jSONAR Simplifying Security

SONARG

Optimizing Guardium DB Monitoring with SonarG

As the information security landscape continues to get more complicated and the emphasis on data level security grows ever more important, database activity monitoring (DAM) is thriving as a key control for the database layer. This is of course logical, since databases are the core repository for a large percentage of sensitive data within an enterprise organization. As such, they represent an important element of the data security and compliance strategies applied to ongoing efforts to protect against malicious behavior and InsiderThreat.

For many years the industry's leading DAM solution has been the IBM Guardium platform, deployed globally at some of the largest enterprise clients in the world and protecting hundreds of thousands of databases. SonarG was architected to further optimize the value and capabilities of the Guardium platform by taking advantage of modern Big Data technologies to streamline the collection and analysis of the large and growing pools of Guardium DB activity monitoring data.

While preserving the critical functionality and output of the Guardium system, SonarG has emerged as a valuable enhancement focused on optimizing three key areas:

- *Simplify Data Collection and Management*
- *Create broader access to valuable DB activity data*
- *Enable high performance analytics across expanded data sets*

To accomplish these goals, the SonarG system delivers the powerful combination of increased functionality to improve security and compliance controls, as well as substantial cost savings in both infrastructure HW and operational overhead. This paper provides a high level overview of the SonarG architecture, as well as further insight into the various benefits that it delivers.

DATA MANAGEMENT

DB monitoring systems typically generate a tremendous amount of data as they oversee data access and in particular privileged user activity. The data volumes can easily grow into many TBs, resulting in challenges around how to effectively collect and manage this data from both a logistical and cost perspective.

Historically this data has been used primarily for compliance reporting; however, many organizations are now expanding their focus to leverage this DB activity data for more comprehensive security analysis. As a result, the data footprint will accelerate in its growth, driven by more databases monitored, wider security collection policies and the desire to expand the data retention period from several months to a year or more.

Growing by an order of magnitude in the not too distant future is not an unreasonable projection. To optimize the data collection and management challenge, SonarG enhances the IBM Guardium deployment architecture in order to take advantage of next generation data warehousing technology that leverages advances in storage efficiency, scalability and query performance. Figure 1 below depicts the traditional Guardium architecture, including the optional second layer of aggregation for additional data consolidation and reporting.

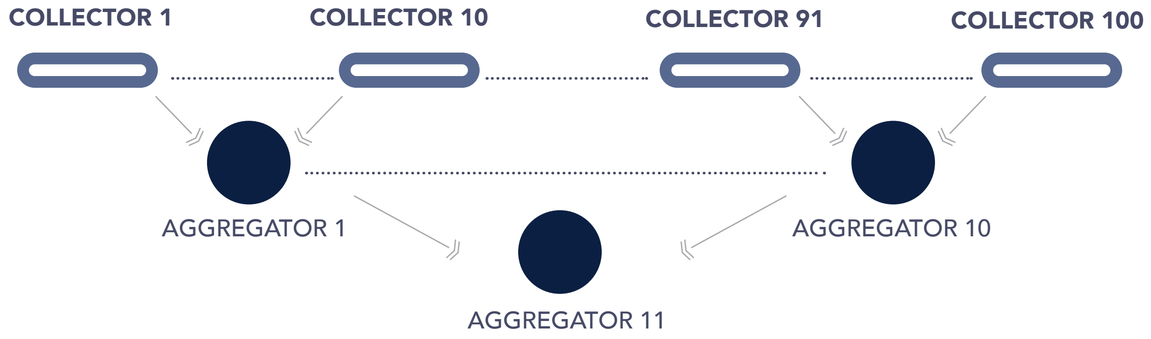


Figure 1 – Guardium Infrastructure Architecture

In the SonarG architecture the entire aggregation layer is eliminated and collectors communicate directly to the SonarG central warehouse, as shown in Figure 2. This greatly simplifies the data collection mechanics and facilitates much more efficient collection of larger datasets using less HW infrastructure.

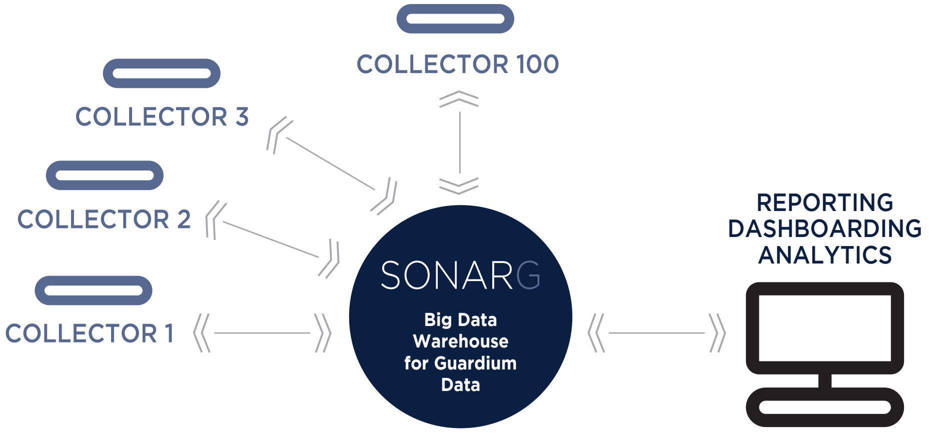


Figure 2 – SonarG Simplified Guardium Architecture

On an hourly basis each collector pushes the latest activity information to the SonarG warehouse, where it is merged with previous data to construct a single view of all activity across the entire collector landscape. The SonarG approach consolidates activity data across multiple CM domains, which is especially important for larger enterprise deployments that are typically deployed using multiple CMs. From an infrastructure perspective there are many advantages to this approach, including the following:

- *Eliminate HW and operational costs of aggregation*
- *Reduce latency of data consolidation from 24 hours to 1 hour*
- *Dramatically reduce collector storage footprint*
- *Improve collector throughput based on reduced workload*

The data warehouse embedded within the SonarG solution has several strengths that make it especially well suited for optimizing Guardium environments as well as other large scale collection and analytics challenges:

HIGH PERFORMANCE ANALYTICS ENGINE SonarG uses a columnar-compressed data warehouse specifically architected to perform analytic workload queries across large-scale data sets at very high performance. An example would be scanning 10 Billion SQL statements for a particular SQL command in seconds to minutes.

EASE OF USE AND SCALABILITY The architectural premise behind the SonarG data warehouse is merging the scalability of Hadoop with the flexibility/ease of use of NoSQL data structures and the high performance of MPP-class machines.

LOW COST HW A typical SonarG implementation relies on a single production server comprised of 2 XEON processors, 64-96GB RAM and anywhere from 10-50TB of low cost SATA storage. This commodity configuration is readily available for less than $15K from various industry leading server providers.

By incorporating the SonarG Big Data warehousing technology into their architecture, enterprise clients are able to reduce their Guardium HW footprint by more than 25% while they are simplifying the activity data collection flow and improving their ability to accommodate much larger datasets. The net result is a substantial savings on infrastructure costs, as well as having created a data warehouse ideally suited for enabling a variety of users and use cases to leverage valuable DB activity data.

DATA ACCESS

Once the Guardium activity data has been consolidated into a high performance Big Data warehouse using SonarG, enterprises can consider how best to apply this data to address a variety of use cases. As more and more data is collected and organizations expand their efforts to leverage this data, the need for simplified and relevant access by a variety of stakeholders also increases. It is no longer realistic to funnel report definition and execution solely through the Guardium administrators, as is typically done for audit/compliance reporting, since this quickly becomes an inhibitor to data access for broader usage. Organizations need a "Self-Service" model that enables data to be directly accessible by various stakeholders, including operations, DBAs, security, compliance, etc.

SonarG provides access to any data created by the Guardium system as it is all passed to SonarG during the hourly ingestion cycles. As with any modern data warehousing solution, SonarG was specifically designed to enable access to its data via a variety of methods. There are pre-built reports for the typical scenarios, a customer report builder, a dashboard builder and facilities for publishing this output to other applications. For example, a dashboard (Figure 3) is easily published as a web service and embedded within an operations console such that a wealth of systems data can be easily viewed completely independent of the Guardium system.

Beyond the pre-built reports that exist within the system, as well as the ability to easily create and publish custom reports or dashboards, SonarG also supports a number of alternative methods for accessing the data in the warehouse:

- *BI/Visualization Tools – Compatible with hundreds of tools includingTableau, Qlik, etc.*
- *REST API – complete API solution for automated interaction*
- *SonarSQL - A SQL-enabling layer that allows users to write queries in SQL to access data stored in SonarW*
- *SonarR - A gateway that allows users access queries and data from SonarW from within the "R" environment*

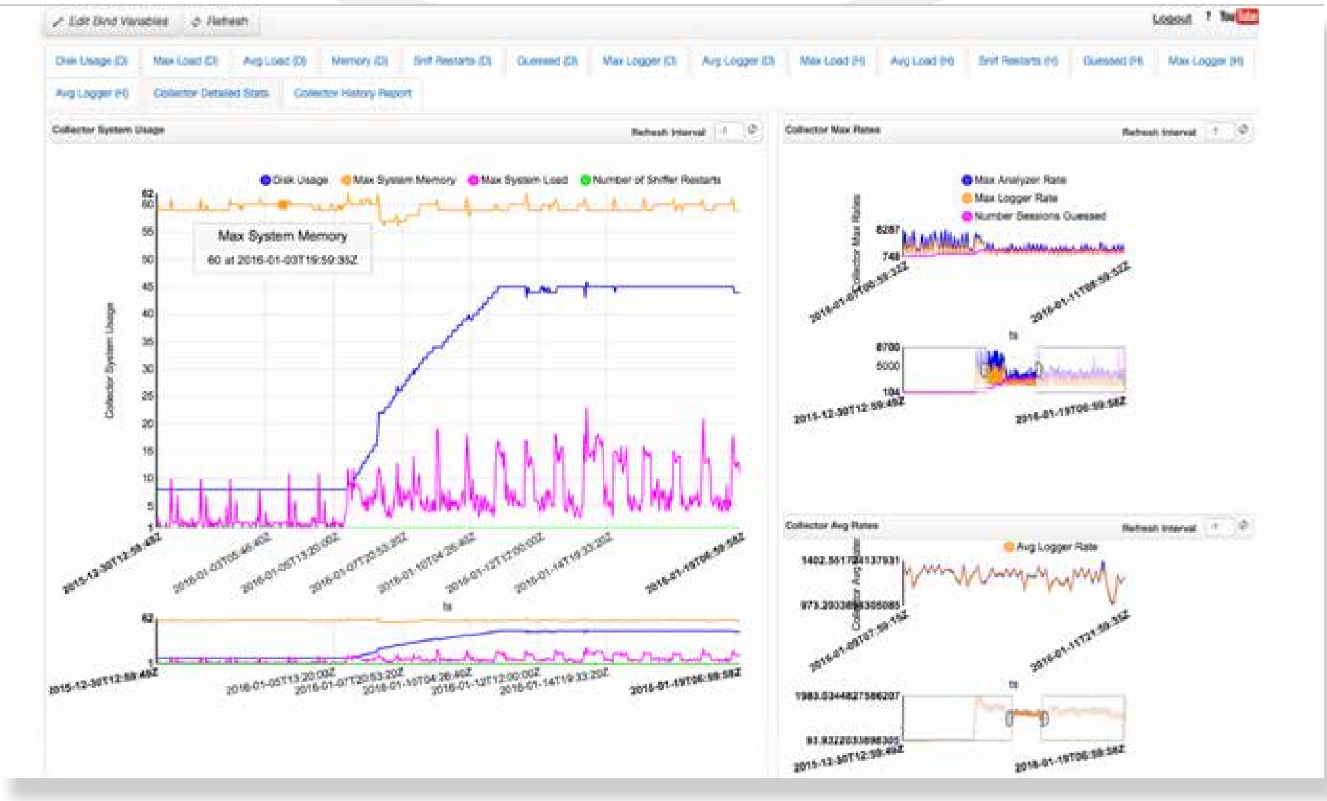
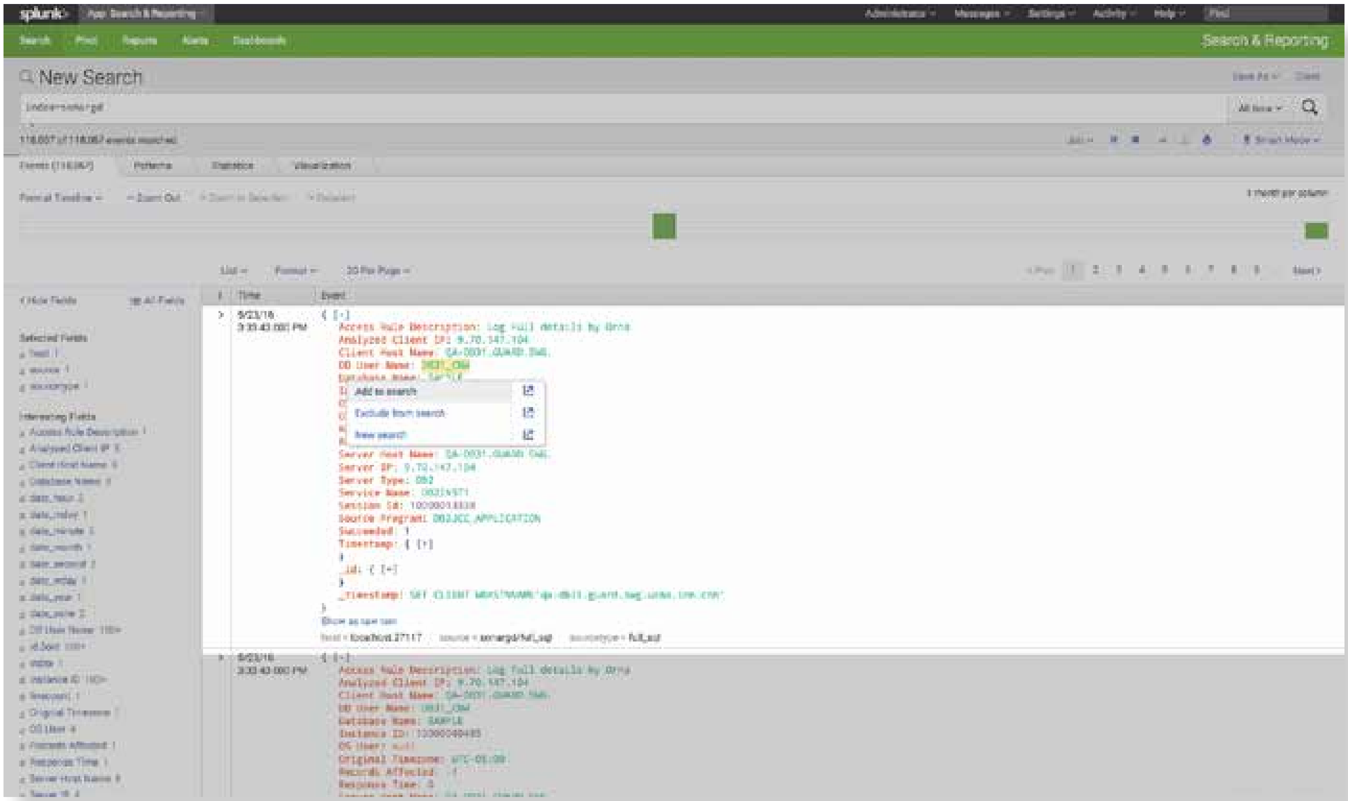


Figure 3 – SonarG Operational Dashboard

Another common scenario is security teams that rely upon Splunk for log aggregation/search and would like to incorporate DB activity information into this framework. While it is quite easy to pass data to Splunk via syslog and many customers use that today for extended retention, there are a number of critical pitfalls in this approach that reduce the value of the fine-grained Guardium data. There is a substantial loss of data quality and context, as the data exported via syslog does not contain key parameters including Objects & Verbs, Success/Fail indicators, Session IDs, normalized time and others. These losses are compounded by the inefficiency of the syslog protocol and resulting bloated data sets that are passed, often resulting in increased Splunk licenses costs since these are determined by indexed data volumes per day.

Rather than incur the double penalty of reduced data quality and increased costs, SonarG enables activity information to be easily accessed via the Splunk UI, as though the data had already been indexed into Splunk but in fact still resides in SonarG. Figure 4 provides an example of using the Splunk UI to access Guardium DB activity information residing within the SonarG system.



```
index=sonargd
```

118,087 of 118,087 events matched

```
9/23/16
3:33:43.000 PM
{ [-]
   Access Rule Description: Log full details by Orna
   Analyzed Client IP: 9.70.147.104
   Client Host Name: QA-DB31.GUARD.SWG.
   DB User Name: DB31_CNW
   Database Name: SAMPLE
   Server Host Name: QA-DB31.GUARD.SWG.
   Server IP: 9.70.147.104
   Server Type: DB2
   Service Name: DB2INST1
   Session Id: 10000031838
   Source Program: DB2JCC_APPLICATION
   Succeeded: 1
   Timestamp: { [+]
   }
   _id: { [+]
   }
   _timestamp: SET CLIENT WRKSTNNAME 'qa-db31.guard.swg.usma.ibm.com'
}
host = localhost:27117   source = sonargd/full_sql   sourcetype = full_sql

9/23/16
3:33:43.000 PM
{ [-]
   Access Rule Description: Log full details by Orna
   Analyzed Client IP: 9.70.147.104
   Client Host Name: QA-DB31.GUARD.SWG.
   DB User Name: DB31_CNW
   Database Name: SAMPLE
   Instance ID: 12000049485
   OS User: null
   Original Timezone: UTC-05:00
   Records Affected: -1
   Response Time: 0
```

Figure 4 – Splunk UI Access to SonarG Data

In addition to a plethora of methods for accessing the data contained within the SonarG warehouse, there is also tremendous flexibility in importing data from external sources that complement the Guardium activity data for use cases such as application mapping, DB ownership, ID reconciliation, etc. As a NoSQL database, SonarG can readily import virtually any data type using a variety of interfaces including CSV, JSON, files, etc and easily create data mash-ups that exploit the power of NoSQL schema-on-read and enrich the activity data for more effective usage.

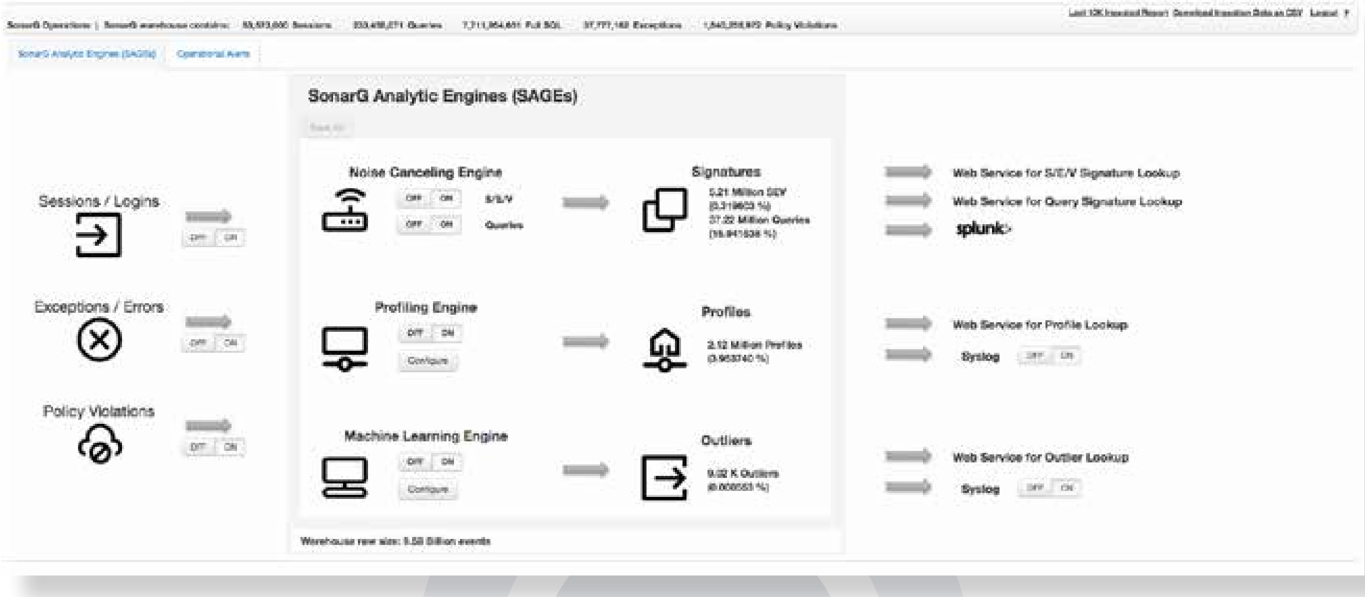
SECURITY ANALYTICS

DAM monitoring data provides excellent visibility into DB activity and thus more and more organizations are interested in applying rigorous analysis upon this data to better understand behavioral profiles and anomalies.

As discussed earlier, this is leading organizations to expand their Guardium collection scope and retention periods in order to capture richer information to use as the foundation for this analysis. SonarG is a powerful platform for enabling security analytics on DB activity data, based on its large-scale data store, embedded high performance query engine and its flexibility in supporting a wide range of access methods. All three of these architectural characteristics combine to make SonarG a powerful and highly effective analytics platform.

As an example of the raw power of the platform, consider a scenario where you would like to query a year's worth of Full SQL, which in this case is 27 Billion statements collected from 100 collectors over a year's period, to identify specific DDL activity from a specific privileged user. Simply being able to cost effectively house such a large data set (30TB raw activity data) is a key benefit of the SonarG system. But to then be able to execute a Regular Expression query that isolates the 19 events satisfying the criteria in just over 4 minutes demonstrates the performance benefits of SonarG's compressed-columnar query engine, especially when you consider that this was executed on a single node system with only 2 CPUs.

Building upon this foundation of capacity and analytical performance, the SonarG Analytics Engine (SAGE) facility provides a number of out-of-the-box analytics engines specifically optimized for addressing common database security and analytical concerns. As shown in Figure 5 below, SAGE includes a Noise Canceling Engine (NCE), a Profiling Engine (PE) and a Machine Learning Engine (MLE).



SonarG Operations | SonarG warehouse contains: 55,573,600 Sessions 233,458,271 Queries 7,711,864,651 Full SQL 37,777,162 Exceptions 1,640,258,872 Policy Violations

Signatures: 5.21 Million SEV (0.319603 %), 37.22 Million Queries (15.941538 %)

Profiles: 2.12 Million Profiles (0.953740 %)

Outliers: 9.02 K Outliers (0.000563 %)

Warehouse raw size: 5.58 Billion events

Last 10K Inserted Report
Download Insertion Data as CSV
Logout
SonarG Analytic Engines (SAGEs)
Operational Alerts
SonarG Analytic Engines (SAGEs)
Sessions / Logins
Exceptions / Errors
Policy Violations
OFF ON
Noise Canceling Engine
S/E/V
Queries
Profiling Engine
Configure
Machine Learning Engine
Signatures
Profiles
Outliers
Web Service for S/E/V Signature Lookup
Web Service for Query Signature Lookup
splunk>
Web Service for Profile Lookup
Syslog
Web Service for Outlier Lookup

Figure 5 — SAGE: SonarG Analytic Engines

Data flows from key activity domains (Sessions/Logins, Exceptions/Errors or Policy Violations), into the various engines for rigorous analytical inspection and the resulting output is available via various flexible interfaces including web services, Splunk or syslog messaging. These engines are easily invoked and require minimal configuration effort, yet deliver powerful insight. Each engine can be enabled separately but if all three are enabled they will share resources and computation. The goal of SAGE is to minimize the level of effort required to develop complex database security programs and enable organizations to rapidly benefit from their high value output.

Each of the three SAGE engines addresses key scenarios that are fundamental to any database security program, as described below:

1. *NOISE CANCELING ENGINE Based on the high volumes of activity information being collected by DAM systems, many implementations are struggling with extremely high alert volumes and the diluted value of these alerts simply based on excessive volumes. How does one effectively deal with 22M alerts a day on sensitive object access passed to an inspection workflow?*

   The Noise Canceling Engine (NCE) operates as an analytic reduction machine that evaluates very large volumes of raw data and reduces these into a much smaller set of signatures with counters using various analytical dimensions. Built-in reducers are included for connection data, query data, exceptions and policy violations. Additional reducers may be added and existing reducers may be changed in terms of which dimensions are used and their associated configuration parameters.

2 *PROFILING ENGINE Inspecting and enforcing database connections is a key element of any database security strategy since this represents a primary control point for which tools, users, applications, etc., should be interacting with which databases. Trusted Connection (TC) monitoring relies on this “whitelist” definition to evaluate access. Unfortunately, many organizations struggle to implement TC enforcement based on the difficulty of gathering the necessary information from application owners, DBAs, operations teams and others that is required to construct an effective whitelist.*

   The Profiling Engine (PE) focuses on database connections and is primarily used for implementing Trusted Connection analysis. The PE is used to automatically catalog all connections and construct the definition of the whitelist via analytical inspection. Alternatively, data from an external data-source such as a CMDB can be uploaded into the system and used as input for defining trusted connections. Each time a new connection is observed it is both flagged as a “new connection” (to be inspected and verified) and added to the connection profile as a “learned connection” to avoid being repeatedly flagged and to reduce the amount of manual work involved in managing the connection profiles.

3. *MACHINE LEARNING ENGINE While the PE facility provides an excellent mechanism for understanding legitimate versus unwanted connections, perhaps the most critical area of focus for database security is behavioral analysis for users that tracks and persistently evaluates large volumes of historical activity data in order to rapidly isolate unusual activities such as unauthorized malicious code or privileged users attempting to stockpile sensitive data. This level of fine-grained inspection is challenging at the database layer because of the large volumes of data and numerous behavioral variables.*

The SonarG Machine Learning Engine (MLE) learns user behavior and persistently evaluates this behavior to identify anomalies or outliers. Factors considered include user connections, user exceptions, violations attributed to users or any combination of these parameters. The MLE compares user behavior based on the following dimensions:

- *Behavior of different users on the same database instance to determine if any user’s actions represent a statistical anomaly*
- *Behavior within a user category, for example, comparing each DBA to every other DBA’s behavior*
- *User behavior over time to isolate unusual behavioral patterns as compared to that user’s history*

MLE is a powerful facility for evaluating user behavior and coupled with the large scale data retention capabilities of SonarG provides the mechanism for automatically defining, overseeing and enforcing the assessment of every database user’s actions. What makes MLE especially powerful is that it is delivered as a complete and automated engine that requires minimal configuration in advance of receiving tremendous insight into user behavior. Of course as with all of the SAGE engines, a user has the flexibility to adjust any of the pre-defined configuration parameters and/or add additional dimensions to refine their inspection.

SUMMARY

The use of Database Activity Monitoring continues to expand as enterprises increase their emphasis on data level security and look to leverage the output of DAM systems well beyond traditional compliance reporting. SonarG provides the optimal platform for exploiting valuable DB activity data generated by the industry’s leading DB activity monitoring solution, IBM Guardium. By exploiting next generation Big Data technologies, SonarG is able to deliver key benefits that fully complement the Guardium system, expand functionality in the critical security domain and realize significant infrastructure and operating cost savings.