# Application Overview: Adaptive Network

## 1. Application Details

- **Name**: Adaptive Network
- **Platform**: Windows Service (deployed via Ivanti DSM)
- **Technology Stack**:
    - C# (.NET Framework)
    - SQL Server
    - LLDP packet analysis

## 2. Purpose

The purpose of the Adaptive Network application is to dynamically assign the appropriate VLAN and static IP address to production computers based on the physical location (factory hall) where they are connected, by detecting the VLAN configuration of the switch port they are plugged into.

## 3. Background

Due to infrastructure limitations (lack of RADIUS server or MAC-based VLAN assignment capabilities on the server side), we implemented a client-side VLAN adaptation mechanism. The logic is executed directly on the endpoint devices (production computers).

The factory consists of 4 production halls, each with its own dedicated VLAN: - Hall 1: VLAN 1021 - Hall 2: VLAN 1022 - Hall 3: VLAN 1023 - Hall 4: VLAN 1024

Non-production (office) devices use VLAN 1088 via DHCP and are not subject to this process.

## 4. Workflow – Step-by-Step Process

### 4.1. Deployment

When a production PC is installed and configured, Ivanti DSM pushes the Adaptive Network Windows Service to the device. The service runs silently in the background on system startup.

### 4.2. Interface Monitoring

The service continuously monitors the status of the Ethernet adapter. Once the adapter is detected as active (Up), the service triggers the VLAN detection procedure.

### 4.3. VLAN Detection

The service scans network traffic and filters LLDP (Link Layer Discovery Protocol) packets. It extracts the VLAN ID from the LLDP payload (e.g., VLAN 1021).

### 4.4. Temporary IP Assignment (Dummy Address)

Based on the detected VLAN ID, the service assigns a temporary static IP address in the format: 10.90..254.

Example: For VLAN 1021 → 10.90.21.254
This IP is known to be always free and is only used to establish initial network connectivity with the internal systems (e.g., SQL server).

### 4.5. Database Connection

Using the temporary IP address, the service establishes a connection to a central SQL database. Each VLAN/hall has a dedicated table within the database storing the available IP pool.

### 4.6. Request for Free IP Address

The service sends a query to fetch the first available IP address for the detected VLAN.

Example response: 10.90.21.60

### 4.7. Permanent IP Configuration

The service removes the temporary IP address. It sets the received permanent static IP (e.g., 10.90.21.60) on the same network interface.

### 4.8. Database Update

The new IP address is marked as assigned in the SQL database, preventing future reuse.

## 5. Automatic Relocation Handling

If a technician moves the computer to a different hall (e.g., from Hall 1 to Hall 2) without notifying the IT team:

The service detects that the adapter is connected but unable to communicate with the expected servers. This triggers a re-execution of the entire process, including: - VLAN detection via LLDP - Assignment of a new temporary IP (e.g., 10.90.22.254) - Fetching a new permanent IP from the correct VLAN pool (e.g., 10.90.22.73) - Applying the new IP - Updating the database

## 6. Additional Tool – Admin Console for IP Management

To support the system, a secondary C# .NET application was developed for IT administrators.

**Purpose of the Admin Tool**: - Manage and maintain the IP address database - Perform CRUD operations (Create, Read, Update, Delete) on records: - Add new devices - Edit device information or assigned IPs - Delete records when devices are retired

This tool ensures centralized control and transparency over all IP assignments across the production network, including production PCs, scanners, tablets, and other non-standard clients.

## 7. Benefits

- Automated VLAN detection based on physical switch connection
- No need for server-side MAC-based VLAN assignment (e.g., via RADIUS)
- Decentralized intelligence – runs at the client level
- Avoids IP conflicts through real-time database synchronization
- Easily scalable to other production halls or devices
- Fail-safe mechanism – detects relocation and reconfigures automatically