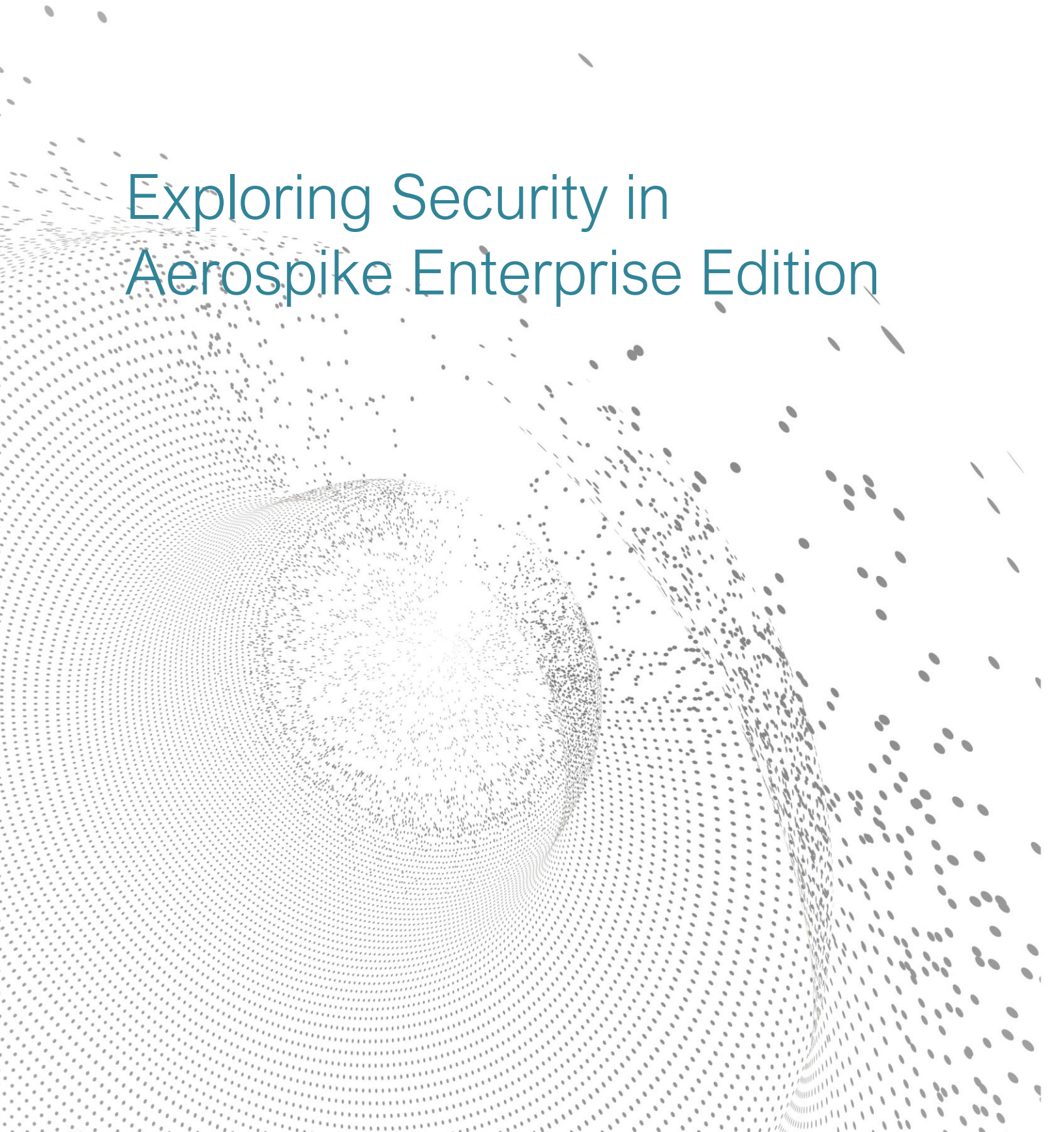
AEROSPIKE

# Exploring Security in Aerospike Enterprise Edition



# Table of Contents

# Executive overview

When it comes to securing operational data, it pays to play it safe. And Aerospike clients in financial services, retail, telecommunications, adtech and other industries are doing just that, thanks to the system's comprehensive suite of security features that protect client data from unauthorized or inappropriate access. Aerospike even enables administrators to define fine-grained audit trails to track important system events, such as attempted logins and data access operations.

Indeed, Aerospike's operational database management platform provides strong, enterprise-grade security for both data center and cloud deployments. But perhaps that's no surprise – Aerospike has distinguished itself for years with its ultra-fast data access speeds, remarkable cost efficiency, and extremely high availability and reliability. Of course, these features alone aren't sufficient to prompt multi-national firms to entrust their mission-critical applications to Aerospike. Without appropriate mechanisms for safeguarding client data, Aerospike wouldn't enjoy the wide deployment it now has around the globe.

This paper helps you explore the security features that Aerospike offers. These include mechanisms for authenticating users to the system, authorizing users to perform specific operations, encrypting data at rest and in motion, and more. If you're not a database security specialist, don't worry – this paper will explain basic concepts as needed. And if you're not already familiar with Aerospike, a separate white paper will quickly bring you up to speed on its technology and help you understand how firms are using its platform to cut their server footprints up to 90% and enjoy total cost of ownership (TCO) savings of $1 to $10 million *per application* compared with other solutions.

# About security

Conceptually, database security needs are simple: firms want to ensure that the right people have the right level of access to the right data. But fulfilling those needs isn't always straightforward. That's because safeguarding mission-critical data requires strong mechanisms to authenticate a prospective user, limit a user's access to authorized operations, encrypt sensitive information as it flows over the network and (possibly) when it's stored persistently, and audit (or log) various activities that occur in the database. Let's look at these areas a little more closely.

**Authentication** involves validating the identity of someone attempting to access the system, often by requiring a user to log in with a valid ID and password that an administrator has previously created. The authentication process may be strictly internal to the system, requiring an administrator to create an account for each database user, or it may involve some external process, in which an administrator uses a service outside the DBMS to define valid users and instructs the DBMS to integrate its security checks with this service. The latter can be particularly convenient for managing large numbers of users who need access to a variety of different enterprise applications and services. The Lightweight Directory Access Protocol (LDAP) is one popular means for maintaining a centralized collection of user accounts and passwords for various application and system services.

**Authorization** ensures that authenticated users can access only the resources they're permitted to access. Typically, administrators confer a set of **privileges** or **permissions** to a given user; these privileges regulate whether or not that user can read from or write to a certain collection of data or perform specific database operations, such as create new user accounts, change configuration settings, create or drop database objects, etc. For convenience, some DBMSs support **roles** with pre-defined sets of privileges that can be assigned to users.

**Encryption** protects sensitive data from being compromised by encoding it in some way. Subsequent data processing requires a corresponding decryption key; without it, the data's contents can't be deciphered. In a clustered computing environment, it's important to consider **transport-layer security** (TLS) encryption

services, which protect data as it flows over the wire between server nodes as well as between the client and the server. In addition, some firms may require encryption for data at rest – i.e., data stored on a traditional hard disk or in non-volatile memory, such as a solid-state drive (SSD).

An audit trail records various system events defined by an administrator, such as successful or unsuccessful attempts to read or write specific data. Each logged event typically includes several pieces of information, such as the nature of the event, a timestamp, user data, point of origin, and so on. Audit trails provide for accountability, help firms detect potential intrusions, aid enterprise data provenance efforts, and support problem analysis. Because audit trails introduce additional I/O overhead to user transactions and other operations, it's important for a database platform provider to offer a fine-grained selection of which activities (if any) to track.

## Aerospike's approach to security

Aerospike provides a strong set of security mechanisms to help enterprises safeguard their mission-critical operational data. In a moment, you'll have a chance to explore its approach to authentication, authorization, encryption, and auditing. But first, if you're not already familiar with Aerospike, let's quickly review the basics.

Aerospike is a distributed non-relational DBMS that delivers extremely low data access latencies for operational data sets that span billions of records in databases of 10s – 100s TB. Its patented hybrid-memory architecture™ (HMA) delivers exceptional performance using a much smaller server footprint than alternate solutions. As shown in Fig. 1, Aerospike uses dynamic random access memory (DRAM) for index and user data. Optionally, applications can commit each write directly to fast, non-volatile memory (SSDs), which Aerospike treats as raw devices for speed and efficiency. Sophisticated (and automatic) data distribution techniques, a "smart client" layer, and other features further drive Aerospike's speed and cost efficiency. For more about Aerospike's overall architecture and capabilities, see this white paper.

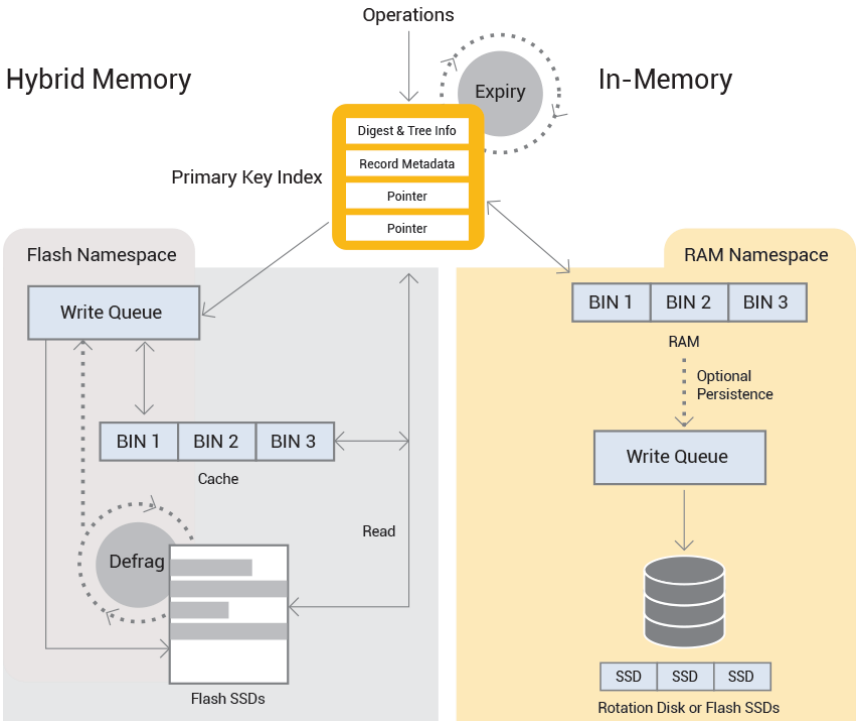



*Figure 1: Overview of Aerospike architecture*

Global firms have deployed Aerospike to support systems of engagement and to act as a system of record. Systems of engagement foster collaboration and interaction; they're often driven by social media, mobile device usage, cloud computing, real-time chats, and streaming data. Systems of record support more classic data management applications; they serve as the authoritative record for critical business data, such as account information or sales records.

Many firms deploy both types of systems. And increasingly, both impose some common – and aggressive – demands on their operational database platforms, such as:

- Service-level agreements (SLAs) that require sub-millisecond database response times.
- High throughput for mixed workloads (e.g., 3 – 5 million operations per second).
- Support for managing billions of business records in databases of 10s – 100s TB.
- High scalability for handling unpredictable increases in data volumes and transactions.
- Low total cost of ownership (TCO).

Enterprises in banking, wealth management, telecommunications, retail, travel, and other industries are using Aerospike in production to fulfill such ambitious requirements. For example, Aerospike is powering production applications for intra-day trading, real-time fraud detection, real-time telecommunications billing, real-time bidding for online advertisements, web-based personalization services, and more.

Let's explore how Aerospike enables its enterprise clients to protect their operational data from unauthorized access as well as track important system events, such as failed attempts to violate user authentication, failed attempts to access unauthorized data, successful changes to sensitive business data, and more.

## Authentication

Aerospike provides internal authentication services and can integrate with an external authentication system – specifically, LDAP – to verify user identity. Configuration settings enable administrators to specify which approach the system will use.

Creating a new Aerospike user account is simple. The following example illustrates how an administrator of an Aerospike cluster configured for internal authentication would create an account for user "chris" with a password of "foo".

```
aql> # create a user
aql> create user chris password foo
OK
```

The SHOW USER command verifies that "chris" is known to the system. Because the account was created without any specific roles (sets of privileges), none are cited in the report. We'll discuss roles and privileges in a separate section.

```
aql> show user chris
+-------+-------+
| user  | roles |
+-------+-------+
| chris | ""    |
+-------+-------+
```

Aerospike stores the user names and a hash of corresponding passwords on every node in the cluster. When a user logs in, the Aerospike client software applies the same hash to the supplied password and sends it

across the network to the target node. Maintaining a copy of security information in a local file on each node provides for the lowest possible read latency and highest level of availability.

Optionally, administrators can override Aerospike's default security settings to configure Aerospike for LDAP authentication or "hybrid" authentication, which supports both local and LDAP-based user accounts. Details on how to configure Aerospike for LDAP authentication can be found in the product documentation. Note that Aerospike strongly recommends – and fully supports – transport-layer security (TLS) between the client and server when using external LDAP authentication because the user's plain text password must be sent to LDAP for authentication. Aerospike never stores the LDAP password on its server nodes.

When operating with LDAP authentication, Aerospike maps the supplied username to an LDAP DN (distinguished name), which is used for the authentication (bind) request. To accommodate various LDAP configurations, administrators can define a "user-query-pattern" to easily resolve a situation in which the supplied username is not in the LDAP DN, which often occurs when using LDAP Active Directory or when user DNs are in different LDAP organizational units (OUs).

## Authorization: roles and privileges

Closely associated with user authentication is user authorization, which defines the operations a given (authenticated) user can perform. In Aerospike, users can be assigned to various **roles**, each of which confer a set of privileges. Administrators can also create new roles, essentially combining different collections of privileges under one role that can be assigned to users.

Aerospike offers 6 pre-defined roles that address common collections of privileges that an administrator might want to grant to a user. Note that some roles are cumulative – e.g., a user with **read/write** privileges can perform all operations associated with **read** privileges and is further granted **write** authority. Here's a short summary of Aerospike's pre-defined roles:

- **Read.** This encompasses various types of read operations against a specific user data set (e.g., get a record, scan a data set, query a data set) and also permits the user to read server configuration data. In addition, users can change their own passwords.
- **Read/write.** This includes all read role privileges as well as the ability to perform various write operations against a specific data set.
- **Read/write/UDF execution.** In addition to read/write privileges, this role enables users to execute user-defined functions (UDFs) directly or to execute queries that invoke UDFs.
- **Data administration.** This includes creating and dropping various database objects, using a monitoring system, aborting scans and queries, and changing user passwords.
- **System administration.** In addition to all data administration privileges, this role supports setting dynamic server configuration variables and enabling specialized logging (i.e., auditing).
- **User administration.** As the name implies, this role governs management of user accounts – e.g., creating and dropping users, granting and revoking roles to users, querying information about users and their roles, etc.

As we just mentioned, someone with **user administration** authority can create new roles based on those supplied by Aerospike and confer these roles to users as desired. Indeed, the process is simple and straightforward, as shown in the following example that creates two new roles spanning read/write/UDF execution privileges for setA and setB in namespace (database) "test".

```
aql> # create a role with read-write-udf privileges on set "setA" in namespace "test"
aql> create role setA-user privileges read-write-udf.test.setA
OK
aql> show role setA-user
+------------+--------------------------+
```

```
| role        | privileges                 |
+-------------+----------------------------+
| "setA-user" | "read-write-udf.test.setA" |
+-------------+----------------------------+
1 row in set (0.000 secs)
OK
aql> # create a role with read-write-udf privileges on set "setB" in namespace "test"
aql> create role setB-user privileges read-write-udf.test.setB
OK
```

With these two roles defined, it's easy to create a new user account for these roles. The following example creates a new account for "fred" (with a password of "fredspwd") and assigns this user three roles: user-admin (an Aerospike pre-defined role for user administration), setA-user, and setB-user. The latter two roles were just created in the previous example.

```
aql> # create a user with several roles
aql> create user fred password fredspwd roles user-admin,setA-user,setB-user
OK
aql> show user fred
+--------+-----------------------------------+
| user   | roles                             |
+--------+-----------------------------------+
| "fred" | "setA-user, setB-user, user-admin" |
+--------+-----------------------------------+
1 row in set (0.001 secs)
OK
```

If using LDAP, an Aerospike administrator links Aerospike roles to desired LDAP accounts to regulate the behavior of authenticated users.

## Encryption

Aerospike supports the ability to encrypt – or encode – data as it flows across the network and (optionally) when it is persisted to storage. The former is sometimes referred to as encrypting "data in flight" or "data on the wire," while the latter is sometimes referred to as encrypting "data at rest." Both capabilities are important for security.

Aerospike can encrypt data as it flows across the network through its transport-layer security (TLS) services, which comply with stringent industry standards such as TLS 1.2 and AES-256 (Advanced Encryption Standard with 256-bit keys). Aerospike's TLS implementation addresses network traffic:

- Between clients and a server cluster
- Between nodes in a server cluster
- Across server clusters configured for cross-data center replication.

Configuring Aerospike TLS is straightforward: administrators generate appropriate certificates and specify certain parameters in an Aerospike configuration file. Certification authentication is bidirectional for maximum safety. For client-to-server cluster services, administrators can even choose from 3 types of authentication options.

But encrypting data on the wire is only part of the story. Some firms require encryption of highly sensitive data at rest, too. For such situations, Aerospike can be configured to use symmetric AES-128 encryption technology on storage devices. Administrators simply specify an encryption key file containing the key

needed to encrypt/decrypt any data stored in the target database (namespace). Aerospike recommends storing this key file on encrypted and secure operating environments for maximum protection. Although administrators typically activate Aerospike's data-at-rest encryption feature on an empty database, it's also possible to enable or disable data-at-rest encryption on a populated, operational database without suffering any downtime.

As you might expect, the performance impact of encrypting data-at-rest depends on your environment, including the hardware you've deployed and the average size of your data records. To reduce runtime performance impact, Aerospike encrypts data on a per-record basis rather than a per-disk sector basis. Still, encrypting data-at-rest can come at a cost, so it's best to benchmark the potential impact using your own data and deployed hardware. However, Aerospike ran several tests using Intel Xeon processors, which feature encryption capabilities built into the hardware to accelerate runtime performance. Aerospike observed a 20% performance decrease of transactions per second (TPS) with record sizes smaller than 512 bytes, while there was no measurable performance loss with record sizes of 1 KB or greater.

## Audit trail

The ability to track system events provides important accountability and diagnostic capabilities. That's why Aerospike enables administrators to define and configure audit trails to log attempted and successful database operations to local files, the Aerospike log file, or the default sink.

As you might imagine, writing audit log records introduces additional I/O overhead, which is why Aerospike supports quite granular levels of audit trail definitions. Simply put, administrators choose what they want to audit by adding entries to a configuration file, specifying actions such as:

- Security violations (including authentication failures and role violations).
- Successful authentications.
- Successful or attempted data operations (including various write/read operations). For maximum flexibility, administrators can limit such audit records to specific data sets in specific namespaces (databases).
- User administration operations (including creating/dropping users, granting/revoking user privileges, creating/dropping roles, etc.).
- System administration operations (including creating/dropping secondary indexes, registering/removing UDFs, changing dynamic server configurations, etc.).

In general, auditing security violations and other seldom-occurring events will have no discernible performance impact, while logging every attempted or successful data operation on numerous data sets can slow runtime performance in systems under heavy load as well as increase storage requirements for the logs. It's best to measure potential performance impact and storage needs in a test environment if you plan to maintain extensive audit records.

Aerospike designed its audit records to be easy to interpret. For example, here's a record indicating that on Jan. 12, 2018 someone from client IP address 127.0.0.1 port 38405 attempted to log in as "admin" but failed to provide a valid password, causing an authentication failure:

```
Jan 12 2018 13:31:14 GMT: INFO (security): (security.c::5023) authentication failed
(password) | client: 127.0.0.1:38405 | authenticated user: <none> | action:
authentication | detail: user=admin
```

Of course, successful operations can be logged, too. This example shows that "user1" logged into and authenticated with Aerospike on Oct. 9, 2018 from client IP address 127.0.0.1 port 47694. This user then wrote a record to the "testset" data set.

```
Oct 09 2018 04:22:36 GMT: INFO (security): (security.c:5482) permitted | client:
127.0.0.1:47692 | authenticated user: user1 | action: login | detail: user=user1
Oct 09 2018 04:22:36 GMT: INFO (security): (security.c:5482) permitted | client:
127.0.0.1:47694 | authenticated user: user1 | action: authentication | detail:
user=user1
Oct 09 2018 04:22:38 GMT: INFO (security): (security.c:5482) permitted | client:
127.0.0.1:47694 | authenticated user: user1 | action: write | detail: {test|testset}
[D|f59124986e96ad175b374c9487945bbcad537b74
```

# Summary

Firms shouldn't have to sacrifice the security of their operational data to get extremely fast performance, low total cost of ownership (TCO), and high data availability. And, thanks to Aerospike, they don't need to make such compromises. Today, firms in financial services, telecommunications, retail, adtech and other industries are using Aerospike's comprehensive suite of security features to safeguard their mission-critical data while continuing to enjoy extremely low data access latencies, high availability, and remarkable cost efficiency. Authentication, authorization, auditing, and data encryption services are among the enterprise-grade security features that Aerospike clients are using to support their systems of engagement as well as systems of record deployed on premises or in the cloud.

And while Aerospike isn't the only alternative for managing real-time operational data, it has set itself apart from other alternatives on several fronts. Mainframe solutions may be secure and reliable, but they're also expensive to maintain and difficult to adapt to evolving business needs. Similarly, relational DBMSs offer strong security features and a familiar data model well-suited to certain business applications, but they seldom deliver the speed needed for real-time workloads, even with extensive (and costly) tuning efforts. Two-tier solutions that couple a caching layer with an operational data store (relational or otherwise) alleviate some performance challenges but introduce administrative headaches and high ownership costs.

Aerospike's database platform delivers extremely low data access latencies (often sub-millisecond) for read/write workloads over large volumes of operational data at a fraction of the TCO of other solutions. And it does so without forcing administrators to put their sensitive business data at risk.

Want to learn more about Aerospike's security features, ultra-fast performance, and low TCO? Contact Aerospike to schedule a briefing or discuss projects. Want to explore Aerospike independently? Visit Aerospike's web site resources to peruse free on-demand webinars, white papers, videos, and other materials. You can even get your hands dirty by downloading the free Community Edition.

Enterprises around the world are already enjoying tangible business results by modernizing their IT infrastructures with Aerospike. Why not explore what Aerospike can do for you?

# Resources

Aerospike documentation feature guide: security

Aerospike documentation operational manual: security

Maximize the Value of Your Operational Data, Aerospike white paper

AEROSPIKE

## About Aerospike

Aerospike is trusted by leading enterprises around the world to help them confidently deploy mission critical, strategic operational applications that make digital transformation possible. Our enterprise-grade database is deployable anywhere, delivers unmatched uptime, predictable performance, and exceptionally low TCO. Aerospike has customer deployments that have run for years with no service disruption, handling hundreds of terabytes of data, supporting trillions of transactions per month, with sub-millisecond latency. Aerospike customers include Adobe, Airtel, FlipKart, Kayak, Nielsen, Nokia, and Snap.



2525 E. Charleston Road, Suite 201
Mountain View, CA 94043
Tel: 408 462 AERO (2376)
www.aerospike.com