SailPoint

WHITE PAPER

IdentityIQ: Architected for **Performance and Scale**

Performance and scalability have always been primary considerations during development of IdentityIQ. Both are of utmost importance and as such, SailPoint takes a cross-functional approach to ensure the best possible outcomes for our customers. It is a perpetual goal of our Engineering team to develop with scalability in mind and constantly search for new ways to improve overall performance of the product.

In concert with these efforts, our Professional Services team runs a full-time performance lab dedicated to benchmarking product performance and also interacting with customers to address performance concerns. Oftentimes, discoveries made in the performance lab lay the groundwork for future product improvements.

For over 10 years, IdentityIQ has been successfully addressing the identity governance requirements of a wide range of customers. As these requirements have become more critical to our customer's organizational success and as auditors continue to expand the depth of their examination, the need to scale has never been more pervasive. We are currently managing deployments with 2+ million identities, 7+ million accounts, 17+ million assigned entitlements, and thousands of applications.

In general, there are three complementary approaches we take to managing these very important measures in creating a successful deployment for any size customer.

- The first is focusing on raw performance. This includes lowering processing times, reducing transaction times/sizes, measuring throughput, etc.
- The next is intelligently optimizing complex tasks by reducing the amount of data to be processed. Examples of this include paging on grids, delta operations for aggregation and identity refresh, and availability of filters for certain processing tasks.
- Finally, we look to provide increased configuration and visibility into operational health and status of the system. This allows for faster diagnosis of problems and greater ability to control how, when and where processes are run.

With these in mind, let's take a look at some of the features that allow the product to scale, enhancements in progress for 2018, and explore the ways we test IdentityIQ for scale.

Scalability Features

Raw Performance

Database Optimization: Through our testing process and by working on a wide variety of deployments, we are able to identify bottlenecks that affect performance of batch processes or the UI. We can frequently address issues by managing the indexes used in the database to reduce query times and provide relief for both processing resources and user experience.

Hibernate Tuning: We aim to maximize the benefit we can get from using Hibernate as our persistence manager. In this area, we have optimized cache usage and utilized the option for Dynamic Updates and Inserts. The use of dynamic updates on certain tables allowed us to reduce commit transaction sizes by as much as 95%. This can dramatically reduce the demand placed on the database server and allow the product to scale more freely.

Process Partitioning: The addition of partitioning to Aggregation, Identity Refresh, and Certification Generation allows large processes to run concurrently and complete faster. The sizes and/or number of partitions can be configured for these tasks. This allows IdentityIQ to scale horizontally. Based on the number of servers and CPU threads available in an environment, partition sizes can be adjusted to maximize utilization of those resources.

Performance Optimization

Optimized Aggregation: The addition of this option allows an aggregation task to only commit changes for accounts/identities that have changed since the last aggregation (all accounts must still be read). Doing so significantly reduces the time and resources to complete an aggregation. In a case where no changes have occurred, task time may be reduced by 99%.

Delta Aggregation: Many connectors support specific delta aggregation functionality that allows for only processing accounts which have changed since the last execution. This greatly reduces both the runtime of the task and the amount of data processed, freeing up system resources for other tasks.

Delta Identity Refresh: The introduction of this option to the Identity Refresh task allows the process to skip any identities that have not changed since last execution.

Use of Paging: Almost all components of the UI utilize paging and/or deferred loading to limit the amount of data returned from the initial request. Over time, this has become a standard practice when designing new UI models.

Object Filtering: Many tasks (identity refresh, certification generation, etc.) allow you to provide filters to ensure only the required objects are processed by the operation, resulting in more efficient utilization of resources. We have increased the number of available filters and tasks which support them from release to release.

System Monitoring

Provisioning Transactions: These objects can be created for every action sent through the Provisioner. The level at which to store these is configurable depending on your needs. Choosing to only keep failures will allow faster diagnosis of failed provisioning attempts. Choosing to keep everything will allow better forensics for auditors looking to match requests with actual actions sent to endpoint resources. Time-based pruning is configurable to prevent an overly large table.

Task Monitoring: Runtime statistics are stored on Task Definitions to provide visibility into average/expected runtime vs. current execution time. You can also see on which host a specific task was executed. All this information can be used to quickly identify environmental issues and begin the resolution process.

Host Affinity: Within a task definition, you can specify a specific host on which it executes. It can be a CSV list, where the execution will happen on the first host with an active Heartbeat. This particular feature is not applicable to partitioned tasks as the partitions are processed by the Request Processor.

System Configuration

Servers: Server objects in IdentityIQ allow each Host to register itself so they can be monitored and tuned. This object maintains the last Heartbeat of the server, CPU utilization and also allows configuring the maximum number of threads available for use.

ServiceDefinitions: Introduction of ServiceDefinitions allows controlling which hosts are running which processes. For example, it can allow you to disable the task scheduler such that no complex batch processes are run, freeing that server up to serve as a UI host. The existing services are FullText, Heartbeat, PluginSync, Request, ResourceEvent, SMListener and Task.

RequestDefinitions: RequestDefinitions allow an additional level of control as to which types of requests a host will process. For example, you may want a host to only process Aggregation requests while another host only processes Email. These definitions also allow specifying the maximum number of threads to allow on a host to avoid overloading the server.

Enhancements for 2018

Certification Generation

Historically, certifications have been generated in a "top down" manner, meaning the certification itself is created and then items are built into it. We are introducing a new "bottom up" certification generation process that we hope will eventually replace the old method. Using a bottom up approach, the entities are created independently and then grouped into a certification afterward.

This creates two specific advantages:

1. Each entity can be processed independently, reducing bottlenecks and blockages related to XL identities, granting an increased ability to partition work.
2. The independent nature of entity generation reduces contention on parent certification objects and thus will reduce lock contention. This should lead to quicker ability to generate certifications.

Additionally, we will be building in parameters to allow special handling of XL identities, either to exclude them from a given campaign or to assign them to a special owner. These options should further reduce the impact of XL identities on certification generation.

Certification Scheduling

Concurrent with the backend certification generation work, we are working to reimagine the scheduling process for a certification. The new scheduling UI will allow filtering of who and what you want to certify. These filters will allow a smaller set of identities and/or Items to be targeted, thereby reducing generation time. Today, in large environments we typically suggest breaking up very large certification campaigns. This is done primarily with Exclusion Rules, which still require processing a full population of identities and Items to then decide which to exclude. This new scheduling paradigm should provide greater flexibility to target the desired population and eliminate overhead.

Task Resiliency

In the past certain tasks, such as Identity Refresh and Aggregation, have required a complete re-execution if failure occurs during the process. For 2018, we are aiming to make these tasks more recoverable, allowing them to resume from the point at which they failed.

Admin Console – Environment Monitoring

We are expanding the Admin Console to provide Environment Monitoring capabilities. This page will provide status for applications and servers, as well as statistics for servers. Additionally, you will be able to configure the services that should be run on a particular host. Previously, this could only be done through XML editing.

Server Statistics

Server Statistics can be used to track various environmental data points that can help identify issues. Some examples are: CPU usage, memory usage and response time to the database. You can also create custom statistics that can be populated by a rule, allowing inclusion of third-party hosting data from sources like AWS Cloudwatch. These statistics can be used to trigger alerts, so an administrator can take swift action if a Host begins performing poorly. Combined with the other functions of the Admin Console, IdentityIQ can provide a good picture of how an environment is functioning. These statistics are gathered by a new ServiceDefinition called Monitoring.

Application Status

Hosts can also be configured to periodically check connectivity to endpoint resources. IdentityIQ will track the success of the response and also the time taken to receive it. Having this information can help to identify issues with Aggregation or Provisioning tasks.

Testing Performance and Scalability

As new features are added to the product we continue to monitor application performance across several key areas. We have several methods in place for testing the performance and scalability of the product.

Manual

An important part of our quality process is ensuring a first-rate end user experience. This requires extensive manual testing. During this process, the test team designs thorough sets of functional, integration and regression test plans. These tests take into consideration the possible scale concerns of a new feature (to make sure the UI and backend both perform well) and upgrade considerations.

Automated

The IdentityIQ Engineering team runs nightly tests across all of our supported database platforms to keep constant watch on performance metrics for key tasks like Aggregation, Identity Refresh and Certification Generation. Our nightly dataset includes 125k+ identities, 3M+ identity entitlements, 110k unique entitlements, 500k accounts, and 5k+ manager certifications. We also have datasets including XL identities and other known complicated data shapes.

Performance Lab

Perhaps most importantly, SailPoint has an entire team dedicated to evaluating and tuning the performance and scalability of IdentityIQ. This team is part of the Professional Services organization but works closely with Engineering. Through extensive customer interactions and data points, this team has a unique view into common challenges customers encounter during deployment and

maintenance of the product. In addition to benchmarking performance results, this team also engages in scale concurrency testing of the UI to aid in making server recommendations to customers.

Conclusion

SailPoint will continue to focus on performance and scalability with each release as our customer base and the size of their deployments continues to grow. As new requirements emerge, and new technologies evolve to address them, we are committed to constant improvement towards meeting any challenge. Scaling is not just a numbers game. Successfully managing a large deployment requires thoughtful analysis of the desired use cases, the shape of the data and the capabilities of the product. The partnership between SailPoint and our customers is key to identifying the best approach. Our greatest success is that of our customers.

Roadmap Disclaimer: The development, release, and timing of any features or functionality described for our products that are not currently available remains at our sole discretion on a when, and if available, basis and may not be delivered at all and should not be relied on in making a purchasing decision.

WP1226-1806