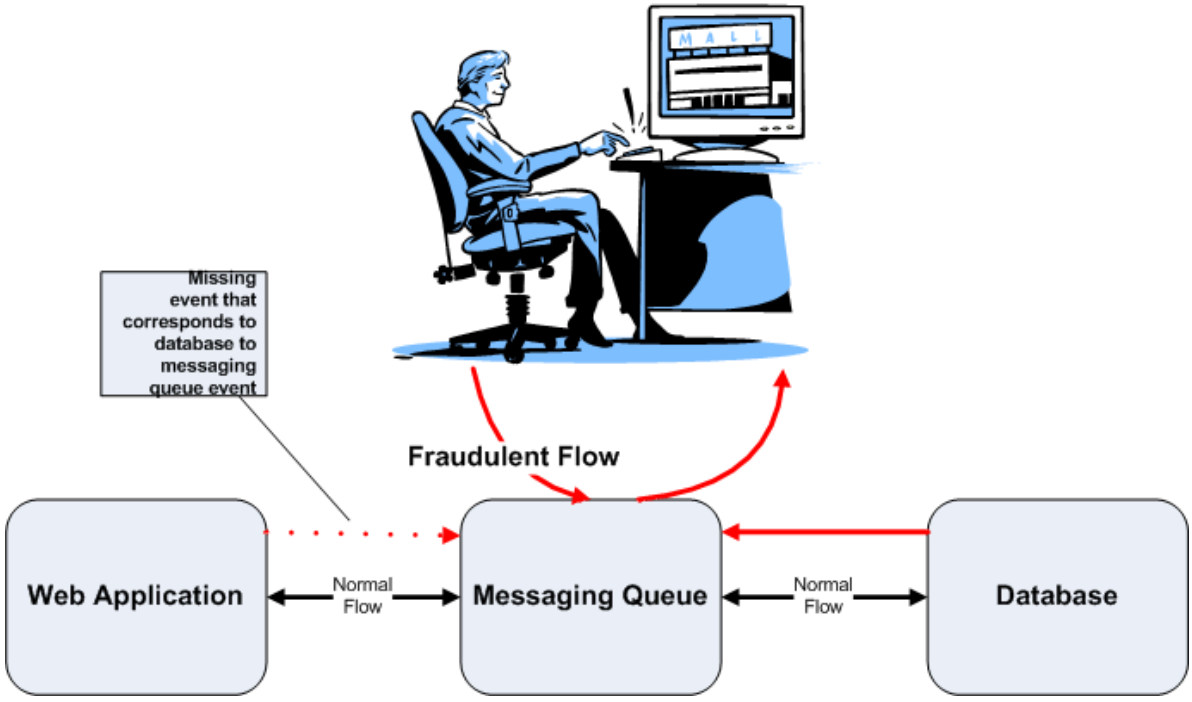

Figure 3.7: Missing events can be detected by analyzing patterns in multi-channel, network-level monitoring data.

This example illustrates the kind of analysis that can be performed when we have comprehensive monitoring at the network level. We see from this example that we are not dependent on logging by the database or the message queue. This comprehensive monitoring avoids the sometimes complex process of integrating logs from multiple systems to create a comprehensive picture of activity at the application level.

With sufficient data, we can formulate a baseline of typical patterns of activity. Variations from these baseline activity patters may be indicative of fraudulent or malicious activity. As we can see from the two preceding examples, during the analysis phase, we can look for patterns of known abuse, such as frequently accessing sensitive information, or look for patterns that do not fit with the typical patterns of legitimate business operations.

In some cases, these patterns (or heuristics, as they are sometimes called) may be industry specific or general. For example, both a hospital and a bank could use similar heuristics for detecting celebrity snooping. This is not always the case. A financial institution, for example, will require heuristics for detecting suspicious activities that include wiring funds to outside institutions. Regardless of the type of heuristics used, analyzing monitoring data with these patterns is not the final word on detecting fraud and abuse. Once alerts have been triggered about suspicious activity, a more-detailed analysis of findings is required.

Analyzing Findings

Multi-channel monitoring at the network level is a valuable tool for detecting and preventing fraud and abuse, but it is not infallible. The output of such systems are indications of anomalous activities that warrant further investigation; they are not, by themselves, definitive proof that fraud or abuse has occurred.

Looking Before You Leap: Understanding the Context of Suspicious Events

Consider the previous example about a manager with an unusually high volume of activity with sensitive records. This could have been a case of celebrity snooping or there may be a legitimate explanation; perhaps the manager was

- Conducting an audit of sensitive information to ensure access controls were properly configured
- Responsible for a territory with a disproportionate number of VIP customers, such as Los Angeles or Washington, DC
- Investigating a privacy breach

In all of these cases, there is a broader business context that even the most comprehensive multi-channel monitoring system could capture. The broader context will help us understand the limits to multi-channel monitoring, including two fundamental problems anytime we use past patterns to categorize current and future events: false positives and false negatives.

False Positives: Accusing the Innocent

At its most basic level, a multi-channel monitoring and analysis system is a classification system. It categorizes events within business systems as either normal or suspicious. (Actually, there may be grades of suspiciousness, but we will oversimplify here for a moment.) When we categorize something as being in a category to which it does not belong, that is a false positive. Examples of false positives are generated when

- The characteristics of normal baseline behavior changes over time without updating monitoring heuristics
- Normal baseline behavior for one line of business, territory, or set of employees is significantly different from the baseline for the rest of the organization
- Thresholds for categorizing events as suspicious are relatively loose (this is not necessarily a problem and may be done intentionally; we will return to this issue when discussing false negatives)

Similar limitations in the application of classification techniques lead to the problem of false negatives.

False Negatives: Getting Away with Fraud

A false negative occurs when an actual fraudulent event takes place and is missed by the fraud detection system. This occurs when the heuristics and pattern recognition techniques are insufficient to correctly categorize the fraudulent activity as such. It is not hard to imagine how this can happen.

Consider the example of the fraud that inserted transactions into the messaging middleware bypassing the user interface. If the insider had sufficient knowledge about the way the application worked, he might use a program to simulate inserting a transaction into the Web interface. Since there is no need for the interface to actually submit the transaction for processing, the insider might craft a malformed transaction that is rejected by the middleware application. This will leave a trace in the stream of network activity. Unless the analysis heuristics delve into the detailed attributes of the event, they may miss the fact that this event was rejected by the middleware.

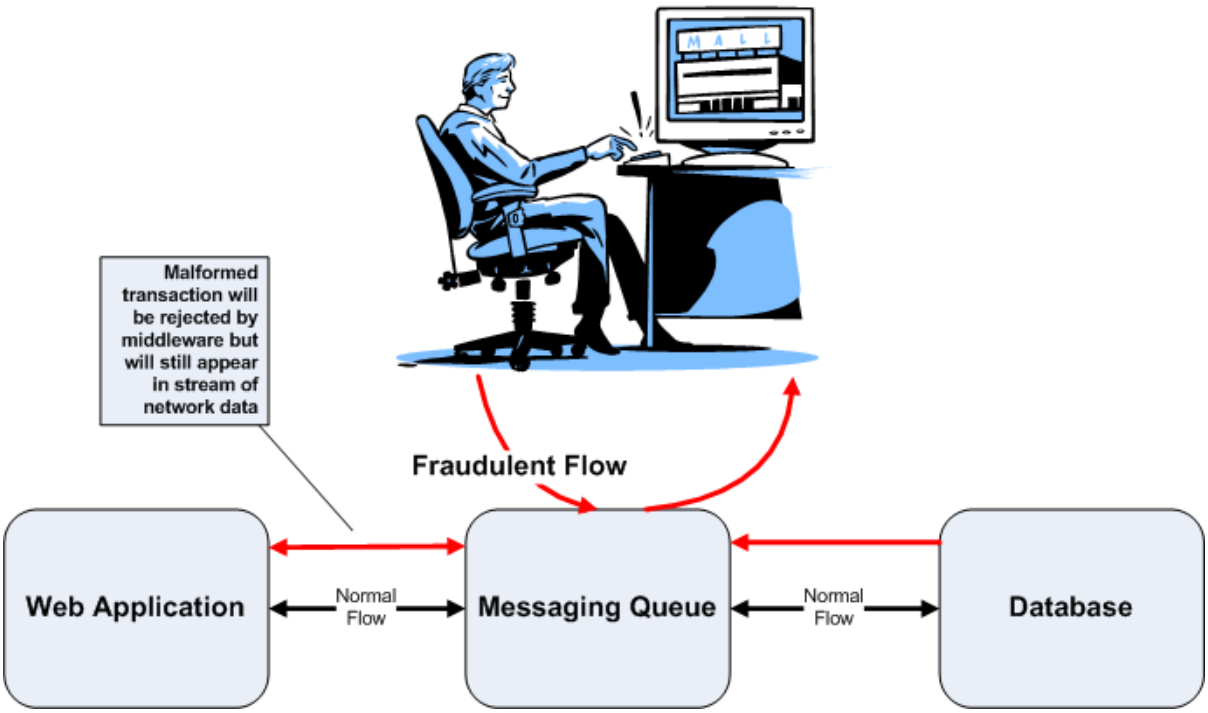

Figure 3.8: To avoid detection, an insider might create bogus events to generate events in the stream of application events. Unless the analysis techniques look into details of the event, false negative classifications might result.

With the risk of false positives and false negatives, it is important to verify potentially fraudulent or abusive behavior with other data. This can require further investigation to gather corroborating evidence, such as indications that new application code was deployed to a production server outside of the normal deployment process, the insider in question has recently changed privileges, or the potentially fraudulent transactions are linked to an account with improper supporting documentation.

Application activity analysis begins with collecting data through multi-channel monitoring, analyzing it with a set of heuristics and pattern recognition techniques to identify potentially malicious or fraudulent activity, and verifying the findings. In cases where fraud or abuse has occurred, an appropriate security response should follow.

Information Security Response

Acts of fraud or abuse call for both immediate and longer-term responses. A common set of responses includes:

- Incident response
- Forensic investigations and case management
- Post-event assessment and policy review

The goals of information security response are applicable to a wide range of security breaches. We cannot examine the full scope of what a response would entail, so we provide a brief overview instead.

Incident Response

Incident response is the first set of actions taken to control the impact of a security breach. The first priority is to prevent further damage beyond what has already occurred. In the case of financial fraud, we want to prevent any further transfer or loss of funds. When privacy has been breached, we want to block the perpetrator's access to sensitive and private information. There are many ways to implement these controls. User accounts can be disabled and privileges to execute applications can be withdrawn. The insider's devices, such as desktop and laptop computers, can be blocked from accessing the network. In cases where the source of the leak or fraud is not immediately apparent, entire applications or databases can be rendered inaccessible to any user. This is an extreme measure but one that might be warranted in some circumstances.

Another top priority for an incident response is to preserve evidence. Once the immediate threat to the business and its customers is contained, security professionals will want to begin gathering evidence. Ideally, these professionals are following an incident response plan that is already in place. These plans describe procedures for collecting and preserving evidence, notifying IT and business executives of the security breach, and undertaking other steps to support forensic investigations.

Forensic Investigations

How did it happen? The answer may come from a forensic investigation in which IT professionals, auditors, and others familiar with the security breach will reconstruct the events that allowed the fraud or abuse to occur. The output of a forensic investigation will include:

- Description of the sequence of events initiated by the perpetrator
- Information about the applications and hardware used to commit the fraud or abuse
- A list of possible parties involved, which in some cases might involve unknown persons
- Vulnerabilities in applications and weaknesses in business procedures that were exploited

The purpose of the forensic investigation is to provide as many details as possible about what occurred and how. The next step is to decide what to do about these weaknesses.

Post-Event Assessment and Policy Review

The last stage of the information security response is to determine what, if any, changes are required in policies, procedures, or application design to prevent similar types of fraud from occurring again. A cynic might deride this process as simply "closing the barn door after the cows are gone," but that attitude misses the point.

Successful businesses learn and adapt. Risk management strategies may have missed something that is highlighted by the forensic investigation. Application developers may have been unaware of a vulnerability in a custom program. Patch management procedures may not have been executed correctly leaving a critical server vulnerable to compromise. We could go on with more examples, but the point is that in spite of our best efforts, we still have to work with complex systems that harbor unknown vulnerabilities, and we occasionally make mistakes when executing even the best-designed procedures.

In the post-event assessment, we have an opportunity to take the information gleaned from the forensic investigation to identify weaknesses in our systems and procedures. This is an opportunity to revise policies and procedures to avoid the same kind of fraud or abuse in the future. If we are fortunate, these changes will also help mitigate the risk of other means of fraud and abuse as well.

Demonstrating Compliance

Throughout this chapter, the focus has been on techniques for detecting fraud and abuse. An added benefit of deploying a multi-channel monitoring and analysis solution is that it also supports governance and compliance efforts.

The monitoring policies and procedures are relevant to ensuring compliance with a range of regulations, such as the Sarbanes-Oxley (SOX) Act, the Payment Card Industry Data Security Standard (PCI DSS), the Gramm-Leach-Bliley Act (GLBA), and the Health Insurance Portability and Accountability Act (HIPAA). The use of heuristic event detection provides the flexibility needed to adapt monitoring procedures to ensure compliance with regulation-specific requirements. Also, as discussed in the previous section, multi-channel monitoring can be integrated into other security controls and procedures. In particular, it can enhance auditing and monitoring procedures while collecting valuable data for incident response activities.

Summary

Effective controls for preventing fraud and abuse make use of multi-channel monitoring. Insiders with knowledge of business processes and application implementation details can and will exploit multiple channels, such as mainframe applications, client-server programs, Web interfaces, database vulnerabilities, and any number of other pieces of IT infrastructure and software. By combining a comprehensive monitoring program with application activity analysis, businesses can detect, analyze, and block fraud and abusive activities. In some cases, fraud and abuse will occur but the multi-channel monitoring system can provide a wealth of data useful for forensic investigations. Ultimately, that information can be used to improve the policies and procedures that protect critical business operations.

Download Additional Books from Realtime Nexus!

Realtime Nexus—The Digital Library provides world-class expert resources that IT professionals depend on to learn about the newest technologies. If you found this book to be informative, we encourage you to download more of our industry-leading technology books and video guides at Realtime Nexus. Please visit http://nexus.realtimepublishers.com.