✅ 🔹 **Step 1: Understand the problem**

- IDS = Intrusion Detection System → detects malicious activity in network traffic.
- IDPS = IDS + Prevention → can also block traffic.
- Your AI part = use ML/DL to classify **benign vs malicious traffic**.
- Goal = Detect both **known** and **unknown (zero-day)** attacks by learning traffic patterns.

✅ 🔹 **Step 2: Learn the basics**

Before coding the project, you should be comfortable with:

1. **Python** basics.
2. **Pandas, Numpy** → handling datasets.
3. **scikit-learn** → for initial machine learning models.
4. (Optional later) **TensorFlow or PyTorch** → for deep learning models.

👉 You don't need TensorFlow/PyTorch from day 1. Start with scikit-learn (easier).

✅ 🔹 **Step 3: Get a dataset**

You can't train without real traffic data. The most popular IDS datasets are:

- **NSL-KDD** (classic benchmark, small, good for starting).
- **CICIDS2017 / CICIDS2018** (modern, contains many attack types like DDoS, Brute Force, Botnet).
- **UNSW-NB15** (also widely used).

👉 Start with **NSL-KDD** (easy, small). Later, try CICIDS2017.

✅ 🔹 **Step 4: Preprocess the dataset**

- Clean missing values.
- Convert categorical features (like protocol type) into numbers (label encoding or one-hot encoding).
- Normalize features (so ML learns faster).
- Split into **train (80%)** and **test (20%)**.

🔹 **Step 5: Train a simple model (baseline)**

- Use **RandomForest, Decision Tree, or Logistic Regression** in scikit-learn.
- Example: Train → Test → Print Accuracy, Precision, Recall, F1.
- This gives you a **baseline performance**.

🔹 **Step 6: Move to anomaly detection**

- Try **Isolation Forest** or **One-Class SVM** → these detect **new/unknown attacks** by finding anomalies.
- Compare with step 5 results.

🔹 **Step 7: Deep learning models (optional, later)**

- Use TensorFlow or PyTorch when you're ready.
- Try:
  - **Feedforward Neural Networks** → basic classification.
  - **Autoencoders** → anomaly detection.
  - **LSTMs** → detect sequential attack patterns in traffic flows.

🔹 **Step 8: Real-time simulation**

- Capture packets using **Wireshark / tcpdump** or Python libraries like **scapy**.
- Pass live features into your trained model.
- If malicious → log/alert/block (e.g., block IP using firewall rules).

🔹 **Step 9: Add Prevention (IDPS part)**

- When detection = malicious →
  - Auto-block IP using Python + firewall (iptables in Linux).
  - Or send alert to admin.

🔹 **Step 10: Document and improve**

- Record accuracy, false positives, false negatives.
- Compare ML vs DL models.
- Optimize preprocessing + feature engineering.