Zen Analytics LPR Architecture Document

Table of Content

1. Servers Distribution

a. Zen Services Overview

All Services can be installed and configured separately on different devices on the same network or on a single device. Server Requirements may change according to the size of project, number of cameras, modules installed, and type of module configuration required in a project.

i. Zen Management Service

Zen Analytics Management Service is a Windows service and a Secure Web API service responsible for handling all the main configuration management of the system, initializing **Zen Analytics Service** configuration, and ensuring data distribution to Management Clients.

The following Service will:

- Configure Server Parameters.
- Configure and assign available Milestone XProtect Cameras.
- Configure Analysis Modules (License Plate Recognition Module, …).
- Configure Post-Process Actions (Watchlist, API Integration, Input/Output Triggers).
- Map the Character, State, Type, and Color results to customized values.
- Authorize and initialize Milestone XProtect Connection.
- Initialize and Prepare **Zen Live Broadcaster Service**.
- Distribute Module data results to Smart Clients.

ii. Zen Analytics Service

Zen Analytics Service is a standalone Windows Service processing network application responsible for requesting images from Milestone XProtect through the VideoOS MIP SDK Streaming module (can also utilize the Mobile API Streaming SDK), and processing the images based on the configuration and calibration done using the **Zen Management Client** which then saves the results inside the database and broadcasts the results using **Zen Broadcasting Service**.

To ensure the highest possible accurate results and detections without the need of any additional hardware requirement (Inductive loop, Lidar, Laser beams, ...), our platform is using its internal AI (Artificial Intelligence) module for Object Detections to detect vehicles/License plates to send an easy and simple data to the OCR engine to extract characters and do its job.

According to the configured modules' processing types and parameters, this service must be installed on a hardware with a NVIDIA GPU (its specifications are provided per Project) to handle the advanced AI calculations to ensure accurate and fast results.

Based on the project requirements per site, NVIDIA GPU is mandatory for the above tasks where its size will be calculated depending on the processing load needed per site based on the traffic, number of cameras, and any additional criteria which request any processing from the internal AI engine.

To prevent data-loss in case of network communication failure with the **Management Service**, and to make sure that no license plate is left without being communicated to the CCR (central control room – usually installed in the HQ), the system uses a local database and a special service to guarantee the transactions. This database can also be used locally if needed.

iii. Zen Analytics Broadcasting Service

Zen Analytics Broadcasting Service is a standalone Windows Service delivery network application responsible for Live data delivery and client coordination.

This service will also recover lost client connection and ensure re-connection establishment in case of disconnection or network loss, guaranteeing a stable Live Data Stream. Can be configured to cache results for redistribution to ensure Live delivery if needed (Currently only using in-memory caching).

b. Dependencies

i. Milestone XProtect

- #### SDKs

  The System uses latest Milestone XProtect SDKs to ensure Milestone XProtect configuration, camera selection and Input/Output for event Triggering and Input Device Listening, in addition for using recorded Images for multiple view from different cameras on single camera detection.

  Currently using Milestone XProtect VideoOS-SDK_2020R2

- #### Streaming

  Direct Streaming from Milestone XProtect VMS which allows to use all properties available in Milestone XProtect VMS Management without the need for restarting **Zen Analytics Services**.

  Utilizes Milestone adaptive streaming feature for easy resolution and result quality enchantments.

- #### Plugins

  Zen Analytics Management Client and **Zen Analytics Smart Client** are plugins built using the Milestone XProtect SDK Plugin Services which allows an immersive usage and design inside the Milestone XProtect Management Client and Smart Client and uses latest Controls for easy use and fast configuration for the Management and quick and clear result display and live results for Smart Client.

  Both plugins are following all Milestone XProtect guidelines which allow the user access management to enable or disable the views base on Milestone XProtect plugin user access control.

- #### Alarm Manager and Event Server

  Zen Analytics Service uses the Milestone XProtect Event Server to send Alarms/Events for configured rules in **Zen Management Client** and triggered by **Zen Analytics Service**.

  The Following allows for:

  - Configuring Milestone XProtect Alarm Definitions for custom actions on events send by the system.
  - View Alarms in the Milestone XProtect Management Client and apply all the functions provided.
  - Connect External Devices to be controlled via events (Open gates, sound alarms, …)

ii. Database (Data Storage)

- **MongoDb**

Zen Analytics Services uses mainly the MongoDb Database Service to handle the data storage.

MongoDB is NoSQL database management program. NoSQL is used as an alternative to traditional relational databases. NoSQL databases are quite useful for working with large sets of distributed data.

The NoSQL DBMS uses a single master architecture for data consistency, with secondary databases that maintain copies of the primary database. Operations are automatically replicated to those secondary databases for automatic failover. (Check Failover Section for more Information)

2. Communication

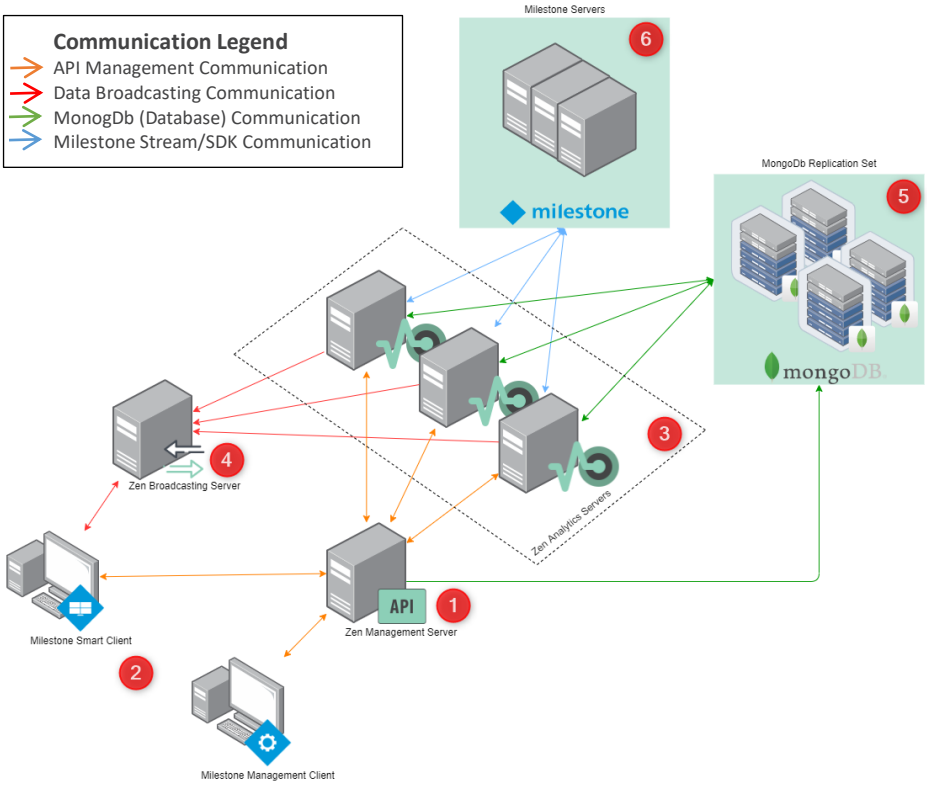

Figure 1 - Servers Diagram

a. 1 Zen Management Service - 2 Milestone Plugins Communication

Default HTTP/HTTPS API Communication. Communication can be modified to handle two way public/private authorization or token wise authorization.

b. 1 Zen Management Service - 5 MongoDb Driver Communication

Custom built Integration for NoSql connectors on top of MongoDb Driver communication to ensure fast and quick access high availability configuration.

c. 2 Milestone Plugins - 4 Zen Broadcaster Service Communication

Live Communication

Custom built socket implementation to ensure disconnections handling, loss of data, network delays and multiple user delivery for live data.

The plugin is considered as a data consumer client that receives the results and displays them using a clear live display.

On Demand Results
Default HTTP/HTTPS API Communication, which can be modified to handle two way public/private authorization or token wise authorization.

d. 4 Zen Broadcaster Service - 3 Zen Analytics Service Communication

Live Communication
Custom built socket implementation to ensure disconnections handling, loss of data, network delays and multiple user delivery for live data.

The analytics Service is considered as data producer where it sends all detected, modified then mapped data to the registered consumer clients for live instant display.

e. 3 Zen Analytics Service - 5 MongoDb Driver Communication

Custom built integration for NoSql connectors on top of MongoDb Driver communication to ensure fast and quick access high availability configuration.

f. 3 Zen Analytics Service - 6 Milestone Connection Communication

The communication between the analytics and Milestone XProtect servers are using the Milestone XProtect VideoOS SDK components which helps in instant Milestone XProtect Image Live Stream, Recording Feed and Camera configuration.

3. Server Fail Over

1. MongoDb Replication Set

Mainly to allow the MongoDb Replication and take out the single point failure problem minimum of 3 MongoDb Servers must be configured on separate devices in case replication is needed.

A replica set is a group of MongoDb instances that maintain the same data set. A replica set contains several data bearing nodes and optionally one arbiter node of the data bearing nodes, one and only one member is deemed the primary node, while the other nodes are deemed secondary nodes.

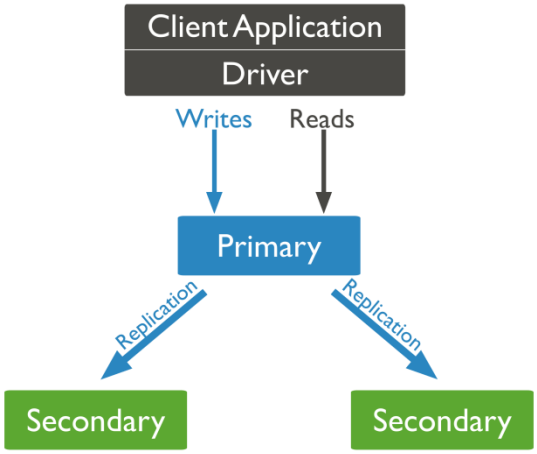

Figure 2 - Normal Server State

The secondaries replicate the primary's data and apply the operations to their data sets such that the secondaries' data sets reflect the primary's data set. If the primary is unavailable, an eligible secondary will hold an election to elect itself the new primary.

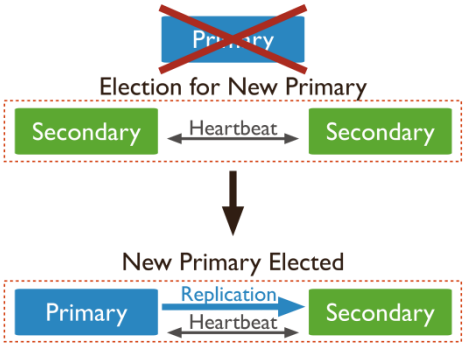

Figure 3 - One Failed MongoDb Node

When a primary does not communicate with the other members of the set for more than the configured period, an eligible secondary call for an election to nominate itself as the new primary. The cluster attempts to complete the election of a new primary and resume normal operations.

2. Windows Load Balancing and Data Clustering

Zen Analytics is compatible with Default Windows Clustering and Load Balancing for APIs and general Server Failure and Disaster Recovery.

a. Overview Explanation

Windows Clustering encompasses two different clustering technologies. These technologies implement the following two types of clusters.

A network load balancing cluster filters and distributes TCP/IP traffic across a range of nodes, regulating connection load according to administrator-defined port rules.
A failover cluster provides high availability for services, applications, and other resources through an architecture that maintains a consistent image of the cluster on all nodes and that allows nodes to transfer resource ownership on demand.
The following are the programming interfaces for the Windows Clustering technologies:

The Network Load Balancing Provider allows developers to create remote administration and configuration tools as well as customized user interfaces for Network Load Balancing clusters.
The Failover Cluster APIs allow developers to create cluster-aware applications, implement high availability for new types of resources, and create remote administration and configuration tools.

b. Simple Representation

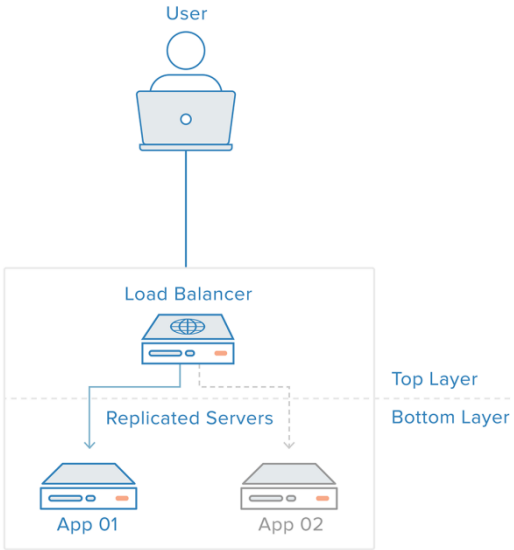

Figure 4 - Load Balancing

Any of the available severs covered in (Section 1) can be applied in the following load balancing and failover infrastructure where in (Figure 4) any of the 3 Main **Zen Analytics Services** Can be duplicated as (App01) and (App 02) with a custom Load-balancer Server that will handle any load on the servers and in case of any failure to keep the communication and management up and ready with no data loss and no missed access control if any is configured.

4. Zen Analytics LRR Recognition Process Flow

a. Overview

The following is the life cycle process in which the **Zen Analytics Service** follows on each configured camera to verify accurate License Plate Recognition to extract the License Plate Text, Color, State, Type, Confidence and Plate Image crop.

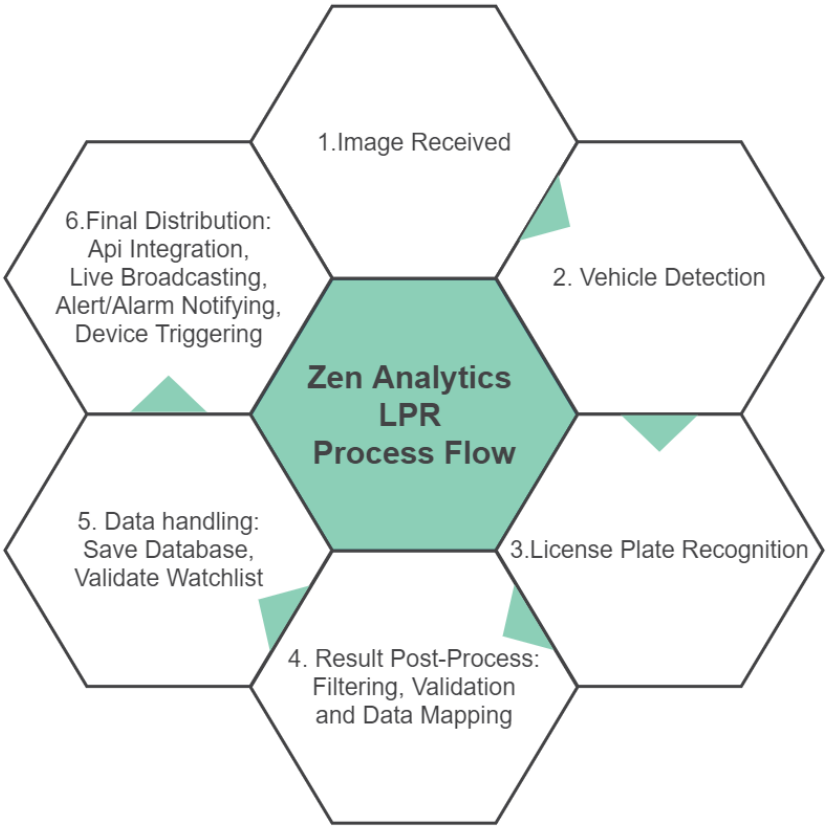

Figure 5 - LPR Process Flow

1. Image Stream

The First step in this process where the Image is received form Milestone XProtect as bytes then converted to necessary format in which **Zen Analytics Service** needs. Then based on the configured Image parameters, the system masks, resizes and warps the image under the optimal algorithms for the module itself (License Plate Recognition needs special Skewing, Rotating, Image Quality Enhancement)

2. Vehicle Detection

This step is an optional step, which mandates the availability of a GPU on the Server. This step increases the recognition accuracy, and system performance allowing further enhancement to the processing speed and Plate Recognition quality and reducing the workload on the system.

The following uses advanced AI software built in-house and can be modified and updated accordingly.

3. License Plate Recognition Engine

The License Plate Recognition is the vital step in which the system detects all details of the License Plate. This is done by passing the processed image to the ARH Carmen LPR Engine.

A Calibration stage is needed for ARH Engine so that the results recognized are up to the required standards.

This part requires the ARH Dongle to be connected physically for the system to recognize plates and must be supplied with proper Country Engine.

4. Result Post Processing

This step in the process is considered one of the most important steps in the LPR analysis process, at this point the system will start refining the results and according to the mapped data and the configured parameters in addition to post processing algorithms for remodeling and upgrading the results recognized by the previous LPR engine step.

At this stage, the following is done:

- Color, State and Type Map where the values from the LPR Engine are changed from engine values, encoded number), to a more readable and understandable values (ex: 3533 => 'Red').
- Based on the configured engine country, plates are recognized, and data/characters will be filtered out, random spaces are trimmed.
- Plate Images are enhanced, cropped and wrap/skew corrected.

5. Data Handling

Once the data is transformed, this stage the results are saved along with the date of the detection and the camera where the plate is captured to be viewed later and filtered in charts, history grids, and used in report engines.

In addition to the database saving, Watchlist cross-matching occurs flagging it for the system to take the required actions, for example if the plate needs to be hidden (whitelisted) from the database or alerted or used for access control will be flagged for the next final stage.

6. Final distribution

The final stage in the process is distributing the results to the configured consumer clients in **Zen Analytics Smart Client** plugins and handle the Event/Alarm Manager requests followed by the Milestone XProtect device outputs triggers in previous stage.
In addition to that, the API integration post requests if any are configured.

Therefore, the following is achieved in this stage:

- Live Broadcasting Results for registered Smart Clients.
- Milestone XProtect Event/Alarm Notification for all watch listed results.
- Triggering Outputs Configured (Mostly for access control).
- API Integration Requests will be sent with recognized plates and images.

5. API Integration

a. General

i. Overview

The API ingratiation is simply a mapping of the recognized plates with its data and sent to the configured API using the specified authorization token and data content-types.

For more Information on how to configure the integration please refer to the "*Zen Analytics LPR Management Client User Guide*" Section 8. *External API Integration*

ii. Integration Details

Restful API

Zen Analytics API Integration is built over REST API Requests with both POST and GET Method Types.

RESTful APIs were designed to take advantage of existing protocols. While REST - or Representational State Transfer - can be used over nearly any protocol, when used for web APIs it typically takes advantage of HTTP.

One of the key advantages of REST APIs is that they provide a great deal of flexibility. Data is not tied to resources or methods, so REST can handle multiple types of calls, return different data formats, and even change structurally with the correct implementation of hypermedia. This flexibility allows developers to build an API that meets your needs while also meeting the needs of very diverse customers.

Compatible Security Authorization Methods

- **JWT Bearer**

A JSON Web Token (JWT) is an open standard (RFC 7519) that defines a compact and self-contained way for securely transmitting information between parties as a JSON object. This information can be verified and trusted because it is digitally signed. JWTs can be signed using a secret or a public/private key pair.

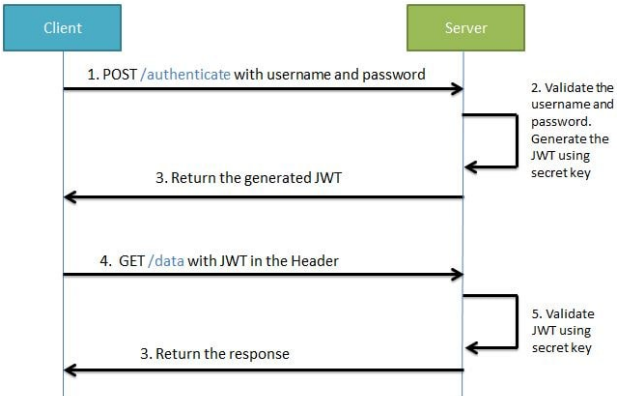

Figure 6 - Client Server JWT Authentication

- **Basic Authorization**

basic access authentication is a method for an HTTP Client to provide a user name and password when making a request.

Figure 7 - Client Server Basic Authorization

Responses Handling

Zen Analytics Integration can handle all default responses of a Restful API Interface and will reissue a new request for the data needed to be sent to ensure availability and no request data loss that may affect the Integration Integrity.

Those are the basic http Responses Covered:
HTTP response status codes indicate whether a specific HTTP request has been successfully completed. Responses are grouped in five classes:

- Informational responses (100–199)
- Successful responses (200–299)
- Redirects (300–399)
- Client errors (400–499)
- Server errors (500–599)

Refer to https://developer.mozilla.org/en-US/docs/Web/HTTP/Status for more Response details.

Data Sent from Zen Analytics
Zen Analytics will Send all mapped request body according to the *External Integration* configuration. The Data will be sent in a compact form called JSON.

JavaScript Object Notation (JSON) is a standard text-based format for representing structured data based on JavaScript object syntax. It is commonly used for transmitting data in web applications.

The following is one of the recognition Results from the system that can be mapped:

{

"Time":"637472884751160000",
"Image": byte[5651],
"PlateText": "Z105066",
"PlateCountry": "Lebanon",
"PlateState": "207052",
"PlateColor": "White",
"Confidance": 59,
"Datasource": “LPR Camera 1”,
"FullFrame":byte[5651],
"Recognized": true
}

6. Table of Figures