Simulação de Ataques DoS e Replay com PIC16F877A

Clara Marcela Grossl

Universidade Federal de Santa Catarina

Contexto: A Fragilidade do Protocolo CAN

O Problema de Segurança

- **Ausência de Criptografia:** As mensagens trafegam em texto plano pelo barramento.
- **Sem Autenticação:** Qualquer dispositivo conectado pode se passar por um sensor crítico (ex: freio, motor).
- **Superfície de Ataque Ampliada:** Portas OBD-II e Centrais Multimídia modernas são vetores de entrada para invasores.

Objetivo do Trabalho

Desenvolver um sistema embarcado para demonstrar, na prática, como travar uma ECU via saturação de buffer e clonar chaves de acesso (Replay).

Arquitetura de Hardware (O Testbed)

Núcleo de Processamento:

- **Microcontroladores:** 2x PIC16F877A.
- **Clock:** Cristal de 20MHz
- **Protocolo:** UART Hardware emulando a camada física CAN.

Periféricos e Interface:

- **Entradas:** 5 Botões Digitais + 2 Potenciômetro (ADC).
- **Saídas:** 2 x Display LCD 20x4, Buzzer e LEDs de Status.
- **Armazenamento:** EEPROM interna para persistência de dados "roubados".

Vetor de Ataque 1: Denial of Service (DoS)

Estratégia: Saturação de Interrupção (ISR Starvation)
O dispositivo atacante inunda o barramento com tokens de 1 Byte em alta frequência. Isso força a ECU Vítima a entrar e sair da rotina de interrupção constantemente, impedindo a execução do *Main Loop*.

Comando	Tipo de Ataque	Consequência Prática
'A', 'B', 'C'	Stress Testing	Aumento de latência (Lag)
0x00	**Buffer Overflow**	**Travamento Total (Crash)**

Tabela 1: Mapeamento dos Comandos do Fuzzer

Vetor de Ataque 2: Sniffing e Replay

Conceito: Captura e retransmissão de sinais legítimos para execução não autorizada.

1. **Sniffing (Escuta Passiva):** O Atacante monitora o pino RX silenciosamente, aguardando tráfego.
2. **Interceptação e Filtro:** O algoritmo identifica um padrão específico.
3. **Persistência (EEPROM):** O código capturado é salvo na memória não volátil. Isso permite que o ataque seja executado mesmo se o dispositivo atacante for reiniciado.
4. **Replay (Injeção Ativa):** O atacante reproduz o sinal exato no barramento, validando a ação na Vítima.

Mecanismo de Defesa: Watchdog Timer (WDT)

Recuperação de Falhas em Sistemas de Tempo Real
Foi implementado um mecanismo de defesa via Hardware para mitigar o travamento por DoS.

Configuração Técnica do WDT

- **Prescaler:** 1:128.
- **Timeout Estimado:** $\approx$ 2.3 segundos (Baseado no ciclo nominal de 18ms).
- **Lógica de Atuação:**
  - *Normal:* O software limpa o WDT (‘CLRWDT‘) a cada ciclo do loop principal.
  - *Sob Ataque:* O loop trava devido às interrupções, o WDT "estoura"e força o pic a voltar as configurações normais.

Conclusão

O projeto demonstra que a segurança física veicular pode ser comprometida logicamente.

A utilização de microcontroladores de arquitetura simples (PIC) provou-se eficaz para simular cenários complexos de cibersegurança, servindo como ferramenta de baixo custo para auditoria e ensino de protocolos industriais.

Obrigado!

Dúvidas?

clara.grossl@grad.ufsc.br

UNIVERSIDADE FEDERAL
DE SANTA CATARINA