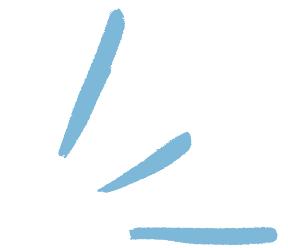




USP - UNIVERSIDADE DE  
SÃO PAULO - ICMC

# Projeto PUB

Thierry de Souza Araújo



# Tópicos

- Visão Geral
- Funcionamento da rede
- Código
- Serviço único
- Serviço duplo
- Considerações finais

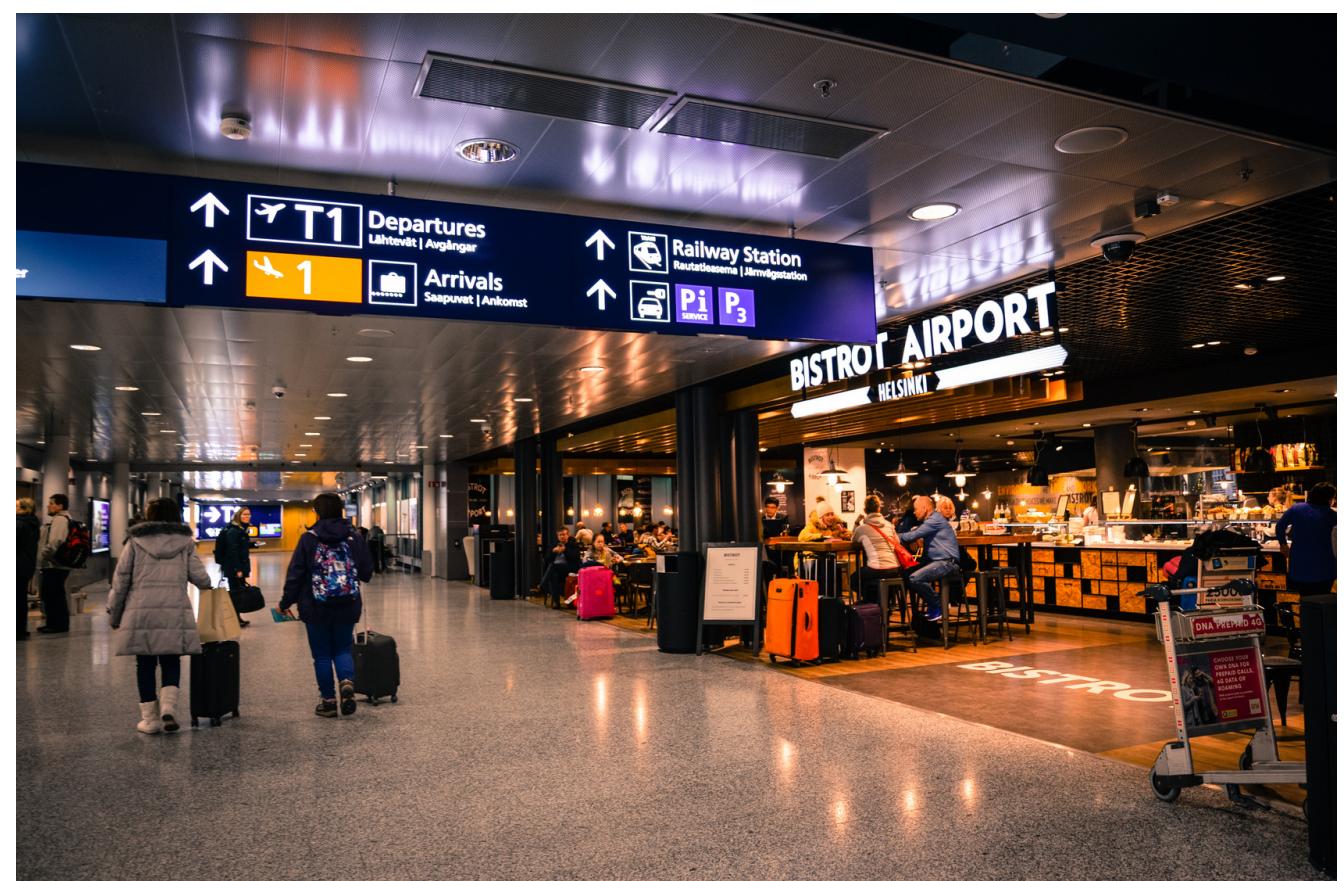
[https://github.com/ThierryAra/projeto\\_pub.git](https://github.com/ThierryAra/projeto_pub.git)

# Visão Geral

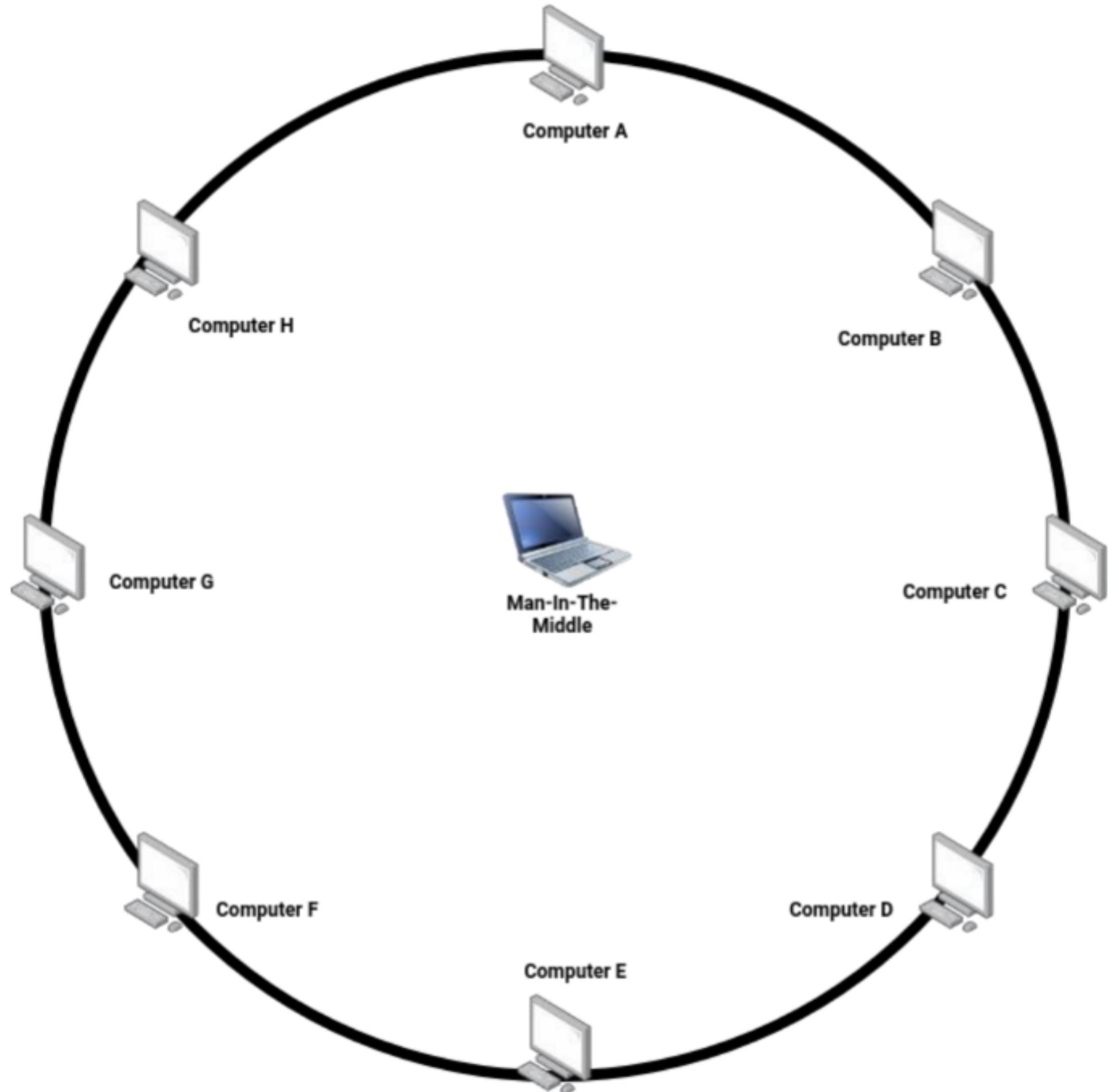
Um aeroporto internacional se comunica por uma rede de topologia em Anel bidirecional.

Os computadores são de acesso restrito dos funcionários e, além de se comunicarem por uma rede via cabo, também são livres para se comunicar por uma rede sem fio aberta. Alguns deles possuem funções específicas que auxiliam no funcionamento da rede.

Será produzido um código que simule a rede e apresente, de acordo com os serviços solicitados e o nó, o melhor caminho considerando o desempenho e segurança.



# Funcionamento da rede



Os computadores A, C, E e G atuam como servidores e possuem as seguintes funções:

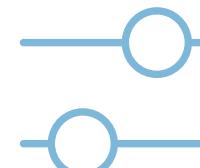
- **A** - Autenticação dos usuários
- **C** - Criptografia e descriptografia dos dados
- **E** - Monitoramento da velocidade da rede
- **G** - Responsável pelo compartilhamento de arquivos

Os **demais** atuam apenas como replicadores dos dados que trafegam na rede.

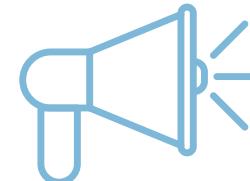
# Pontos assumidos



O custo de envio pela rede aberta foi escolhido arbitrariamente para cada serviço



Os serviços dos nós C e G necessitam de autenticação prévia.



Os servidores sempre retornam a solicitação com sucesso.



O custo de envio dos dados pela rede física se dá pela quantidade de saltos até o alvo

# 'Melhor caminho'

```
● ● ●

def best_path(self, service: str, target_info: list, source: str) -> str:
    path = ''

    path_bfs = self.bfs(source, target_info[0])
    target_info[1][1] = len(path_bfs) - 1 # Calculate cost to send by wire

    # Request
    # Wifi is cheaper (and don't need high security)
    if target_info[1][2] < target_info[1][1] and target_info[1][0] < 2:
        path = f'{source} -> Wifi -> {target_info[0]}'
    else:
        path = ' -> '.join(path_bfs)

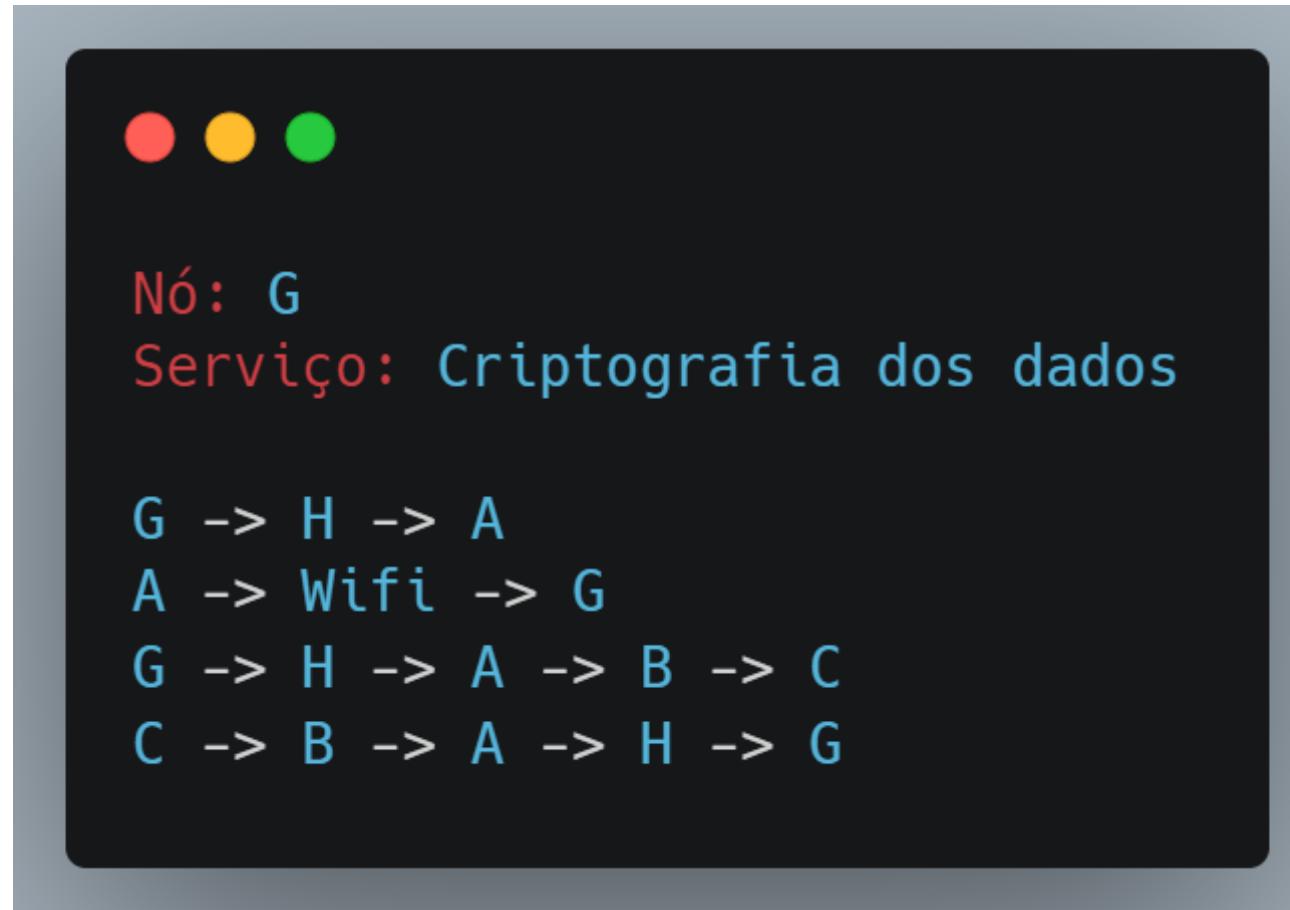
    # Response
    if target_info[1][2] < target_info[1][1] and service != 'Criptografia dos dados':
        path += f'\n{target_info[0]} -> Wifi -> {source}'
    else:
        if path_bfs:
            path_bfs.reverse()
            sec_path = path_bfs
        else:
            sec_path = self.bfs(target_info[0], source)

        path += '\n' + ' -> '.join(sec_path)

return path
```

- Para buscar a melhor direção de envio dos dados utilize uma BFS (Breadth First Search)
- target\_info é um array que possui o nível de segurança exigida pelo serviço (nível 2 necessita de autenticação prévia), o custo de envio pela rede cabeada e, por fim, o custo de envio pela rede sem fio.
- A resposta para o serviço de criptografia é garantido ser pela rede física pois pode ser uma descriptografia de dados e é necessário manter a segurança no retorno.

# 1 serviço (exemplo)



- O serviço 'Criptografia dos dados' exige que o usuário da rede esteja autenticado para garantir a segurança, pois ele também pode descriptografar algum dado.
- Os dados de autenticação são enviados pela rede (G -> H -> A) para aumentar a segurança no envio. A confirmação de autenticidade é enviada pela rede aberta pois apenas confirma que o usuário possui permissão, por isso prioriza o desempenho.
- Finalmente, a solicitação de criptografia ou descriptografia dos dados é enviada pela rede física, também com o objetivo de aumentar o grau de segurança

# RISCOS

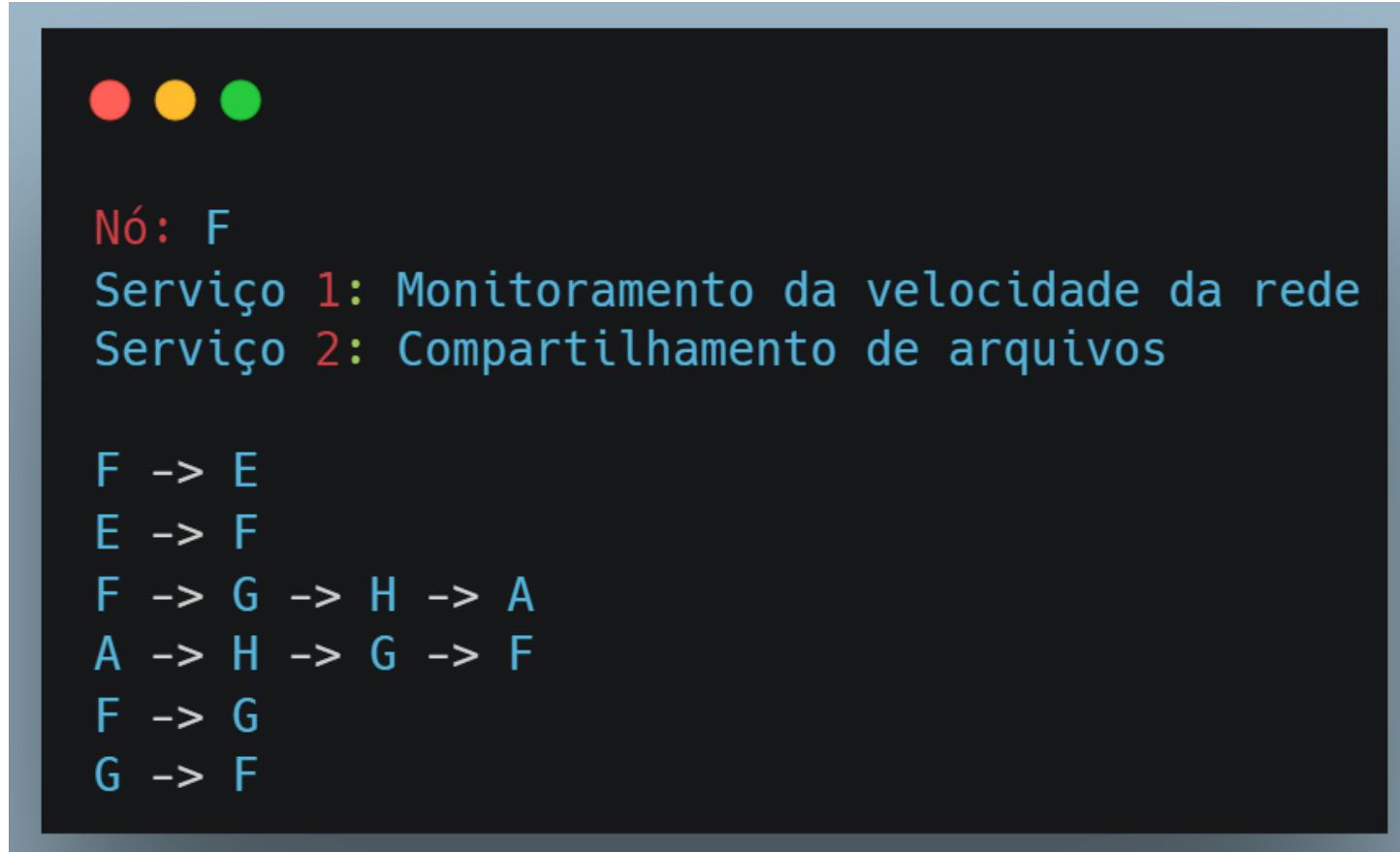


- Se a rede estiver sobrecarregada, o envio pode levar um tempo considerável.
- Há uma exposição dos dados aos nós A e H.
  - Aumenta o potencial de interceptação e
  - Aumenta a possibilidade de alteração dos dados.

# ALTERNATIVAS

- O caminho poderia se alterar se o serviço tivesse a função específica de realizar a criptografia de algum dado:
  - **Envio** pela rede
  - **Resposta** pela rede aberta (dados já seguros)
- Se a criptografia feita pela rede aberta fosse segura o suficiente, o envio poderia ser feito por ela.

# 2 serviços (exemplo)



→ O serviço de monitoramento não exige uma conexão segura, pois os dados enviados são simples e não necessitam de nenhuma confiabilidade. Por isso, a escolha do melhor caminho se dá pelo custo dos meios. Nesse caso, o custo do envio pela rede aberta para esse serviço era 5, portanto o envio foi realizado pela rede cabeada (custo 1).

→ Já, o serviço de compartilhamento de arquivos necessita de uma autenticação prévia, realizada com requisição e resposta via cabo, pois o custo era o mesmo para o envio pela rede aberta (4).

→ Por fim, para a conclusão do serviço é realizada a comunicação do nó F com o nó G pela rede física, garantindo uma melhor performance em relação ao wifi.

# RISCOS

- O serviço de compartilhamento pode ser realizado pela rede aberta, por isso é necessário que os arquivos enviados/baixados estejam criptografados.



# ALTERNATIVAS

- O caminho poderia se alterar se o serviço tivesse a função específica de realizar a criptografia de algum dado:
  - **Envio** pela rede
  - **Resposta** pela rede aberta (dados já seguros)
- Se a criptografia feita pela rede aberta fosse segura o suficiente, o envio poderia ser feito por ela.

# Custos e Escalabilidade



Caso a rede aumenta os custos tendem a aumentar consideravelmente, principalmente em serviços que necessitam ser precedidos de uma autenticação.



Além disso, os dados iriam precisar percorrer caminhos maiores, aumentando o tráfego na rede. Uma possível solução para isso, é a disponibilização de uma rede sem fio privada, incluindo algoritmos de segurança eficientes.



Considerando os serviços que não necessitam de um grau de segurança elevado, a comunicação pode ser focada na rede sem fio, para amenizar o tráfego na rede física.