Detecting Data Breaches and Information Loss Using Network Behavioral Anomaly Detection (NBAD) Technology and Multivariate Event Correlation

Executive Summary

Many organizations accumulate large stores of *non-public information* (NPI) – customer identity data, credit card numbers, medical records - that is subject to myriad regulatory requirements. Detection of data breaches – both intentional and accidental – is a critical requirement for many security programs faced with regulatory requirements. Reliable detection of such breaches using traditional methods and technologies has proven to be wanting. We present methods of behavioral detection of data loss and misuse using QRadar, a security information manager (SIM) technology with behavioral threat detection capabilities. A case study is presented complete with an example implementation and discussion of eliminating potential false positives. Multivariate correlation –corroboration of multiple detection techniques including behavioral and other detection methods including log based detection - is also discussed.

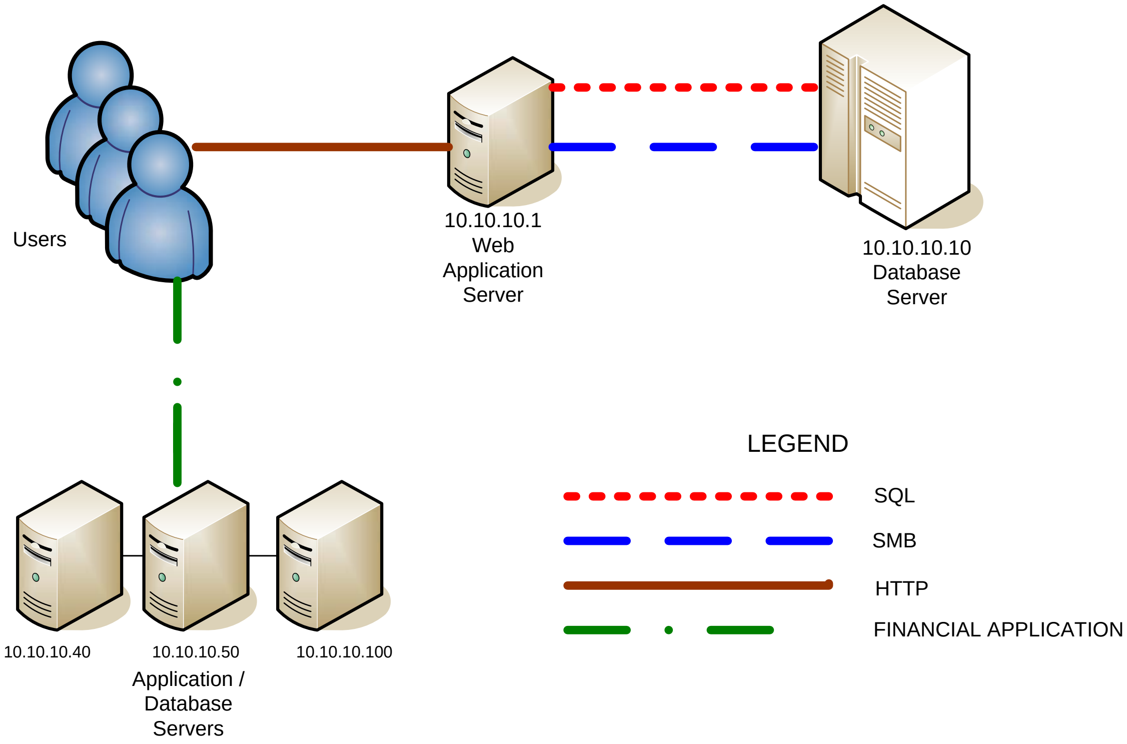
The Problem

Data loss, theft, misuse or “breach” detection is often the result of secondary indications or investigations taking place after the fact. Despite the fact that data loss detection is one of the capabilities that many security teams would most like to implement, reliable detection is often elusive. What makes data loss detection so difficult? There are many reasons.

- Data loss may be accomplished using a vulnerability in a web application – particularly in cases of custom web applications - not known to exist prior to the incident. These attacks may not detected by any products, particularly in uncommon or customized web applications, in which case no obvious attack indications or alarms may be available.
- In cases where web application servers are internet facing, there may be no scanning or reconnaissance alarms if the web applications are easily found or if such alarms have been tuned out.
- Data loss may be occurring through legitimate applications as the result of a misconfiguration in a middleware or client application. In this case, there may be no clear exploit or attack traffic to alert on.
- Data loss may be perpetrated by a legitimate credentialed user, or a hijacked desktop belonging to a credentialed user, in which case no failed authentication alarms may be available.
- Data may be escaping in a binary or encrypted format, preventing inspection by regular expression and content matching tools.

Behavioral Detection

How can we detect data loss in the absence of attack indications or obvious suspicious activity? The answer is to follow the data. While the possible avenues of abusing and misusing applications may be supernumerary, the number of potential avenues for data to escape is finite and quantifiable. In many cases, behavioral detection can be accomplished using flow data. Consider the transaction procession systems diagramed below. In this case, credit card transaction processing is centralized in a handful of systems and NPI data – PCI regulated data, in this case - is concentrated as a result. The expected behavior of the transaction processing systems is quite limited; during normal usage, the users access the systems through a web application, fat client, or “green screen” terminal. No other application traffic should exist between the users and the transaction processing systems – and certainly no SQL or file transfer traffic should take place between these transaction processing systems and end users or other unexpected parties. All of this network behavior, as well as any unexpected deviation from it, can be monitored by collecting *flow data* – a sort of network audit trail – from the intermediate network devices.



In this scenario, we can perform behavioral detection of data loss or misuse by applying the following detection policies to our transaction processing systems using flow data:

- Detect and alert on unauthorized access to database systems by detecting unauthorized SQL sessions.
- Detect all remote administration sessions so they can be corroborated with change controls and expected behavior.
- Detect all unauthorized file share activity that presents risk of data loss or misuse.
- Detect potential avenues of data loss not otherwise specified.
- Detect changes in behavior for the middleware applications that may present risk of data loss or misuse.

How do we implement these detection policies? The actual rules are shown below. Implementation is a multi step process:

1) First, we must collect flow data from the network where these transaction processing systems reside.

2) Second, we must create views which contain the flows which are eligible for our detection policy. Creating and managing views is detailed in the QRadar Administration Guide.
3) Third, we assign sentries to our views to produce events, in a unique category – Information Leak, in this example - when the relevant flows are detected. Creating and managing sentries is detailed in the QRadar Administration Guide.
4) Finally, we create the rules shown below to correlate the events into offenses so they are called to our attention in the Offense Manager.

Requirement	**Rule**

Detect and alert on unauthorized SQL sessions, real or attempted, to the main database; detect SQL traffic between the main database and any host other than the authorized application server.	`Apply Unauthorized NPI Database Access on events which are detected by the system`
	`and NOT where the Source IP is one of the following 10.10.10.1`
	`and where the Event Category for the event is one of the following Suspicious Activity.Information Leak`
	`and where the Destination Port is one of the following 1433`
	`and where the Destination IP is one of the following 10.10.10.10`

Detect and alert on unauthorized SQL activity involving critical database / application servers.	`Apply Unauthorized NPI Database Access on events which are detected by the system`
	`and where the Destination IP is one of the following 10.10.10.40, 10.10.10.100, 10.10.10.50`
	`and where the Event Category for the event is one of the following Suspicious Activity.Information Leak`
	`and where the Destination Port is one of the following 1433`

Detect and alert on remote administration traffic, in the form of terminal services sessions, so this can be easily monitored and corroborated with change controls and expected activity.	`Apply Terminal Service Connections to an NPI Database on events which are detected by the system`
	`and where the Destination Port is one of the following 3389`
	`and where the Event Category for the event is one of the following Suspicious Activity.Information Leak`

Detect and alert on FTP sessions sourcing from a critical database server which is not the result of virus definition updates or patch management. Detect SSH or SFTP sessions sourcing from a critical database.	`Apply FTP/SSH Activity from an NPI Database on events which are detected by the system`
	`and where the Event Category for the event is one of the following Suspicious Activity.Information Leak`
	`and where the Destination Port is one of the following 20, 21, 22`
	
	`Apply Custom-BB-Network Definition: McAfee, Microsoft, Akami Networks on events which are detected by the system`
	`and where the Destination IP is one of the following 216.143.70.0/24, 209.170.117.0/24, 209.170.116.0/24, 207.46.0.0/16, 65.52.0.0/14`

Detect and alert on attempts to read files from shares on critical database servers.	`Apply File Transfer Activity Involving an NPI Database on events which are detected by the system`
	`and where the Destination IP is one of the following 10.10.10.50, 10.10.10.50, 10.10.10.100`
	`and where the Destination Port is one of the following 445`
	`and where the Event Category for the event is one of the following Suspicious Activity.Information Leak`

Detect and alert on file share activity on	`Apply File Transfer Activity (SMB) to`

the main database when the source is not its application server.	CustomerDatabase on events which are detected by the system and NOT where the Source IP is one of the following 10.10.10.1 and where the Destination IP is one of the following 10.10.10.10 and where the Destination Port is one of the following 445 and where the Event Category for the event is one of the following Suspicious Activity.Information Leak
Detect other avenues of data loss not otherwise specified: detect and alert on any traffic sourcing from a critical database with a remote destination that is not the result of virus definition updates or patch management.	Apply Local to Remote Flows Involving an NPI Database on events which are detected by the system and where the Event Category for the event is one of the following Suspicious Activity.Information Leak and where the attack context is Local to Remote Apply Custom-BB-Network Definition: McAfee, Microsoft, Akami Networks on events which are detected by the system and where the Destination IP is one of the following 216.143.70.0/24, 209.170.117.0/24, 209.170.116.0/24, 207.46.0.0/16, 65.52.0.0/14
Detect changes on behavior on the middleware applications that may present risk of data loss or misuse.	Create a view for the middleware application server(s) and apply the following sentries: Behavioral sentry - bytes - both in and out Behavioral sentry - host count - both local and remote Anomaly sentry - bytes - both in and out Anomaly sentry - host count - both local and remote Behavioral sentry - bytes - both in and out Behavioral sentry - host count - both local and remote Anomaly sentry - bytes - both in and out Anomaly sentry - host count - both local and remote

False Positives

In this case study, our detection rules have very low rates of false positives. In the case of terminal server connections, an alarm on a terminal server session from a legitimate admin could be considered a false positive, strictly speaking, as the activity is legitimate. A SQL or SMB session from a database administrator could similarly be considered a false positive. These types of offenses could easily be tuned out at the discretion of the QRadar admins. In some cases, security teams will find it useful to choose to allow offenses to be created on these types of admin activity in order to demonstrate that all access, including privileged access, to PCI systems is being carefully monitored and observed continuously by security analysts. These offenses could also be used to detect unauthorized or unexpected access to PCI systems by privileged users – or to corroborate privileged access with change controls and established maintenance schedules.

Multivariate Detection and Correlation

Behavioral detection can be complemented and correlated with a number of other techniques. QRadar has a number of correlation rules for detecting suspicious database activity patterns including these:

- concurrent logins from multiple locations

- successful and unsuccessful logins from remote hosts
- successful and unsuccessful schema and configuration changes from remote hosts
- successful user changes preceded by schema or configuration change failures
- excessive schema or configuration change failures followed by success

Regular expression based detections of PCI data, for example, can be correlated with the behavioral detection events to identify inappropriate movement of PCI data. Snort, for example, has regular expression based signatures for credit card and social security numbers that can detect such data in ASCII protocol streams. Database and web server logs can be aggregated and mined to detect patterns of misuse. Some classic methods of misuse detection using database logs are;

- watching for selects of customer records outside the user's assigned territory
- detecting inappropriate selects e.g. customer service rep selecting financial data or non-rep selecting customer data
- detecting anomalous access e.g. users accessing tables or rows they rarely or never normally do
- selects of honeytokens - false records with high value created to attract data thieves
- detecting large number of selects
- detecting identifiable specific patterns of misuse like 'select * from account where accountbalance > 1000'

Conclusions

Behavioral detection using flow or log data can be a useful method of detecting activity that presents risk of data loss or misuse. Some behavioral detection rules will be inherently specific to their environment and require some behavioral profiling of the systems eligible for such detection methods. Behavioral detection need not be stand-alone and can be correlated with other detection methods. While no detection methods are perfect, most have value, and correlating disparate detection alerts can produce higher quality and accuracy of detection.