UNIVERSIDADE FEDERAL DE SANTA CATARINA
CAMPUS ARARANGUÁ ou CENTRO CTS
DEPARTAMENTO DEC
CURSO ENGENHARIA DE COMPUTAÇÃO

Augusto Daleffe & João Victor Pavan

Auditor de Rede sem Fio

(Relatório Final)

Araranguá

06 de junho de 2025

SUMÁRIO

Araranguá

06 de junho de 2025

1. Visão Geral

O projeto tem como intuito desenvolver um auditor para redes sem fio, a fim de trazer à tona vulnerabilidades comuns dentro do meio. Para isso, serão utilizados ESP32, componentes responsáveis pela simulação de uma rede sem fio e seus usuários.

Durante a simulação serão apresentados ataques direcionados tanto ao ponto de acesso quanto aos participantes do meio, mostrando seu funcionamento e como é possível identifica-lo e neutraliza-lo

Araranguá

06 de junho de 2025

2. Descrição do projeto final

O projeto proposto tem enfoque no desenvolvimento de um auditor de redes sem fio Wi-Fi utilizando a versatilidade dos microcontroladores ESP32. A proposta central é desenvolver uma solução robusta para identificar deficiências e vulnerabilidades em redes, simulando cenários de ataques e analisando o desempenho para garantir a segurança e eficiência da infraestrutura.

O projeto se baseia em uma topologia de rede estrela, onde um Ponto de Acesso (AP), também construído sobre um ESP32, será o responsável por realizar a auditoria na rede. Paralelamente, serão desenvolvidos algoritmos de Inteligência Artificial (IA) e/ou algoritmos clássicos para a análise profunda dos dados coletados, com o objetivo de identificar padrões e anomalias que possam indicar falhas de segurança.

Um dos pilares do projeto é a simulação de ataques reais utilizando clientes maliciosos que serão adicionados propositalmente à rede. Pretende-se abordar alguns dos ataques mais conhecidos, como Spoofing, Ataque de Desautenticação, Flooding, DHCP Starvation, entre outros. Espera-se que ao replicar esses cenários seja possível efetuar uma análise de desempenho precisa e a validação da eficácia do algoritmo desenvolvido.

O segundo pilar deste projeto é a construção da rede Wi-Fi. Optamos por uma topologia de rede estrela, que é a mais comum em ambientes domésticos e pequenos escritórios. Nela, um único Ponto de Acesso (AP) central se conecta a 'n' clientes, simplificando a gestão e o acesso. Essa escolha permite a simulação de cenários mais próximos ao cotidiano de redes Wi-Fi, tornando os resultados obtidos mais relevantes para análises futuras.

Araranguá

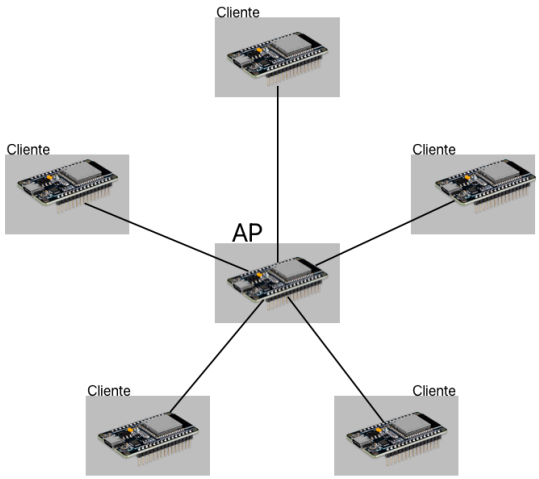
06 de junho de 2025

2.1. Componentes de Hardware

2.1.1. Microcontrolador ESP32

O ESP32 é um chip versátil que possui diversas capacidades, incluindo conectividade Wi-Fi e Bluetooth, processamento dual-core, memória RAM e Flash, além de uma variedade de interfaces de comunicação e pinos GPIO.

No projeto será aplicado a topologia de rede estrela, onde, um ESP32 será configurado como ponto de acesso, enquanto demais ESP32 atuarão como clientes, que irão se conectar ao ponto de acesso, que posteriormente, simulam atividades maliciosas.



2.2. Componentes de Software

A implementação do projeto será realizada utilizando o Espressif IoT Development Framework (ESP-IDF), que é o framework oficial de desenvolvimento para a série ESP32. Os seguintes componentes são a base para o desenvolvimento do projeto:

- **Sistema Operacional (FreeRTOS):** O ESP-IDF é construído sobre o FreeRTOS.

- **Driver Wi-Fi(esp_wifi):** Fundamental para configuração do wifi
- **LightWeight IP (LwIP):** Configuração do DHCP no ponto de acesso. Também usado para criar conexões TCP/UDP.
- **Eventos (esp event):** Sistema para despacho de eventos do sistema.

2.3. Funcionamento do projeto

O projeto consiste em um Access Point (AP) em ESP32 que detecta três tipos de ataques: Auth Flood, Deauth Flood e Packet Flood. O arquivo principal AP.c contém:

1 - Inicialização do AP:

Onde é configurado o ESP32 no modo Access Point e inicia o servidor TCP:

```
void app_main(void)
{
    wifi_init_ap();       // Configura e inicia o Wi-Fi no modo AP
    xTaskCreate(tcp_server_task, "tcp_server_task", 4096, NULL, 5, NULL);
    // Loop principal
    while(1) {
        show_ap_status();
        vTaskDelay(pdMS_TO_TICKS(30000));
    }
}
```

2- Detecção de Ataques:

Auth Flood: excessivas tentativas de autenticação.

Deauth Flood: desconexões em alta frequência.

Packet Flood: envio massivo de pacotes TCP em pouco tempo.

A lógica de detecção baseia-se em contadores por segundo e, ao exceder o limite, o AP bloqueia o MAC do atacante por meio de uma blacklist

3- Servidor TCP e Monitoramento

Recebe pacotes dos clientes, acumula estatísticas (contador de pacotes) e verifica se ultrapassou o limite:

```
if (detect_packet_flood(client_mac)) {
    add_to_blacklist(client_mac, 3); // 3 = PACKET_FLOOD
    return;
}
```

2.4 Metodologia Experimental

A metodologia adotada baseou-se majoritariamente em testes práticos utilizando uma rede composta por módulos ESP32. Um dos dispositivos foi configurado como ponto de acesso (AP) e auditor da rede, enquanto os demais atuaram como clientes, sendo alguns programados para simular comportamentos maliciosos. Os testes foram conduzidos diretamente nesse ambiente, sem o uso de ferramentas externas, com o objetivo de verificar se o sistema era capaz de identificar e reagir adequadamente aos ataques.

Todos os comportamentos esperados foram validados com base na observação dos logs gerados e no comportamento do AP em tempo real. Os testes serviram para confirmar na prática a efetividade das rotinas de detecção e resposta embutidas no auditor, garantindo que o funcionamento ocorresse conforme o planejado.

3. Resultados e Discussão

Durante os testes realizados, o ponto de acesso implementado no ESP32 demonstrou capacidade de identificar e bloquear corretamente os ataques simulados, incluindo Auth Flood, Deauth Flood e Packet Flood. O sistema de auditoria embutido foi capaz de registrar as tentativas maliciosas e reagir conforme o esperado, com intervenções pontuais e baseadas na frequência e no tipo dos pacotes recebidos.

Em execuções prolongadas, observou-se que o AP conseguiu manter a estabilidade da rede, filtrando os ataques sem impactar significativamente os dispositivos legítimos conectados. Houve situações em que picos de tráfego exigiram reações mais agressivas do sistema, mas, de maneira geral, o comportamento permaneceu dentro do previsto, evidenciando a viabilidade do modelo proposto.

No contexto da disciplina, os resultados demonstram que é possível desenvolver mecanismos básicos de defesa para redes sem fio utilizando plataformas de baixo custo e recursos limitados, como o ESP32. Isso reforça a aplicabilidade prática de soluções embarcadas em segurança de redes, mesmo em ambientes com restrições severas de processamento e memória.

Araranguá

06 de junho de 2025

4. Desafios e Aprendizados

Durante o desenvolvimento do projeto, algumas limitações técnicas se destacaram, especialmente na tentativa de implementar ataques mais complexos, como ARP Spoofing e Evil Twin. Embora conceitualmente viáveis, esses tipos de ataque exigem maior controle sobre o hardware wireless e dependem de ajustes específicos de firmware ou do uso de bibliotecas mais avançadas, que extrapolam os recursos nativos oferecidos pelo ESP32 no ambiente de desenvolvimento utilizado.

Apesar dessas dificuldades, o projeto proporcionou um aprendizado significativo em relação à segurança de redes sem fio. Foi possível compreender e aplicar conceitos práticos de detecção de intrusão, como o monitoramento contínuo de pacotes, a implementação de contadores de eventos suspeitos e a definição de critérios para bloqueio de dispositivos com base em limites de frequência. O trabalho também reforçou a importância do equilíbrio entre reatividade e tolerância em sistemas de segurança embarcados, especialmente quando operando com recursos limitados.

Araranguá

06 de junho de 2025