

10.3 Example Code

10.3.1 Root Node

```
#include <Arduino.h>
#include <painlessMesh.h>
#include <WiFi.h>

// --- Konfigurasi Jaringan ---
#define MESH_PREFIX      "jaringan_mesh_saya" // HARUS SAMA dengan semua node
#define MESH_PASSWORD    "password_mesh"      // HARUS SAMA dengan semua node
#define MESH_PORT        5555

painlessMesh mesh;

// Callback ketika Root menerima pesan dari Leaf
void receivedCallback(uint32_t from, String &msg) {
  Serial.printf("Pesan diterima dari node %u: %s\n", from, msg.c_str());
}

void setup() {
  Serial.begin(115200);

  // Inisialisasi Mesh
  mesh.setDebugMsgTypes(ERROR | STARTUP | CONNECTION);
  mesh.init(MESH_PREFIX, MESH_PASSWORD, MESH_PORT);

  // Set sebagai ROOT
  mesh.setRoot(true);
  mesh.setContainsRoot(true);

  // Pasang callback untuk menerima pesan
  mesh.onReceive(&receivedCallback);

  Serial.println("Mesh dimulai sebagai ROOT");
  Serial.print("Root ESP32 SoftAP IP: ");
  Serial.println(WiFi.softAPIP());
}
```

```
}

void loop() {
  mesh.update();
}
```

10.3.2 Leaf Node

```
#include <Arduino.h>
#include <painlessMesh.h>

// --- Konfigurasi Jaringan ---
#define MESH_PREFIX      "jaringan_mesh_saya" // HARUS SAMA dengan semua node
#define MESH_PASSWORD    "password_mesh"      // HARUS SAMA dengan semua node
#define MESH_PORT        5555

painlessMesh mesh;

// Ganti sesuai nomor leaf
String leafName = "Leaf1";

void sendMessage() {
  String msg = "Hello guys im " + leafName;
  mesh.sendBroadcast(msg);
  Serial.println("Sent: " + msg);
}

void setup() {
  Serial.begin(115200);
  mesh.setDebugMsgTypes(ERROR | STARTUP | CONNECTION);
  mesh.init(MESH_PREFIX, MESH_PASSWORD, MESH_PORT);
}

void loop() {
  mesh.update();
  static unsigned long lastSend = 0;
  if (millis() - lastSend > 5000) { // kirim tiap 5 detik
    lastSend = millis();
    sendMessage();
  }
}
```

```
}  
}
```

10.3.3 Intermediate Parent Node

```
#include <Arduino.h>  
#include <painlessMesh.h>  
  
// --- Konfigurasi Jaringan ---  
#define MESH_PREFIX      "jaringan_mesh_saya" // HARUS SAMA dengan semua node  
#define MESH_PASSWORD    "password_mesh"      // HARUS SAMA dengan semua node  
#define MESH_PORT        5555  
  
painlessMesh mesh;  
  
// Callback ketika node menerima pesan  
void receivedCallback(uint32_t from, String &msg) {  
    Serial.printf("Pesan diterima dari %u: %s\n", from, msg.c_str());  
    // Intermediate node tidak perlu kirim ke server, cukup teruskan pesan  
    mesh.sendBroadcast(msg);  
}  
  
// Callback saat node terhubung  
void newConnectionCallback(uint32_t nodeId) {  
    Serial.printf("Node baru terhubung: %u\n", nodeId);  
}  
  
// Callback saat node terputus  
void changedConnectionCallback() {  
    Serial.println("Perubahan koneksi terjadi!");  
}  
  
// Callback saat node time sync  
void nodeTimeAdjustedCallback(int32_t offset) {  
    Serial.printf("Waktu sinkronisasi, offset = %d\n", offset);  
}  
  
void setup() {  
    Serial.begin(115200);
```

```
// Inisialisasi mesh
mesh.setDebugMsgTypes(ERROR | STARTUP);
mesh.init(MESH_PREFIX, MESH_PASSWORD, &userScheduler, MESH_PORT);

// Callback
mesh.onReceive(&receivedCallback);
mesh.onNewConnection(&newConnectionCallback);
mesh.onChangedConnections(&changedConnectionCallback);
mesh.onNodeTimeAdjusted(&nodeTimeAdjustedCallback);
}

void loop() {
  mesh.update();
}
```

10.3.4 Idle Node

There is no code because this is just a status if the node is not yet connected to the mesh.

Revision #6

Created 2025-08-29 01:35:34 UTC by YP

Updated 2025-09-15 04:26:56 UTC by YP