باسمه‌تعالی

آسیب‌پذیری عدم احراز اصالت در MongoDB و نحوه فعال‌سازی آن

مقدمه

MongoDB یک دیتابیس متن‌باز و رایگان و از نوع دیتابیس‌های NoSQL می‌باشد. در این‌گونه دیتابیس‌ها از ساختار سنتی دیتابیس‌های معمول که بر اساس جداول می‌باشند، استفاده نمی‌شود بلکه اطلاعات به شکل اسناد (documents) و به صورت یک زوج از رشته و مقدار (field & value) مانند JSON ذخیره می‌شوند. در مثال زیر، یک نمونه از اطلاعات ذخیره شده در دیتابیس MongoDB نشان داده شده است که طبق الگوی field:value مقداردهی شده‌اند:

```
{
name: “sue”,
age: 26,
status: “A”,
groups: [ “news”, “sports” ]
}
```

یکی از آسیب‌پذیری‌های رایج این دیتابیس، پیکربندی نامناسب و درنتیجه دسترسی همگان به آن از طریق شبکه اینترنت است. برای رفع این آسیب‌پذیری باید از دسترسی کاربران و برنامه‌های مجاز به تنها آن دسته از اطلاعاتی که مورد نیاز آن‌ها می‌باشد، اطمینان حاصل کرد.

یکی از مهمترین موارد امنیتی که باید درتنظیمات MongoDB لحاظ شود، عملیات احراز اصالت می‌باشد. قبل از دسترسی به سیستم، تمامی کلاینت‌ها باید توسط MongoDB احراز اصالت شوند. با این کار فقط کاربران مجاز می‌توانند به اطلاعات موجود در MongoDB دسترسی داشته باشند.

در بسیاری از دیتابیس‌های مبتنی بر MongoDB با وجود مکانیزم احراز اصالت، این مورد مهم فعال نمی‌باشد. در نتیجه می‌توان به راحتی به این دیتابیس متصل شد و اطلاعات مربوط به آن را استخراج کرد.

MongoDB از مکانیزم‌های احراز اصالت زیر پشتیبانی می‌کند:

- challenge and response mechanism (MONGODB-CR)
- x509 certificate authentication
- LDAP proxy authentication
- Kerberos authentication

البته مکانیزم LDAP فقط هنگامی که MongoDB بر روی سیستم‌عامل لینوکس نصب شده باشد، قابل استفاده می‌باشد و اگر MongoDB بر روی ویندوز نصب شده باشد، نمی‌توان از آن استفاده کرد.

در ادامه نحوه فعال کردن مکانیزم احراز اصالت برای MongoDB پس از ساخت یک کاربر به عنوان administrator بیان شده است. در این حالت ابتدا یک کاربر به عنوان administrator ساخته می‌شود و پس از آن مکانیزم احراز اصالت فعال می‌شود. بعد از این مرحله می‌توان با کاربر administrator به MondoDB احراز اصالت شد و کاربران جدید با دسترسی‌های مشخص تعریف کرد.

این روش زمانی مفید است که ساخت اولین کاربر بر روی MongoDB نیازی به احراز اصالت قبل از reset کردن MongoDB نداشته باشد. روش انجام این کار بدین صورت است:

۱. راه‌اندازی MongoDB بدون احراز اصالت

```
mongod --port 27017 --dbpath /data/db1
```

۲. ساخت کاربر با سطح دسترسی مدیر: در مثال زیر کاربر siteUserAdmin در دیتابیس admin ساخته می‌شود:

```
use admin
db.createUser(
  {
    user: "siteUserAdmin",
    pwd: "password",
    roles: [ { role: "userAdminAnyDatabase", db: "admin" } ]
  }
)
```

۳. شروع مجدد MongoDB ضمن فعال بودن مکانیزم احراز اصالت: در مثال زیر مکانیزم احراز اصالت با استفاده از تنظیمات دستوری authorization فعال شده است:

```
mongod --auth --config /etc/mongodb/mongodb.conf
```

۴. ساخت حساب‌های کاربری دیگر: در این مرحله می‌توان با کاربر مدیر وارد و احراز اصالت شد و دیگر کاربران را تعریف کرد.

برای وارد شدن به محیط دستوری MongoDB از راه دور، می‌توان از دستور زیر استفاده کرد:

```
$ mongo ip_address_of_mongo_server
```

اگر هنگام وارد شدن به محیط دستوری MongoDB هیچ رمز عبوری درخواست نشد، می‌توان نتیجه گرفت که مکانیزم احراز اصالت روی آن فعال نشده است. MongoDB به صورت پیش فرض بر روی پورت ۲۷۰۱۷ فعال می‌باشد و از دیتابیس test استفاده می‌کند.

برای مشاهده دستورالعمل‌های فعال‌سازی دیگر مکانیزم‌های احراز اصالت، می‌توان از لینک‌های زیر استفاده کرد:

- x509 certificate authentication
  https://docs.mongodb.com/v2.6/tutorial/configure-x509-client-authentication/

- LDAP proxy authentication
  https://docs.mongodb.com/v2.6/tutorial/configure-ldap-sasl-activedirectory/
  https://docs.mongodb.com/v2.6/tutorial/configure-ldap-sasl-openldap/

- Kerberos authentication
  https://docs.mongodb.com/v2.6/tutorial/control-access-to-mongodb-with-kerberos-authentication/
  https://docs.mongodb.com/v2.6/tutorial/control-access-to-mongodb-windows-with-kerberos-authentication/