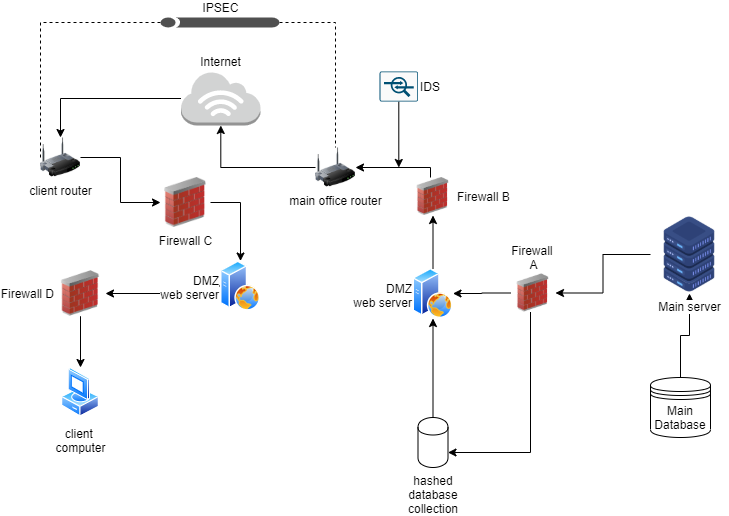
The Sex offender’s background check conformation system.



The system design explanation below is based on the diagram above, starting from the server end (right):

The main database that contain raw information (DOB/Name in plain text) of the sex offenders can only be accessed through the main server. The main server can restrict SQL commands sent to the database by only allowing INSERT and SELECT. This, along with regular backups/ disk level encryption and physical safety measures (fireproof casing, flood proofing), will provide Data integrity and limited access to the database (Appending data only). Every time the main server appends the main database, it can hash, by using SHA256, a concatenated String of offender’s DOB and name; and store that hash, along with an auto generated id and a random salt, on the hashed database collection. A firewall (firewall A) rule can be implemented to only allow communication between the main server and the hashed database collection. that will update the Hashed data. The DMZ webserver will intersect all communication to and from the internal network, as well as interact with the hashed database collection (queries only). This will hide the internal network from the internet while providing an extra layer of security to the local network. To avoid insider attacks, the main server and main database can exist on the same physical location, which can have limited access only by authorized users. Since the data sent from the main server to the hashed database collection is just hashed data (apart from the id and salt), That data will not be useful if an attack occurs. The DMZ server will be responsible for communicating the clients. The DMZ server will authenticate the user and generate A symmetric key and exchange it with the client by using the Diffie Hellman key exchange. After this interaction, the client can then send a number, indicating the amount of records contained in the client database. This can be used to determine if it is necessary to update the client side. If there are new records to send, the web server can then generate a HMAC to each hashed entry by using the symmetric key sent to that specific client + hashed BOD/name + salt. Then the hash values, along with their respective HMAC, can be sent to the client. The HMAC will ensure the authenticity and integrity of the data, while the separate symmetric keys per each client/server connection will add an extra layer of complexity for data integrity. The salt added will greatly reduce the chances of a successful dictionary attack. Dictionary attacks can also be reduced by monitoring the network with an IDS. This can send alerts any administrators if a high number of requests comes into the network. The network traffic can also be scheduled, since the client’s commutations are to update their databases, the requests can be timed, maybe early in the morning, for each client. That can aid in monitoring unusual activity.