Security in couchbase

ANJU MUNOTH

Authentication

- To access Couchbase Server, users must be authenticated.
- Authentication - Identifying who is attempting to access a system.
- Subsequent to successful authentication, authorization can be performed, whereby the user's appropriate access-level is determined.
- Can be performed by means of a username and password, assigned to each administrator or application.
- Can also be performed by means of X.509 Certificates: these support Transport Layer Security, by establishing the identity of a client or server through digital signatures.
  - They also provide keys to support on-the-wire encryption, according to the conventions of Public Key Infrastructure (PKI).
- Couchbase Server assigns each user to one of two authentication domains: the local domain consisting of users whose credentials are maintained by Couchbase Server itself; the external domain consisting of users whose credentials are maintained remotely — for example, on an LDAP server.

Authentication Options

- Couchbase-Server authentication typically relies on a *username* and a *password*, which must be passed into the system by the *user* (meaning, the administrator or application) that is attempting access.
- Specified username and password must match ones already defined
- Must be accessible either on the Couchbase-Server cluster itself, or *externally*.
- External accessibility, which is available only with the Enterprise Edition of Couchbase Server, means either:
    - On a network-accessible directory-server, by means of the *Lightweight Directory Access Protocol* (LDAP).
    - By means of the *Pluggable Authentication Modules* (PAM) authentication-framework.
- If a match is achieved, the user is thereby recognized, and so *may* be granted access. If no match is achieved, the user is denied access.

Users, Usernames, and Passwords

- To access Couchbase Server, all users and applications must authenticate by means of a *username* and *password*.
- *Full Administrator* username and password are established during initialization of Couchbase Server.
- Subsequently, additional users can be added to the cluster as *local* users: each is at that time assigned a unique *username*, and a unique *password*.
- Passwords can be changed by means of the password-reset tool, reset-admin-password; or by Couchbase Web Console
- Users can also be added to the cluster as *external* users, for whom no password need be specified; since the external user is to be authenticated externally.

Usernames and Passwords

- User may be either an *administrator* or an *application*.
  - An application may be a program or server, or may be a simple, single command-line query.
- *Full Administrator* who installs and configures Couchbase Server (and so has full read-write access to the whole system), defines their own username and password during the configuration-process
- Subsequently, this administrator can add additional administrators to the system; assigning a username and password to each.
- Whenever any needs to log into Couchbase Web Console in order to inspect data, statistics, and settings (and possibly make changes), they must specify their own unique username-password combination, at the authentication-prompt provided by the server.
- An application, if it is not using a client certificate for purposes of authentication, must pass its username and password as parameters.
- Couchbase CLI commands, N1QL queries, and executables supported by the Couchbase SDK all provide syntax to allow the passing of a username and password.

Users

- Cluster running Couchbase Server *Enterprise Edition* can have any number of users.
- A cluster running *Community Edition* can have a maximum of twenty users.

Usernames and Passwords for Administrators

- When an administrator logs into Couchbase Web Console, if the console is running on the default port, http://localhost:8091, the specified username and password are passed in the clear, from the browser to the console.

- Optionally, Couchbase Web Console can be configured for secure access, at https://localhost:18091; so that the username and password are passed in encrypted form

Usernames and Passwords for Applications

- To pass credentials, applications must use one of four mechanisms provided by the *Simple Authentication and Security Layer* (SASL) framework.
- PLAIN, and three members of the *Salted Challenge Response Authentication Mechanism* family of hash functions; which are SCRAM-SHA1, SCRAM-SHA256, and SCRAM-SHA512.
- SCRAM mechanisms allow applications to authenticate securely, by transmitting the password only in *protected form*.
- Drivers may need to be updated, to support SHA-based hash functions

Usernames and Passwords for Applications

- In ascending order of strength, the Couchbase password-authentication mechanisms are as follows:
- *PLAIN*: The client sends the password in unencrypted form. All clients support this authentication-method.
  - It is insecure, providing no defence against passwords being stolen in transmission.
- *SCRAM-SHA1*: Uses a 160-bit key.
- *SCRAM-SHA256*: One of a group of hash functions referred to as *SHA2*, SCRAM-SHA256 uses a 256-bit key.
- *SCRAM-SHA512*: Another hash function from the *SHA2* group, SCRAM-SHA512 uses a 512-bit key; and is the strongest supported authentication protocol.

Usernames and Passwords for Applications

- During initial client-server negotiation, the strongest authentication protocol supported by both Couchbase Server and the application's client OS is selected for use.
- For example, if the client supports only the PLAIN protocol, the PLAIN protocol is used; but if the client also supports the SCRAM-SHA1 protocol, then SCRAM-SHA1 is used.
- A challenge-response method can be transmitted through both encrypted and unencrypted channels.
- Note that the SCRAM challenge-response protocols authenticate only the process of password-validation. To secure the subsequent session, TLS should be used

Certificate-Based Authentication

- Couchbase Server supports the use of x.509 certificates, to authenticate clients.
- Ensures that only approved users, machines, or endpoints are authenticated.
- Certificate-based authentication relies on a *Certificate Authority* (CA) to validate identities and issue certificates.
- Certificate includes information such as the name of the entity it identifies, an expiration date, the name of the CA that issued the certificate, and the digital signature of the issuing CA.

Authentication Domains

- Couchbase Server authenticates each user by means of one of two *authentication domains*.
- *Local*
- *External*

Local

Contains users defined locally. This includes:

- *Full Administrator* for Couchbase Server.
- *Locally Defined Users*, which are explicitly created by a Couchbase Server administrator; and each feature a username and password unique within the Local domain.
- *Internal Components* within Couchbase Server that support core functionality (for example, indexing, searching, and replicating), and run with full administrative privileges.
- *Generated Users*, which are created by Couchbase Server as part of the upgrade process from pre-5.0 to 5.0 and post-5.0 versions; each in correspondence with a legacy bucket.
- Each Generated User is assigned a *username* that is identical to the bucket-name; and either a *password* that is identical to the bucket's pre-5.0 password, or *no password*, if the bucket did not feature a password.
- Generated Users are created to ensure that legacy applications can continue to access legacy buckets after upgrade to 5.0 or post-5.0, with the same username-password combination being used for authentication.

External

- Contains either or both of the following:
- Users that are explicitly registered on Couchbase Server as *external*; as supported either by *LDAP* or *PAM*.
- Usernames and passwords are defined and stored remotely; with the usernames also stored on Couchbase Server.
  - Note that external usernames do not clash with local usernames.
- Users that are not defined or registered on Couchbase Server in any way, and are defined entirely on LDAP.
  - In this case, *Native LDAP Support* must have been used to configure Couchbase Server's access to LDAP, with *LDAP Group Support* enabled.
  - If one or more of the user's LDAP Groups has been *mapped* to a corresponding Couchbase-Server user-group, the user can be authenticated on the LDAP server, and then be granted the roles assigned to each of the user-groups to which a mapping has been made.

Authorization

- Couchbase-Server features — including data, settings, and statistics — can be accessed only by users who have been assigned the appropriate *privileges*.
- Privileges include *read*, *read-write*, *execute*, and *manage*.
- Privileges are assigned by *Full* and *Security* Administrators, in correspondence with *roles*.
- When a user successfully authenticates, their assigned roles are examined, and access is granted or denied by Couchbase Server.
- Roles can be assigned to a user in either or both of two ways:
- *Directly*. The user is associated directly with one or more Couchbase-Server roles.
- *By Group*. A Couchbase-Server *user-group* is defined, and roles are assigned to the user-group.
  - The user is made a member of the user-group, and thereby inherits all the roles of the group. A user can be a member of any number of groups.

Role-Based Access Control

- Couchbase provides Role-Based Access Control (RBAC), in which access privileges are assigned to fixed roles; which are in turn assigned to users (each of which may be an administrator or an application) either directly; or indirectly, by means of user-groups.
- Couchbase Server Enterprise Edition provides RBAC with multiple roles for finer access control.
- Community Edition provides multiple users that can be assigned to limited set of roles.
- There are three fixed roles in the community edition of Couchbase providing coarser access control: Bucket Full Access (bucket_full_access[*]), Admin (admin), and Read Only Admin (ro_admin).

RBAC

- *Resource*: An entity the access to which must be controlled. A resource can be specified either individually, by name; or as a group (for example, all buckets), by means of a wildcard character.
- *Privilege*: The right, assigned by Couchbase Server, to apply an action to a resource. Possible actions include *read*, *write*, and *execute*.
- *Role*: An entity associated with a fixed set of privileges.
- *User*: An identity, recognized by Couchbase Server, based on the passing of a *username* and *password*.
  - A user can be assigned one or more *roles*: the privileges associated with each assigned role determine the resource-access granted the user.
  - Users can be *local* (defined on Couchbase Server) or *external* (defined on a remote, network-accessible system). Each user might be an administrator or an application.

RBAC Security Model

- Couchbase RBAC controls access to cluster-resources.
- Resources can only be accessed by users.
- A user may be an administrator or an application.
- Users can be added to Couchbase Server by the Full Administrator.
- Each user must be defined with a username and password.
- When attempting to access resources, each user must authenticate by means of these credentials.
- A user can be assigned one or more roles by the Full Administrator.
- Each role is itself associated with a subset of privileges; a privilege being a form of action, such as Read, Write, Execute, or Manage.
- Each privilege is associated with a resource; such as a bucket, index, view, or DCP stream.

Roles

- A Couchbase role permits one or more resources to be accessed according to defined privileges.
- Couchbase roles each have a fixed association with a set of one or more privileges.
- Each privilege is associated with a resource.
- Privileges are actions such as Read, Write, Execute, Manage, Flush, and List; or a combination of some or all of these.

Roles are of two kinds

- *Administration and Global*:
  - Associated with cluster-wide privileges.
  - Some of these roles are for administrators; who might manage cluster-configurations; or read statistics; or enforce security.
  - Others are for users and user-defined applications that require access to specific, cluster-wide resources.
- *Per Bucket*:
  - Associated with one or more buckets.
  - Roles support the reading and writing of bucket-settings; the management of services, indexes, and replication procedures; and access to data.
- When a user (meaning either an administrator or an application) attempts to access a resource, they must authenticate, by means of a username and password.
- Roles and privileges associated with these credentials are checked by Couchbase Server.
- If the associated roles contain privileges that support the kind of access that is being attempted, access is granted; otherwise, it is denied.

Commonly Used Roles

- Administrators. Able to log into Couchbase Web Console and perform administrative tasks; but unable to read or write data.

- The administrative tasks available are divided into multiple admin roles. For example, the Cluster Admin role allows the management of all cluster features except security; while the Read-Only Admin role allows only the reading of statistics; and the Bucket Admin role allows management only of one or more buckets. See the Admin roles listed below for full details. Note that depending on the administrator's assigned roles, the content of Couchbase Web Console changes: for example, the Security screen is only visible to Full or Security administrators.

- Applications. Able to read or write data; but unable to log into Couchbase Web Console, or in any way modify cluster-settings. For example, the Data Reader and Data Writer roles allows data to be respectively read and written to one or more buckets. Other application-intended roles are Application Access, Data DCP Writer, Data Backup & Restore, and Data Monitor. See below for details on each.

- Developers. Can be given a selection of roles, allowing the right degree of data and console access. For example, the Read-Only Admin role allows the reading of cluster-statistics, while the Data Read and Data Write roles allow access to data on one or more buckets.

Commonly Used Roles

Administrators. Able to log into Couchbase Web Console and perform administrative tasks; but unable to read or write data.

- Administrative tasks available are divided into multiple admin roles.
- Cluster Admin role allows the management of all cluster features except security;
- Read-Only Admin role allows only the reading of statistics; and the Bucket Admin role allows management only of one or more buckets.

Applications -- Able to read or write data; but unable to log into Couchbase Web Console, or in any way modify cluster-settings.

- Data Reader and Data Writer roles allows data to be respectively read and written to one or more buckets.
- Other application-intended roles are Application Access, Data DCP Writer, Data Backup & Restore, and Data Monitor.

Developers -- Can be given a selection of roles, allowing the right degree of data and console access.

- Read-Only Admin role allows the reading of cluster-statistics, while the Data Read and Data Write roles allow access to data on one or more buckets.

Access-Controlled Resources

- The following Couchbase Server-resources are always access-controlled:
- **Clusters**.
- **XDCR Cluster References**.
- **Query Service**.
- **Analytics Shadow Data Sets**.
- **System Catalogs**.
- **Buckets**.
- **XDCR Bucket Replication**.
- **Indexes**. Including *Views*, *Primary Indexes*, *Global Secondary Indexes*, and Search *Indexes*.
- **UI Access**. Allows login to *Couchbase Web Console*. The features available are role-dependent.
- **Curl Access**. Allows execution of the N1QL CURL function by externally authenticated users.
- **Eventing**. Allows configuration and scheduling of the *Eventing Service*.
- **Pools**.

Roles and Privileges

Full Admin
Cluster Admin
Security Admin
Read-Only Admin
XDCR Admin
XDCR Admin
Query Curl Access
Query System Catalog
Analytics Reader
Analytics Admin
Bucket Admin

Query Select
Query Update
Query Insert
Query Delete
Query Manage Index
Search Admin
Search Reader
Analytics Select
Analytics Manager

Application Access
XDCR Inbound
Sync Gateway
Data Reader
Data Writer
Data DCP Reader
Data Backup & Restore
Data Monitor
Views Admin
Views Reader

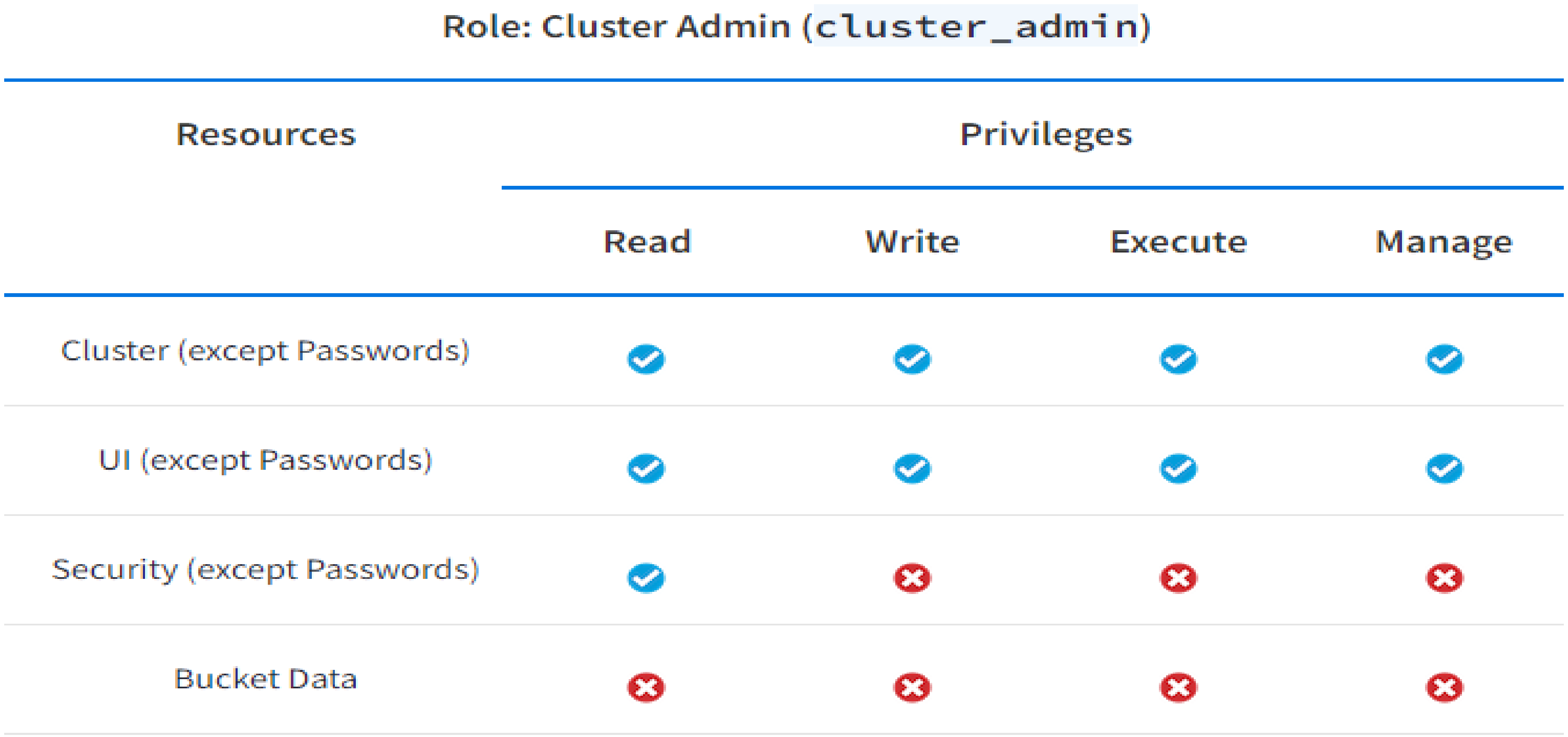
Full Admin

- The **Full Admin** role (an *Administration and Global* role) supports full access to all Couchbase-Server features and resources, including those of security.
- The role allows access to the Couchbase Web Console, and allows the reading and writing of bucket-data.
- This role is also available in Couchbase Server *Community Edition*.

Cluster Admin

- **Cluster Admin** role (an *Administration and Global* role) allows the management of all cluster features except security.
- The role allows access to Couchbase Web Console, but does not permit the writing of data

Cluster Admin



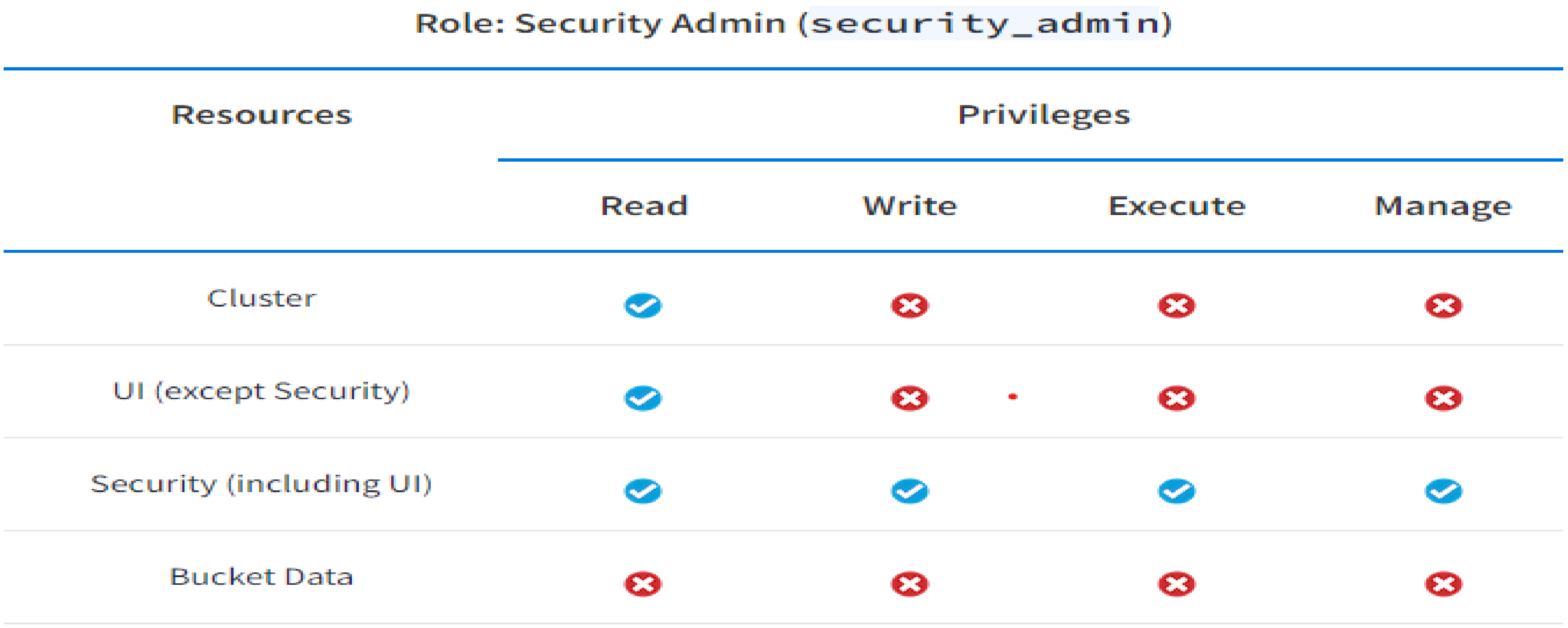
Role: Cluster Admin (`cluster_admin`)

Resources	Privileges			
	Read	Write	Execute	Manage
Cluster (except Passwords)	✓	✓	✓	✓
UI (except Passwords)	✓	✓	✓	✓
Security (except Passwords)	✓	✗	✗	✗
Bucket Data	✗	✗	✗	✗

Security Admin

- The **Security Admin** role (an *Administration and Global* role) allows the management of user roles and the reading of all cluster statistics.
- Role does not permit the granting of **Full Admin** or **Security Admin** roles, and does not permit the administrator to change their own role (which therefore remains **Security Admin**).
- Role supports access to Couchbase Web Console, but does not support the reading of data.

Security Admin



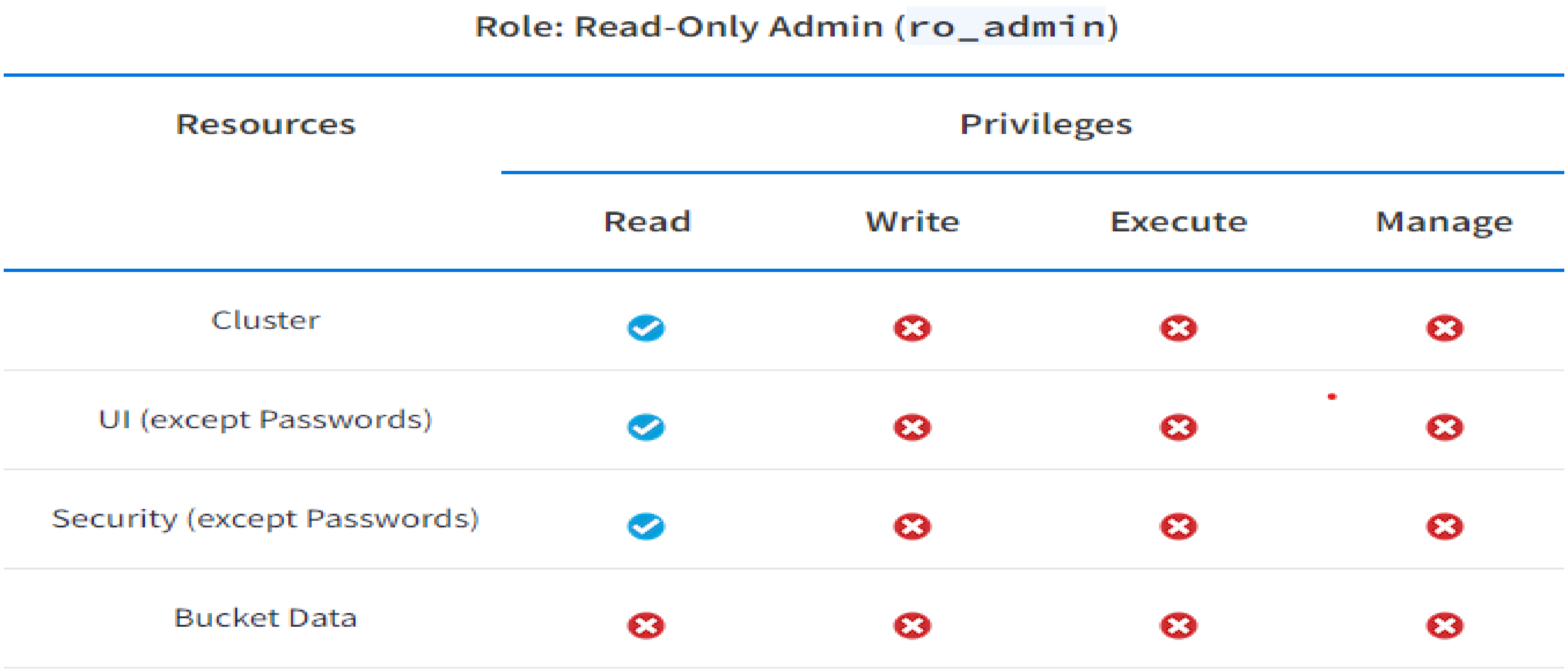
Role: Security Admin (`security_admin`)

Resources	Privileges			
	Read	Write	Execute	Manage
Cluster	✓	✗	✗	✗
UI (except Security)	✓	✗	✗	✗
Security (including UI)	✓	✓	✓	✓
Bucket Data	✗	✗	✗	✗

Read-Only Admin

- The **Read-Only Admin** role (an *Administration and Global* role) supports the reading of Couchbase Server-statistics: this includes registered usernames with roles and authentication domains, but excludes passwords.
- The role allows access to Couchbase Web Console.
- This role is also available in Couchbase Server *Community Edition*.

Read-Only Admin



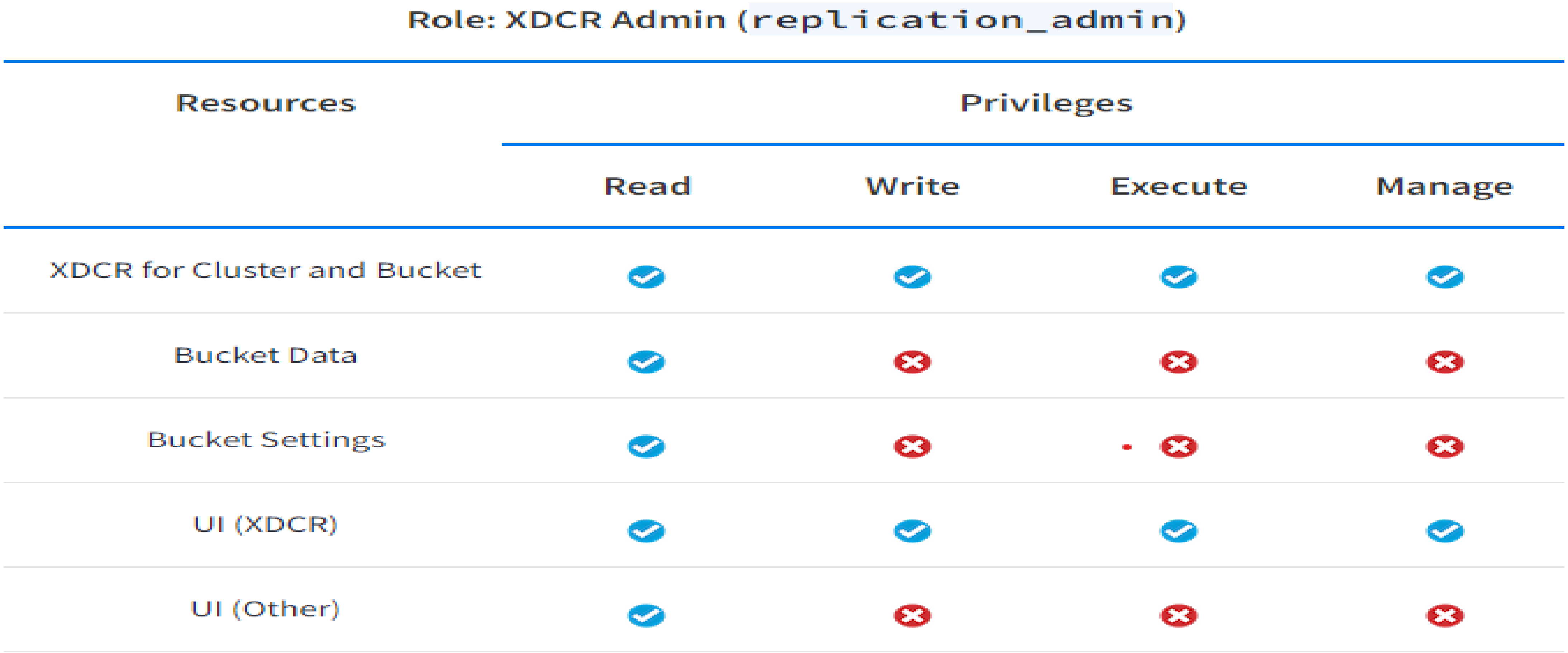
Role: Read-Only Admin (`ro_admin`)

Resources	Privileges			
	Read	Write	Execute	Manage
Cluster	✓	✗	✗	✗
UI (except Passwords)	✓	✗	✗	✗
Security (except Passwords)	✓	✗	✗	✗
Bucket Data	✗	✗	✗	✗

XDCR Admin

- The **XDCR Admin** role (an *Administration and Global* role) allows use of XDCR features, to create cluster references and replication streams.
- The role allows access to Couchbase Web Console.

XDCR Admin



Role: XDCR Admin (`replication_admin`)

Resources	Privileges			
	Read	Write	Execute	Manage
XDCR for Cluster and Bucket	✓	✓	✓	✓
Bucket Data	✓	✗	✗	✗
Bucket Settings	✓	✗	✗	✗
UI (XDCR)	✓	✓	✓	✓
UI (Other)	✓	✗	✗	✗

Encryption in Couchbase Server

- data is encoded such that it is non-readable, other than by authorized parties who possess the appropriate means of *decryption*.
- Prior to decryption, therefore, encrypted data can be securely saved or transmitted.
- Ensures the privacy of user-data, and the integrity of servers and their clients.
- Couchbase Server provides extensive support for data encryption and decryption.
- Multiple areas of the system are affected: therefore, essential information is distributed throughout the documentation set.

Encryption on the Wire

- Allows data to pass in encrypted form between nodes, between clusters, and between a cluster and its clients.
- **Node-to-Node Encryption**. Network traffic between the individual nodes of a Couchbase-Server cluster can be encrypted, in order to optimize cluster-internal security.
- **TLS Configuration**. To support secure communications between nodes, clusters, and clients, Couchbase Server provides interfaces for the configuration of *TLS* and supportive *cipher-suites*.
- **Secure Console Access**. Administrators can connect securely to Couchbase Web Console. Non-secure access can be disabled, for extra security.
- **X.509 Certificates**. These support encrypted communications between nodes, between clusters, and between a cluster and its clients.
- **Secure Ports**. Services are available on secure ports.

Encryption at Rest

- Encryption at Rest (meaning, on disk or other storage-device) allows passwords and data in files and directories to be encrypted.
- Data in Files and Directories. Programs are available for the encryption of data in files and directories.
- System Secrets. Passwords, certificates, and other items essential to Couchbase-Server security can be written to disk in encrypted format.

Encryption in Applications

- Field Level Encryption. This allows fields within a document to be securely encrypted by the SDK, to support FIPS-140-2 compliance.
- Field Level Encryption from the Java SDK. Provides directions for configuring encrypted field-level communication with Couchbase Server.