DATA STORAGE FOR ANDROID (CouchDB) =>

**OVERVIEW**

1. Explain how the storage mechanism works in CouchDB?

What does CouchDB do?

CouchDB is an open source NoSQL database. It is used in many devices for better Web Accessibility and Compatibility. It is useful when you have a big data or a large amount of distributed data. Documents are the primary unit of operation in introduced in CouchDB. And, each Document is having a unique identity in the database. You can perform Database activities by using your browser. CouchDB provides RESTful HTTP API for this purpose. The records in the database are stored as key-value pair.

**What could go wrong?**

CouchDB allows every user to perform every operation, by default. It will not ask for privileges. Normally, CouchDB will listen on loopback interface that is 127.0.0.1 or localhost. But, if you expose CouchDB to outside world, your database might be under risk. Hence, we recommend defining admin user and then performing database operations. Same should be taken care while exposing CouchDB to public via RESTful HTTP API.

We recommend using TLS 1.2 minimum level of encryption for the purpose of communication with the outer world. As of now, CouchDB 1.1.0 uses built in SSL encryption for secure communication purpose.

Make sure that you are communicating over HTTPS protocol, while communicating to outside public. Passwords are sent to CouchDB in plain text. But in production environment, we must use password encryption.

CouchDB stores passwords as hashes, here we can take a look if a stronger hashing algorithm like SHA-2 and weaker hashing algorithms like MD5 should not be used. This we can verify in .ini files with config settings.

CouchDB uses cookie/token authentication. For the first time use, to obtain a token, you have to authenticate by using username and password. After successful login, you will receive a token, that is used by CouchDB to identify your session. This token is active for 10 minutes. We recommend to adjust the token lifetime parameter in ‘couch\_httpd\_auth’ section. Persistent storage of cookies should be disabled by using – ‘allow\_persistent\_cookies = false’. Likewise, to make sure that authenticated users only access the database, you can add - require\_valid\_user = true. This will not allow any anonymous requests to access the database.

 **(There are many parameter configurations; can I add them together in another heading?)**

**Static and Dynamic analysis part is yet to be added.**