Secure and protect Cassandra databases with IBM Security Guardium

Improve security and compliance for open source databases

Kathryn Zeidenstein
Sundari Voruganti

April 19, 2016

IBM Security Guardium® leads the way in providing a monitoring and auditing solution for NoSQL database systems. In this article, we provide an overview of one popular NoSQL database, Apache Cassandra, and explain how and why Guardium can help organizations protect Cassandra data and automate compliance reporting and sign-offs. This article includes detailed instructions and a sample security policy to help you configure Guardium and extract value immediately.

About this tutorial
To view this video, Protect your Cassandra databases, please access the online version of the article. If this article is in the developerWorks archives, the video is no longer accessible.

Introduction

The proliferation and increased use of NoSQL and big data systems aligns perfectly with the increased monetization of stolen data and the resulting increase in data breaches, creating a perfect storm of cybercriminal opportunity. Although breaches on NoSQL systems are relatively rare when compared to relational systems, we expect these incidents to rise as more valuable data is stored in these new systems.

For readers who are already familiar with Guardium, having an overview of the architecture and where to install S-TAPs® will likely be enough to get you started. If you are new to Guardium, this article can help you get started, but it cannot supplant the need for at least some basic understanding of Guardium administration and configuration.

Overview of Cassandra and DataStax

Apache Cassandra is in the class of NoSQL database systems; specifically, it is a column-based store. Its architecture is different from other NoSQL systems because it does not rely
on either a master-slave (like HBase) or sharded architecture (such as MongoDB). It instead has a “ring” architecture, in which each node in the system communicates with all other nodes. The architecture is designed to facilitate performance, scaleout, and continuous availability. An understanding of the basic architecture is helpful for understanding how to deploy Guardium monitoring agents.

DataStax is the company that delivers support and additional enterprise features for Cassandra, such as enhanced security and search.

The Related topics section of this article includes some helpful references on Cassandra basics and the differences between Apache Cassandra and the DataStax distribution.

Although the underlying storage model is column based, Cassandra has its own query language, Cassandra Query Language (CQL), that is used to access data. You can use the query language commands through one of three options: shell (cqlsh), a graphical tool called DevCenter, or a variety of program drivers.

Note: A Thrift API is still supported as well for legacy applications. Guardium supports the monitoring of access through the legacy API as well.

Table 1 is a mapping between Cassandra constructs and relational constructs. Example commands are courtesy of the Cassandra Query Language Reference.

Table 1. Mapping between relational database and Cassandra concepts

<table>
<thead>
<tr>
<th>Relational Concept</th>
<th>Cassandra</th>
<th>CQL Commands</th>
<th>CQL Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>Database</td>
<td>Keyspace</td>
<td>CREATE KEYSPACE</td>
<td>CREATE KEYSPACE Excelsior WITH REPLICATION = { 'class': 'SimpleStrategy',</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ALTER KEYSPACE</td>
<td>'replication_factor': 3 };</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Table and materialized</td>
<td>Column Family</td>
<td>CREATE TABLE</td>
<td>CREATE TABLE emp ( empID int, deptID int, first_name varchar, last_name</td>
</tr>
<tr>
<td>view</td>
<td></td>
<td>CREATE KEYSPACE</td>
<td>varchar, PRIMARY KEY (empID, deptID));</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>CREATE MATERIALIZED VIEW cyclist_by_age AS SELECT age, birthday, name,</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>country FROM cyclist_mv WHERE age IS NOT NULL AND cid IS NOT NULL PRIMARY</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>KEY (age, cid);</td>
</tr>
<tr>
<td>Index</td>
<td>Index</td>
<td>CREATE INDEX</td>
<td>CREATE INDEX user_state ON myschema.users (state);</td>
</tr>
<tr>
<td>Column</td>
<td>Column (which can also be a</td>
<td>INSERT INTO emp (empID, deptID, first_name</td>
<td>INSERT INTO emp (empID, deptID, first_name, last_name) VALUES (104, 15,</td>
</tr>
<tr>
<td></td>
<td>collections of items)</td>
<td></td>
<td>‘jane’, ‘smith’);</td>
</tr>
</tbody>
</table>
Cassandra evolves its authentication and authorization model as it matures. Because the NoSQL space is young and still evolving, security mechanisms are evolving rapidly to keep up. For this reason, it is important that you research the latest documentation for both Cassandra and DataStax Enterprise to get the most current information.

**Roles and permissions**

You can set up users internally in Cassandra (e.g., CREATE USER, ALTER USER, and so on.) and as of 2.2, you can even use role-based access controls such as CREATE ROLE, ALTER ROLE, and so on.

Permission management to objects (data) is handled by using familiar GRANT and REVOKE statements. By default, authentication is disabled in Cassandra. You can set up password authentication. DataStax Enterprise supports authentication via Kerberos and LDAP.

**Encryption:** DataStax Enterprise includes support for transparent data encryption of data as it is written to disk, but it requires a secure local file system. In addition, if you require the commit logs to be encrypted, you will need a file system encryption capability such as that delivered by IBM Security Guardium Data Encryption.

**Auditing:** This is an area where Guardium provides significant added value. Although Cassandra has a general logging facility (which can be used for troubleshooting and point-in-time recovery) it would require a significant amount of work for an organization to create timely audit reports. And as stated in the product documentation, "increasing logging levels can generate heavy logging output on a moderately trafficked cluster."

DataStax Enterprise provides an audit logging facility, which must be enabled on all nodes to capture all desired events. Log events can be written to the file system (which can quickly grow quite large and be difficult to analyze) or to Cassandra tables.

As we will discuss in the next section, Guardium can improve audit reporting and dissemination and also the ability to conduct real-time alerting and near real-time analytics.

**How Guardium facilitates compliance and protects Cassandra databases**

If you are unfamiliar with Guardium, this section offers a quick review of how Guardium works, and the value it brings to complement or replace native database security auditing controls. For the purpose of this article, we'll focus on Guardium Activity Monitoring. Capabilities such as database discovery, data classification, and vulnerability assessment are not currently available in Guardium for Cassandra at this time.
How Guardium activity monitoring works

As shown in Figure 1, Guardium continuously monitors data activity by using lightweight software probes, called S-TAPs, without relying on logs. The S-TAPs also do not require any configuration changes to the Cassandra servers or applications. The other major component in the Guardium architecture is the collector, which is tamper-resistant hardware, software, or virtual appliance. This architecture is optimized for speed and facilitates separation of duties.

Figure 1. Guardium architecture

To keep the processor usage on the database servers very low, the S-TAPS immediately intercept messages at the OS level and forward data activity to the collector. Depending on how much activity you decide to audit, the processor usage on the cluster should be no more than 2-4 percent.

The analysis engine on the collector (known as the sniffer) parses the data and will fire alerts for policy violations as defined by the policy rules. Real-time alerts provide near instant visibility into suspicious or anomalous events. These alerts are visible on the Guardium user interface and can be sent via email and/or integrated immediately with your security intelligence and event management system, such as IBM QRadar®, Splunk, or other similar systems.

Audited data events are stored in the collector's tamper-resistant repository, which is inaccessible other than by the Guardium interfaces and cannot be deleted.

Scalable architecture

To support a variety of deployment sizes, Guardium scales horizontally and vertically by using a federated architecture that can be managed by a single central management console. Capabilities such as enterprise load balancing, updates and configurations, and overall system health can be managed from the Central Manager.

Value of Guardium for Cassandra

As we discussed, Guardium integrates with the OS kernel to provide in-depth, granular visibility into data access. Depending on who, what, where, when, and how that access is occurring,
security policies can perform a number of actions; some examples of these actions are logging the event into its protected database, sending a real-time alert to a security information and event management (SIEM) service or software, blocking access, and even redacting result set data. In addition, this same platform and security policy can be used with any of your other enterprise databases, including relational databases and other NoSQL systems such as MongoDB.

These capabilities provide significant added value of reliance on native auditing, including:

- **Separation of duties:** For security and compliance, it's an accepted best practice not to enable privileged users, such as DBAs, to also be in control of security; in addition, it's critical to monitor everything that they do on any system that includes sensitive data. The information security team can administer Guardium regardless of whether audit logging is enabled on the database server or not.

- **Speed of detection and integration with security operations:** Real-time alerts can be specified for a wide variety of conditions, and you will learn how to specify those here. When breaches occur, being able to detect and react quickly can mean the difference between a hugely damaging loss and a minor inconvenience. You can send alerts by using email or through SNMP to another monitor system. Built-in integration with IBM QRadar and HP ArcSight message types can also be used to automatically forward alert conditions from syslog to those systems. You can customize alert messages to be sent to other systems as well, such as Splunk.

- **Speed of compliance reporting:** Demonstrating compliance can be time consuming and burdensome as these often require some level of regular review and sign-off. Guardium not only lets you create the reports that you need to satisfy audit requirements, it also has a robust workflow capability that integrates into your business processes and saves all sign-offs and reviews as part of the audit trail. You'll read more about this later in the article. Guardium includes hundreds of prebuilt reports, including reports that are designed to jump-start compliance reporting for SOX, HIPAA, PCI-DSS, and Data Privacy. See Related topics for a link to a developerWorks article on using accelerators.

**Figure 2. Compliance accelerators**
faceted browse and text search, together or in combination to quickly do ad hoc investigations of audit data. The following example in Figure 3 shows how easy it is to find out who recently issued a DROP TABLE command by browsing to the DROP TABLE in the verb facet and clicking it to filter the search results to all DROP TABLE events.

**Figure 3. Using Quick Search to investigate activities**

In addition to search, there are other analytics and visualization techniques to help with detection and analysis including:

- Threshold alerts, which can detect when events occur over time, such as an excessive number of downloaded records by a particular user over time.
- Outlier detection, which does sophisticated cluster analysis of user activity to help you find abnormal activity that might otherwise go undetected.
- Investigation and threat dashboards to help analysts understand activity and detect abnormal activity more quickly.
- **Support of secure archiving:** Audit information must be stored for a defined period of time, sometimes years. Guardium is designed with these kinds of requirements in mind and provides secure archiving capabilities for this very reason.
- **Data protection:** Guardium can block user access by cutting the network connection to the database. You can also use dynamic data redaction to obfuscate private data in results sets when they are obtained via a "backdoor" access by using privileged IDs.

**Planning your deployment**

This article focuses on the basics of installing and configuring a single collector system for capturing Cassandra activity. For enterprise deployments, you will need to consider how many
collectors you will need to handle the expected load and you will need to use either distributed reporting or aggregator appliances to conduct reporting on the collected traffic that might be dispersed across multiple collectors.

The Related topics section of this article includes a link to a redbook on Guardium deployment. If you are new to Guardium, we highly recommend that you use experienced Guardium services or Business Partners to help you with initial deployment and to help you develop your initial security policy.

**Installation and configuration**

Before you begin, gather the following information:

- Make sure that the necessary ports are open to ensure proper flow of traffic between the Guardium collector and the Cassandra nodes, Guardium installation manager client and server, and so on. The Related topics section includes a link to a technote that lists ports used by Guardium.
- From IBM FixCentral, download the latest S-TAP for your OS and check the kernel compatibility matrix to ensure that the K-TAP (kernel tap) matches your OS kernel level, or that it works by using flex loading. The kernel compatibility matrix is on FixCentral. Look for the K-TAP Bundle section of Guardium download and click on the KTAP_List_of_Modules_<version> link as shown in Figure 4.

**Figure 4. Accessing the K-TAP - kernel compatibility matrix**

For more information on flex loading, see the technote in Related topics.

- Record the ports used by Cassandra for both Thrift and CQL. The default is 9160 for Thrift and 9042 for CQL, but it’s conceivable that the DBAs changed the defaults for improved security.
- Record the IP addresses or host names of the Cassandra nodes.
- Record the IP address or host name of the Guardium collector that will be logging real-time traffic.
- Make sure that the Guardium collector is installed with the default collection profile policy (see the following screen capture in Figure 5) before traffic starts flowing to the collector. This policy records each instance of a unique connection, consisting of a five-part tuple consisting of the client IP, source application, database user, server IP, and service name—this is to avoid flooding the collector with traffic.

**Note:** Cassandra does not use service names, so this value is hardcoded in Guardium.
Install the S-TAPs

Because data can be accessed by using any node in Cassandra, you must install S-TAPs on all nodes. Using the Guardium Installation Manager is the recommended way to install S-TAPs, but you can use the shell installer as well.

Configure the inspection engines

When your S-TAP successfully installs and displays as green in the S-TAP control panel, you can then tell Guardium specifically what kind of traffic to listen for by configuring Guardium inspection engines with the correct Cassandra ports for your environment.

In our case, we want Guardium to listen for both Thrift messages (default port 9160) and Cassandra query language traffic (default port 9042). You can do this by configuring inspection engines from the appliance, either by using the UI or the Guardium API commands from the command line interface.

To go through the UI, navigate to Manage>Activity Monitoring>S-TAP Control, and click the blue pencil icon for the appropriate S-TAP host. Expand Add Inspection Engine. Figure 6 shows what it looks like for one S-TAP host.
Figure 6. Configured inspection engines for Cassandra

An alternative is to log in to the Guardium appliance with one of the CLI user IDs and run the Guardium API `create_stap_inspection_engine`, as shown in Listing 1.

Listing 1. Use Guardium APIs to create inspection engines

```bash
grdapi create_stap_inspection_engine client=0.0.0.0/0.0.0.0 protocol=Cassandra ktapDbPort=9160 portMax=9160 portMin=9160 stapHost <ip of Cassandra server where associated STAP is installed>

grdapi create_stap_inspection_engine client=0.0.0.0/0.0.0.0 protocol=Cassandra ktapDbPort=9042 portMax=9042 portMin=9042 stapHost <ip of Cassandra server where associated STAP is installed>
```

Sample security policy

The following screen capture in Figure 7 shows the security policy summary that we'll use to illustrate how to achieve certain goals, such as masking sensitive data on returned data.
Figure 7. Sample policy used for this article

![Policy Rules](image)

**Important:** Do not use this policy "as is." It is best to start with basic monitoring before you add advanced capabilities such as blocking and redaction. Blocking, in particular, can impact performance and the implementation must be done with care.

**Policy rule configuration**

Policy rules are order-dependent and require some background knowledge. The Resources section includes a link to a page on the Guardium developerWorks community with links to several YouTube recordings that provide more details on policy rules.

As mentioned previously, we recommend that you go through the exercise to go through connection profiling and filter out trusted connections as needed. We also recommend that you go through the exercise of determining which tables are "sensitive" and put those tables into appropriate groups by using the Guardium group builder. In addition, you will benefit from also having groups of users (such as administrators, DBAs, or application users) so that you can create your policy rules based on who is accessing the data.

Policies consist of different kinds of rules:

- **Access:** Logs activity from the database client to the database server.
- **Exception:** Logs exceptions returned from the database such as error codes or failed logins.
- **Extrusion:** Acts on the result set, such as redacting values.

Use filtering criteria in the rule to determine when that rule should fire, such as access by particular users to particular objects. Whitelisting and blacklisting are supported.

The policy that we illustrate will cover the following use cases:

- **Filtering out "noise."**
- **Alerting on failed login attempts that exceed a threshold you set.**
• Logging in detail all access by privileged users.
• Sending an alert when a privileged user inappropriately accesses sensitive data.
• Blocking access to a highly sensitive table.
• Redacting private data before a privileged user can see it.

Skip logging of activities on certain objects

This rule action is used to avoid logging items of no interest when it is not possible to filter out those items on the S-TAP. (Use *Ignore the S-TAP session* to filter out trusted sessions on the S-TAP, which can improve performance on the collector.)

In our case, we wanted to make it easier to focus our reports on the relevant results. Thus, we filtered out access to system tables because Cassandra does a lot of system table checks for each query. Because these queries are attached to the query activity, we cannot filter these items out by using the *Ignore the S-TAP Session* action. We used the group builder to create a member, called CassandraSystemObjects, and added the member `system%` to it. You can add more members or instead add specific system objects that you are comfortable filtering out.

**Note:** This example is for illustration purposes only. It should *by no means* be taken as a recommendation to filter on system object access. As an alternative, you could go ahead and log system access and filter out certain objects at the report level.

**Figure 8. Skip the unnecessary logging of some monitored activities**

![Image showing group builder for Cassandra system objects](image)

**Detect possible brute force attack - multiple failed logins**

The failed login rule lets you specify an action that occurs whenever there are multiple failed logins within a user-specified period of time. The rule is very simple: Select the exception type of `LOGIN_FAILED` and the number of tries before you reset the timer. The rule below specifies that there were three failed logins within 5 minutes.
Figure 9. Track excessive failed logins attempts

The following screen capture in Figure 10 illustrates what the accumulated failure would look like in the Policy Violations / Incident Management report in Guardium:

Figure 10. Policy violation - more than three failed logins in 5 minutes

Detailed monitoring of all privileged user activity

Regulations such as PCI require that all access by privileged users (such as DBAs or system administrators) be audited in detail. You can use the white list approach as we demonstrate here, which compares the database user with your group of privileged users. Or you can use a black list, which would turn on detailed monitoring for all users that are not application users, for example.

Figure 11. Detailed monitoring of privileged users
Send alerts for inappropriate access by privileged users

Our next rule is a natural follow-on. In this example, we continue monitoring the same set of privileged users, but we add on more conditions: The objects that are accessed (in the Cardholder group) and what commands are used. We specify it to send an alert to our SIEM system if this rule is triggered (Figure 12). We are also logging a policy violation to Guardium so we can show you what that looks like in the Policy Violations/Incident Management report (Figure 13).

Figure 12. Send alert when privileged users access cardholder objects by using DML or SELECT

![Figure 12. Send alert when privileged users access cardholder objects by using DML or SELECT](image)

Here is the Policy Violation/Incident Management report (Comply > Reports > Incident Management) that shows the high severity violation.

Figure 13. High severity policy violation

![Figure 13. High severity policy violation](image)

Block access by unknown connection to highly sensitive object

If you have the advanced edition of Guardium, you can consider the use of blocking for specific situations. With blocking (known as S-GATE in Guardium), certain sessions are specially monitored. This is known as "attaching." Actions by users in that session are then checked against the policy rule to see whether they are attempting to access certain tables that you specify in that rule. If so, the rule is triggered with the action S-GATE Terminate, and their connections are cut before their request can go through to the database.

Because of the need to check each command in the attached session against the policy rule, you must be careful about which sessions are attached or you can accidentally impact performance for applications. Use blocking only for protection against backdoor access or to prevent users with elevated credentials from accessing sensitive data.
We have two rules to support blocking: The first rule attaches to any privileged user session. The second one blocks any access by privileged users against customer tables.

**Figure 14. One rule to attach to a session and one rule to block on access to a particular object or object group**

Attach on the session....

Block on the object...

---

**Redaction of results sets**

Finally, let's look at one last rule, an *extrusion* rule, where we want to prevent privileged users from seeing private data. Extrusion rules require that you change the "master" configuration of inspection engines to enable inspection of returned data.

Navigate to **Manage > Activity Monitoring > Inspection Engines** and click **Inspect Returned Data**, as shown below. Then click **Restart Inspection Engines**.

**Figure 15. Configure the inspection engine to look at returned data**

Now you're ready to add the new extrusion rule to your policy. Edit your policy and add another extrusion rule. Add the conditions that will trigger the redaction of result sets, such as particular
users and client IPs. (Note that extrusion rules cannot be specified by object. Guardium is looking for the pattern in all results for that specified connection.)

Next, you must specify the regular expression that Guardium will use to examine the result sets for matching patterns. Guardium includes several prebuilt regular expressions, but you can create your own or modify the existing ones to match what it is you are trying to redact.

To access the regex builder, click the RE icon next to the Data pattern field in the rule.

**Figure 16. Regular expression icon brings up the regular expression builder**

![Figure 16](image)

Select one of the built-in identifiers or create and test your own using the built-in tester, as exemplified in Figure 17.

**Figure 17. Choose an existing pattern or build your own**

![Figure 17](image)

Save and reinstall your policy.

Figure 18 is an example of what redaction would look like if the policy rule is applied for U.S. Social Security numbers.
Figure 18. Social Security numbers are masked in results sets for privileged users

<table>
<thead>
<tr>
<th>empid</th>
<th>first_name</th>
<th>last_name</th>
<th>ssnid</th>
</tr>
</thead>
<tbody>
<tr>
<td>105</td>
<td>joe</td>
<td>smith</td>
<td><em><strong>-</strong></em>-6666</td>
</tr>
<tr>
<td>107</td>
<td>joe</td>
<td>jones</td>
<td><em><strong>-</strong></em>-8888</td>
</tr>
<tr>
<td>108</td>
<td>jane</td>
<td>jones</td>
<td><em><strong>-</strong></em>-9999</td>
</tr>
<tr>
<td>104</td>
<td>jane</td>
<td>smith</td>
<td><em><strong>-</strong></em>-2345</td>
</tr>
<tr>
<td>106</td>
<td>jane</td>
<td>doe</td>
<td><em><strong>-</strong></em>-7777</td>
</tr>
</tbody>
</table>

Compliance reporting automation

A frequently used Guardium capability is the ability to create audit processes. You can assign one or more tasks (such as a particular report) to run on a schedule and send the results to the appropriate set of reviewers in the correct workflow order.

In this case, we decide that we want to create a daily report on all database GRANT statements. We need our PCI reviewer to be made aware of it and then we need our Audit supervisor to sign off on it.

What we did to support this use case is clone one of the built-in reports that matched closely to what we wanted. In our case, we chose the report entitled Execution of Grant Commands.

Tip: To find likely candidate reports, navigate to My Dashboards, Create a new Dashboard. Click Add a new report. In the filter bar, type "grant" and you'll see several possible candidates.

To modify the report, add it to your dashboard and then click the blue pencil icon in the upper left section of the report, which brings up the query for that report. In the query builder, click the Clone button, at which point you can give the report query a new name in the upper part of the query builder. You can add or remove columns (fields) from the report and modify search conditions. In our case, we remove source program name and service name and add the Full SQL field.

Our report now looks like the following screen capture in Figure 19:

Figure 19. Our modified Daily Privileges report

Now we need to add this report to an audit process, schedule it, and send it to appropriate reviewers.
To create a new audit process, navigate to **Comply > Tools and Views > Audit Process Builder**. The builder walks you through the process. The figure below shows our audit process, Daily Reports, with the Privileges Daily Report as a task that runs once a day. The reviewers are users with pci and audit roles.

**Figure 20. Audit process builder helps us create a compliance workflow**

You can send report results to users by email, but in our case our audit users will be logging directly in to Guardium to see their activities. (Note that it is very easy to customize the UI and permissions for different roles. See [Related topics](#) for a link to a video on how to do this.)

Assuming that our PCI user reviews the report, it then gets added to the to-do list of our Audit supervisor, AuditMan, who will see an alert in his to-do list icon in the dashboard as shown in Figure 21.

**Figure 21. To-do list indicator in the banner**

AuditMan will see the report results and then can make comments, escalate, or sign off. These comments and statuses will be stored with the audit process results, which can be archived and retrieved as needed should your organization need to retrieve that for internal or external auditors.
Summary of capabilities

We hope you will find this article helpful in configuring and using IBM Security Guardium to help protect sensitive data in Cassandra from hackers and insider threats and helping you meet your compliance requirements.

The following table highlights the features and capabilities that are available in the Guardium platform for Cassandra.

Table 2. Guardium platform support for Cassandra

<table>
<thead>
<tr>
<th>Capability</th>
<th>Guardium offering</th>
</tr>
</thead>
<tbody>
<tr>
<td>Granular monitoring and auditing</td>
<td>Data Activity Monitor</td>
</tr>
<tr>
<td>Real-time and threshold alerting</td>
<td>Data Activity Monitor</td>
</tr>
<tr>
<td>Advanced analytics (outliers, threat detection)</td>
<td>Data Activity Monitor</td>
</tr>
<tr>
<td>Search and browse of audit data</td>
<td>Data Activity Monitor</td>
</tr>
<tr>
<td>Robust REST-based API</td>
<td>Data Activity Monitor</td>
</tr>
<tr>
<td>Dynamic data redaction</td>
<td>Data Activity Monitor</td>
</tr>
<tr>
<td>Blocking</td>
<td>Data Activity Monitor (Advanced)</td>
</tr>
<tr>
<td>File level data encryption</td>
<td>Data Encryption</td>
</tr>
<tr>
<td>Monitoring and alerting of activity on files,</td>
<td>Activity Monitor for Files</td>
</tr>
<tr>
<td>such as database configuration files or data</td>
<td></td>
</tr>
<tr>
<td>files</td>
<td></td>
</tr>
<tr>
<td>Blocking access to files</td>
<td>Activity Monitor for Files (Advanced)</td>
</tr>
</tbody>
</table>

Appendix: Connection profiling

Connection profiling is a handy way to get an overall view of database accesses to help you determine whether there are connections that can be considered trusted and filtered out on the S-TAP or if there are others that are suspicious. The connection profiling list report shows each unique connection and provides a count of the number of times that connection is made during the reporting period specified. The report below shows that the Cassandra connection we used for this article was used 36 times.
Figure 23. Connection Profiling List report

From this report, you can invoke the Guardium API to add connections to groups, such as suspicious connections or trusted connections that can be used in your security policy rules moving forward.

To invoke the API from the user interface:

1. Right-click the connection row in the report.
2. Scroll down and select **Invoke...**, then **create_member_to_group_by_desc**.
3. Change the description to one of your existing connection groups. Then click **Invoke Now**.
Figure 24. Adding connections to groups directly from the Connection Profiling List report

If you don't want to see this connection in your connection profile report any more, choose the default group, Connection Profile List.
Related topics

- Visit Planet Cassandra for an overview and download link for Apache Cassandra.
- To read more about DataStax enhancements and Apache Cassandra capabilities, visit DataStax's documentation on product compatibility.
- Read DataStax's blog on a discussion of role-based access controls in Cassandra.
- For more about encryption and other security topics, visit DataStax's security documentation.
- Read up on DataStax Enterprise's security documentation, which covers a wide range of topics including security management, authenticating with LDAP, managing object permissions, and more.
- To gain an understanding of Cassandra query language commands, read the Cassandra Query Language guide and reference.
- Read "Accelerate the path to PCI Compliance using Guardium" on developerWorks for an overview of the PCI accelerator. Although this article is based on V9, much remains the same, other than the fact that you no longer have to install a separate patch to get the accelerators.
- This tech note provides more information about flex loading of the K-TAP and how to enable it.
- To get a list of ports used by Guardium, read this tech note.
- Learn more about Guardium Installation Manager in the product documentation.
- The Deployment Guide for Guardium is a must-read for anyone who wants to learn more about Guardium. Although this redbook is based on V9, many of the concepts are the same.
- To really understand policies, it is best to understand the basic concepts of how monitored events flow from the database and are processed in Guardium. This video series provides a good foundation for learning about policies.
- Watch this YouTube video on how easy it is to customize the Guardium user interface in V10. For demonstrations and tech talks, visit IBM Guardium's YouTube channel.
- Follow IBM Guardium on Twitter.
- Read the e-book, "Top tips for securing big data environments," for a higher level view of considerations for big data security.