Um sistema de rastreamento de contatos com foco em anonimidade

Mikaella F. da Silva, Vinícius F. S. Mota

[1]Departamento de Informática
Universidade Federal do Espírito Santo (UFES) – Vitória, ES – Brasil

mikaellaferreira0@gmail.com, vinicius.mota@inf.ufes.br

Abstract. *During the coronavirus pandemic, in addition to social isolation and the use of masks, the use of contact-tracking apps has shown great promise. However, these applications can cause privacy problems. This paper presents a contact tracking system that requires only a unique identifier, based on a signature key of the app and the user, while preserving anonymity. To track contacts, mobile devices act as beacons. Upon finding nearby devices, each identifier of the contact pair is sent to the cloud. If a user reports a positive diagnosis for COVID-19, a web service tracks the identifiers that have had contacts considered at risk. The system considers each identifier as a topic in an MQTT broker and broadcasts an alert message for each topic. The system was tested with twenty volunteers who used the application for two weeks. It was possible to analyze contacts with the potential to spread the virus and observe the users' social behavior, without them being able to be identified.*

Resumo. *Durante a pandemia do coronavírus, além do isolamento social e uso de máscaras, o uso de aplicativos de rastreamento de contatos se mostrou bastante promissor. No entanto, estes aplicativos podem causar problemas de privacidade. Este trabalho apresenta um sistema de rastreamento de contatos que requer apenas um identificador único, baseado em uma chave de assinatura do aplicativo e do usuário, preservando a anonimidade. Para rastrear os contatos, os dispositivos móveis atuam como beacons. Ao encontrar dispositivos próximos, cada identificador do par de contatos é enviado para a nuvem. Caso um usuário informe ter diagnóstico positivo para COVID-19, um serviço web rastreia os identificadores que tiveram contatos considerados de risco. O sistema considera cada identificador como um tópico em um broker MQTT e transmite uma mensagem de alerta para cada tópico. O sistema foi testado com vinte voluntários que utilizaram a aplicação por duas semanas. Foi possível analisar contatos com potencial de espalhamento do vírus e observar o comportamento social dos usuários, sem que eles pudessem ser identificados.*

1. Introdução

A recente pandemia causada pelo novo coronavírus revelou a necessidade de utilizar mecanismos eficientes para o controle da dispersão de doenças altamente transmissíveis. Além das medidas de isolamento e distanciamento social associado ao uso de máscaras, governos e outras entidades buscam recursos tecnológicos para auxiliar no controle da pandemia. Sistemas de rastreamento de contatos ganharam destaque nesse cenário. O

rastreamento de contatos é o processo de identificar, avaliar e gerenciar pessoas expostas a uma doença para evitar a transmissão posterior. Quando aplicado sistematicamente, o rastreamento de contatos quebra as cadeias de transmissão de uma doença infecciosa, portanto, é uma ferramenta essencial de saúde pública para controlar surtos de doenças infecciosas [World Health Organization 2020]. Essa medida, atrelada à realização de testes em massa e isolamento social, foi extremamente importante para a contenção do vírus na Coreia do Sul [Kang et al. 2021]. No entanto, a implementação desse tipo de sistema levanta preocupações sobre privacidade.

Sistemas de rastreamento de contato podem coletar dados considerados sensíveis, como nome, CPF, e-mail e localização. Além disso, o fato de uma organização registrar com quem cada indivíduo está tendo contato gera desconforto pela sensação de estarem sendo "vigiados". Outra preocupação é até que ponto os aplicativos podem ser reaproveitados para rastrear seus usuários e como os dados coletados podem ser usados após o término de uma pandemia [Ahmed et al. 2020].

Neste sentido, este trabalho apresenta um sistema de rastreamento de contato totalmente anônimo que preserva a privacidade dos usuários, tanto no contexto do servidor, quanto em relação aos usuários entre si. O sistema desenvolvido foca no rastreamento de contato que preserva a privacidade do usuário por meio da anonimidade no contexto do servidor e em relação aos indivíduos no qual se teve contato. Desta forma, não é possível identificar quem são os usuários infectados, além de não revelar nenhuma informação sensível dos usuários, para que se sintam confiantes em utilizar a aplicação e ainda seja possível realizar análises da rede de contatos e espalhamento do vírus.

O sistema é composto por um aplicativo móvel, um broker MQTT, para envio de mensagens por tópicos, e um conjunto de serviços web, que armazenam e gerenciam os pares de contatos, além de gerar alerta em caso de infecção. O aplicativo móvel é responsável por identificar pessoas próximas, enquanto serviços web armazenam e rastreiam os contatos de um usuário infectado para detectar pessoas em potencial risco e notificá-las. Uma estrutura de tópicos para mensagens *publish/subscribe* que utiliza apenas identificadores únicos criptografados dos contatos garante a privacidade. Para rastrear os contatos, o aplicativo utiliza *Bluetooth Beacon*. Ao receber um *beacon*, o aplicativo publica a data e hora, seu identificador e o identificador anunciado pelo via mensagem MQTT. Um serviço na nuvem armazena estes dados para processamento assíncrono. Quando uma pessoa é diagnosticada com COVID-19, os contatos de risco são os que ocorreram nos 15 dias anteriores por pelo menos 15 minutos e a uma distância inferior a 2 metros [World Health Organization 2020]. Ao informar diagnóstico positivo pelo aplicativo, um serviço web, chamado rastreador de contatos, filtra os os identificadores únicos dos contatos de risco desta pessoa. O sistema notifica usuários em risco de contágio, enviando mensagens para o tópico de identificador único do usuário.

A partir das informações armazenadas no banco de dados é possível analisar a evolução do grafo de contatos ao longo do tempo. Desta forma, este trabalho demonstra que o grafo de contatos para este tipo aplicação e os efeitos no espalhamento do vírus podem ser analisados de forma ponderada nas distâncias entre os usuários e no tempo de contato entre os mesmos, preservando a anonimidade dos usuários. O aplicativo foi utilizado por 20 usuários ao longo de duas semanas. Foram gerados 866 registros de contatos, dos quais 341 contatos foram superiores a 15 minutos.

O restante desse trabalho está organizado como se segue: A Seção 2 discute arquiteturas de sistemas de rastreamento de contato e os trabalhos relacionados. O sistema de rastreamento de contatos proposto é descrito na Seção 3. A Seção 4 apresenta a análise dos dados coletados pelo sistema proposto. Por fim, a Seção 6 apresenta as considerações finais e trabalhos futuros.

2. Trabalhos relacionados

Sistemas de rastreamento de contato, ou *contact tracing*, são responsáveis pelo processo de identificar, avaliar e gerenciar pessoas que foram expostas a uma doença para evitar a transmissão posterior [World Health Organization 2020]. A ideia é localizar pessoas que tiveram contato com alguém infectado, ou seja, pessoas com potencial de também estarem infectadas, e passar recomendações, como fazer testagem e isolamento social. A [World Health Organization 2020] orienta como definir um contato no contexto da covid-19, e dentre os citados, podemos destacar a pessoa que esteve menos de 1 metro de distância a um caso de covid-19 por mais de 15 minutos, onde o início da doença foi confirmado entre 2 a 14 dias atrás.

Há três tipos de arquiteturas comumente utilizadas em sistemas de rastreamento de contato, a centralizada, a descentralizada e a híbrida [Ahmed et al. 2020, Morio et al. 2023]. Na Arquitetura centralizada, o usuário deve registrar-se no servidor, provendo informações que possam ser utilizadas para identificá-los e contatá-los, caso necessário. O servidor fica responsável por gerar identificadores (ID) temporários para cada dispositivo. Na arquitetura descentralizada, o usuário é responsável por criar os IDs temporários e também identificar se teve contato com um infectado. O protocolo *Private Automated Contact Tracing* define que os dispositivos são responsáveis produzirem um pseudônimo, compartilhado com outros usuários no momento em que estiverem próximos [Rivest et al. 2020]. Por fim, a arquitetura híbrida propõe que a geração e o gerenciamento dos ID's temporários ficam sob responsabilidade dos dispositivos para garantir a privacidade e anonimidade, enquanto a análise de risco e notificação de usuários devem ser de responsabilidade do servidor centralizado.

O artigo [Lee et al. 2021] compara abordagens centralizadas e descentralizadas de protocolos de rastreamento de contatos e propõe um protocolo híbrido. Nesse protocolo, cada usuário gera localmente uma chave e envia para uma autoridade central. A autoridade central gera uma nova chave usando a chave do usuário e a chave da autoridade e a envia para o usuário, chamada de *re-key*. Os usuários computam *tokens* usando a chave do usuário e a *re-key* juntas e mantém localmente a lista dos *tokens* recebidos quando os usuários entram em contato. Se um usuário for diagnosticado positivo, o usuário envia a lista de *tokens* recebidos para o servidor, que publica para o público. O usuário que receber a lista, localmente deriva os seus *tokens* e checa se os *tokens* estão na lista que recebeu. Se conter um *token*, então o usuário entrou em contato com alguém infectado.

De maneira similar, o artigo [Bay et al. 2020] propõe o protocolo chamado *BlueTrace* e uma diferença é que a autoridade central fica responsável por identificar se o usuário teve contato com algum infectado. Basicamente, os usuários devem se registrar no sistema fornecendo o número de celular. O servidor gera um identificador único e aleatório e o associa ao número de telefone. Quando dois usuários se encontram, eles compartilham entre si identificadores temporários. Os identificadores alternam com

frequência para evitar que terceiros rastreiem os usuários, sendo gerados a partir do identificador do usuário, horário de criação e tempo de expiração, criptografados com AES-256-GCM e, em seguida, codificado em Base64. O histórico de encontro do usuário é armazenado localmente no dispositivo do usuário. Se um usuário estiver infectado ou sujeito a rastreamento de contato, ele será solicitado a compartilhar seu histórico de contato com a autoridade de saúde relevante com o uso de um PIN. Somente a autoridade de saúde consegue descriptografar o histórico de encontro e usar informações de identificação pessoal para contatar usuários potencialmente infectados.

[Ahmed et al. 2020] apresentam a diferença entre os protocolos *BlueTrace* e ROBERT (*ROBust and privacy-presERving proximity Tracing protocol*) [Castelluccia et al. 2020]. Enquanto o *BlueTrace* armazena informações pessoais (número de telefone), ROBERT armazena apenas identificadores anônimos referidos como *EphIDs*, fornecendo, então, um nível de privacidade. Os protocolos também diferem no processo de notificação. ROBERT requer que todos os usuários verifiquem frequentemente seus *EphIDs* usados com o servidor para determinar se eles são sinalizados como em risco. Em contraste, com o *BlueTrace*, as autoridades de saúde podem notificar proativamente os usuários em risco.

O sistema proposto neste trabalho retira a responsabilidade do usuário de identificar se o mesmo se encontra em risco, diferente de [Lee et al. 2021], já que é uma tarefa que requer recursos computacionais e pode expor dados de contatos de usuários, uma vez que todos os usuários fazem *download* dos *tokens* do contaminado. Além disso, o sistema não coleta informações pessoais, como faz o protocolo *BlueTrace*, visando a anonimidade, mas mesmo assim consegue notificar os usuários, sem que eles tenham que ficar verificando constantemente se estão em risco, como apresentado no protocolo ROBERT.

3. O sistema de rastreamento de contatos

3.1. Visão geral

O sistema é composto por um aplicativo móvel, um conjunto de serviços *web* e um broker MQTT. O aplicativo móvel é responsável por identificar pessoas próximas, enquanto os serviços *web* rastreiam os contatos de um usuário infectado para detectar pessoas em potencial risco e notificá-las. As mensagens de contato e de notificação são enviadas de maneira assíncrona via um broker MQTT, chamado EMQ. A Figura 1 apresenta uma visão geral da arquitetura do sistema proposto.

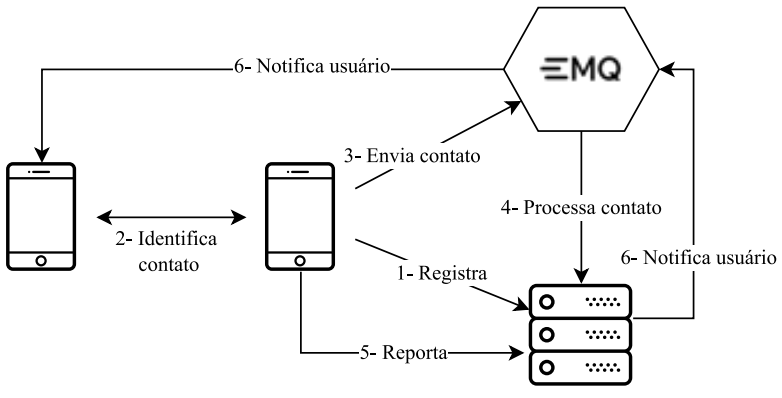


Figure 1. Visão geral do Sistema de Rastreamento de Contato.

Inicialmente, o usuário precisa **registrar** no sistema (1). No caso, utiliza-se um identificador do sistema para gerar um identificador único universal (UUID) do usuário. O aplicativo no dispositivo utiliza esse identificador para **identificar contatos** próximos (2). O dispositivo envia *beacons* com seu identificador, utilizando *Bluetooth Low-Energy* (BLE), periodicamente. Ao receber um *beacon*, é possível estimar a distância entre esses dispositivos baseando-se na intensidade do sinal recebido. A cada intervalo predefinido de tempo, os dispositivos **enviam o conjunto de dados** (3) contendo os contatos, por meio do protocolo MQTT, para armazenamento e processamento na nuvem (4). Caso um usuário **reporte** diagnóstico positivo para COVID-19 (5), um serviço web de rastreamento irá buscar no banco de dados todos os identificadores que tiveram contato com este usuário nos últimos 15 dias, que o contato tenha durado mais de 15 minutos e a distância entre eles seja menor que 2 metros, conforme recomendações da OMS. Este serviço irá **enviar uma notificação** para todos os usuários que se enquadrem na regra de contato de risco (6). Essa notificação é enviada via uma mensagem publicada em um tópico do servidor MQTT específico para o usuário em risco.

Importante ressaltar que o sistema não valida o diagnóstico, pois isto requer informações pessoais. Portanto, está sujeito a falsos positivos. As subseções seguintes detalham os serviços de registrar um usuário no sistema, de detecção de pessoas próximas e de processar um contato recebido do usuário, e a implementação do sistema.

3.2. Registro do usuário

Os dados necessários são obtidos pela própria aplicação, são eles: um par de chaves assimétricas e o identificador do sistema. Sistemas operacionais de dispositivos móveis proveem um identificador único do sistema[1]. Este identificador é a única informação associada ao usuário utilizada pelo sistema.

A aplicação gera um par de chaves com o algoritmo de curva elíptica e armazena na memória do aplicativo. A chave pública é compartilhada com o servidor no processo de registrar, enquanto a chave privada é utilizada pelo dispositivo para assinar as mensagens. Ao verificar a assinatura com a chave pública armazenada do usuário, o servidor consegue atestar que a mensagem, garantindo o controle de acesso e integridade da mensagem.

O SSAID é um número de 64 *bits* expresso como uma *string* hexadecimal, exclusivo para cada combinação de chave de assinatura de aplicativo, usuário e dispositivo. O sistema operacional Android exige que todos os aplicativos sejam assinados digitalmente com um certificado antes de serem instalados em um dispositivo ou atualizados. Portanto, o identificador é único para a aplicação que o obtém, mesmo que ele seja desinstalado e reinstalado. Importante ressaltar de identificadores de *hardware*, por exemplo, endereço MAC, podem ser facilmente duplicados e por isto, devem ser evitados.

A Figura 2 apresenta o processo de registro do usuário no sistema. O aplicativo busca um par de chaves e um identificador gerado pelo servidor a partir do seu SSAID. As situações em que o aplicativo não encontrará essas informações são: primeiro acesso do usuário no aplicativo e após apagar os dados do aplicativo. Quando não encontra o par de chaves e/ou o identificador, solicita o registro para o usuário. Em seguida, o aplicativo gera o par de chaves assimétricas e salva na memória.

[1]Neste trabalho, consideramos apenas o identificador do sistema Android, chamado de *Service Set Android Identifier* (SSAID) ou também *Android ID*

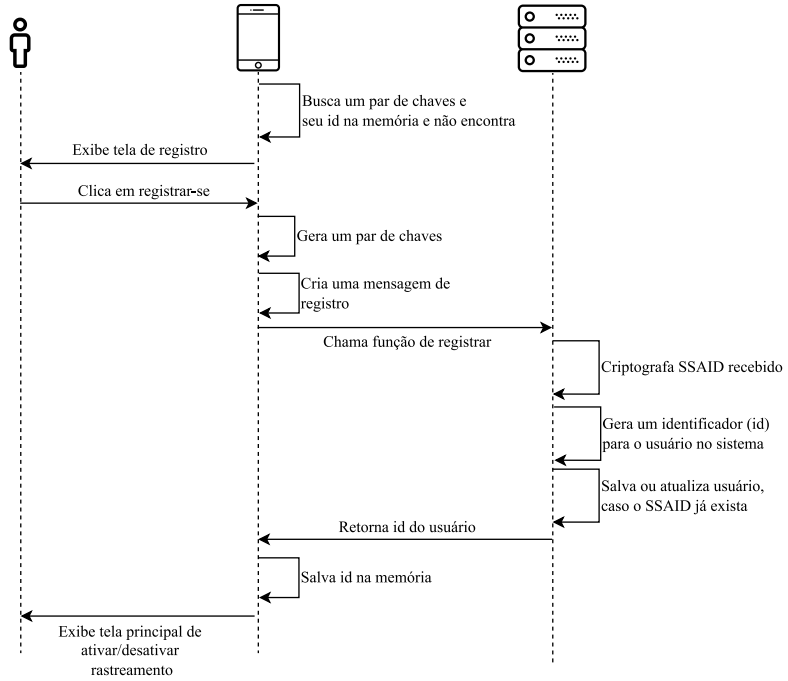


Figure 2. Processo de registrar um usuário no sistema.

O aplicativo envia para o servidor a chave pública e seu SSAID por meio de uma chamada de procedimento remoto (gRPC). O servidor, então, criptografa o SSAID e gera um identificador único (UUID) de 16 *bytes*, que será usado pelo aplicativo como anúncio nos beacons para dispositivos próximos. O SSAID não pode ser compartilhado com outros usuários, já que é uma informação privada usada para registrar ou atualizar um usuário.

O identificador gerado, o SSAID criptografado e a chave pública são salvos no banco de dados. Caso já exista um registro com o mesmo SSAID criptografado, a chave pública é atualizada, porque indica que o usuário se mantém com o mesmo dispositivo, mas desinstalou o aplicativo. Como o SSAID é único para aquele aplicativo e usuário naquele dispositivo, pode-se assumir ser a mesma pessoa. Nesse caso, o UUID gerado é descartado e o identificador já contido no banco é mantido. O servidor retorna o identificador UUID para o dispositivo, que será persistido na memória do aplicativo.

3.3. Detecção de contato

As chances de contágio aumentam com contato próximo e prolongado com uma pessoa infectada. A interface *Bluetooth* permite capturar a potência do sinal recebido do outro dispositivo e assim, estimar uma distância que auxiliará na avaliação de risco. Neste trabalho, os dispositivos móveis atuam como *beacons*, através da tecnologia *Bluetooth Low Energy*, transmitindo seu identificador único (UUID) utilizando o protocolo *AltBeacon*. O usuário registrado possui um UUID de 16 bytes, que será utilizado pelo serviço *beacon* da aplicação.

Ao habilitar o rastreamento de contatos, o dispositivo inicia o serviço beacon de transmissão e escaneamento em *foreground*, mantendo uma notificação fixa enquanto o serviço estiver ativo. A Figura 3 mostra o fluxo de detectar e enviar contatos para o *broker* MQTT.

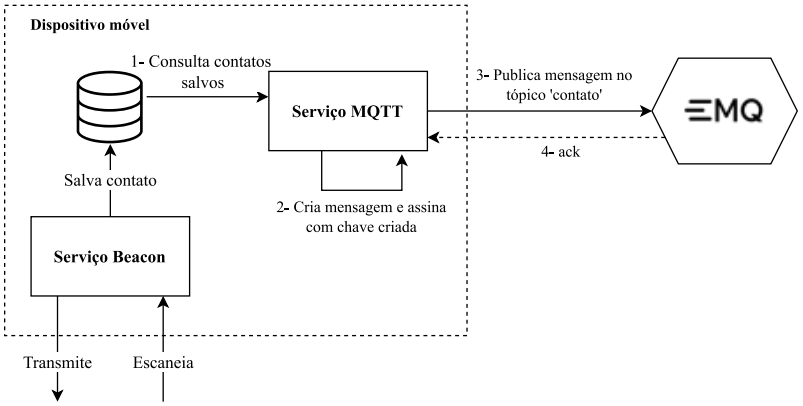


Figure 3. Detectar dispositivos próximos e enviar o contato para o serviço web.

O dispositivo envia *beacons* a cada 15 segundos e armazena-os como contatos em um banco de dados local. Um contato contém o identificador recebido, o *timestamp* do início do contato, o *timestamp* do fim do contato, a distância média aproximada, a intensidade do sinal recebido (RSSI) e o nível de bateria do usuário que recebeu aquele *beacon* no momento. Caso o usuário receba *beacons* com o mesmo identificador em um intervalo de 2 minutos com uma variação de distância de no máximo 1 metro entre as mensagens dos *beacons*, então o último registro de contato salvo no banco de dados referente àquele identificador é atualizado com um novo *timestamp* de fim de contato e a distância média recalculada. Caso o tempo entre as mensagens de *beacons* seja maior que 2 minutos ou a variação de distância maior que 1 metro, é necessário adicionar um novo registro de contato na tabela de contatos do banco.

O dispositivo atua como *publisher* MQTT e publica mensagens no tópico `contato` do servidor MQTT. A mensagem deve conter os dados do contato, o UUID do contato e o UUID do usuário que está enviando o contato. O sistema de autenticação se baseia em uma assinatura feita com a chave privada do usuário para cada mensagem enviada, garantindo autenticidade e integridade da mensagem. Dessa forma, o campo `signature` é adicionado na mensagem para ser validado pelo servidor, o qual contém um *array* de byte relativo à assinatura. Uma mensagem de publicação de contato é apresentada no formato JSON abaixo:

Listing 1. Mensagem de contato com assinatura

```
{
    "id": 1,
    "contact": {
        "otherUser":  "123e4567-e89b-12d3-a456-426655440000",
        "firstContactTimestamp": 2435465634,
        "lastContactTimestamp": 2435465815,
        "distance": 120, //cm
        "rssi": -15,
        "batteryLevel": 51
    },
    "user": "3d0ca315-aff9-4fc2-be61-3b76b9a2d798",
    "signature": [48,69,2,33,0,204,95,194,149,2,35,126,50,19,34,248, ...]
}
```

3.4. Reportar COVID-19 e Notificação de Contatos de Risco

Para reportar covid-19, o usuário deve informar a data de início dos sintomas e a data em que recebeu o diagnóstico positivo. Caso seja assintomático, é recomendado informar a data de início dos sintomas igual à data do diagnóstico. A mensagem de diagnóstico positivo contém as datas informadas pelo usuário e a data em que o usuário reportou, ou seja, a data do dia em que enviou a mensagem ao servidor.

Ao receber a mensagem, o serviço valida a assinatura da mensagem com a chave pública do usuário. Caso seja válida, o diagnóstico é salvo no banco de dados e também na cache. É importante salvar esse diagnóstico na cache com uma expiração de 15 dias para que, posteriormente, quando um usuário enviar um contato com uma pessoa infectada, o acesso ao diagnóstico seja rapidamente consultado e ele, seja notificado do risco de contágio. No entanto, o processamento desse diagnóstico continua acontecendo assincronamente. Este processo é composto por dois serviços que processam de maneira assíncrona: rastreador de contatos e notificador de alerta de risco.

3.4.1. Rastreador de contatos de risco

Segundo a OMS, ao ter contato com uma pessoa infectada por mais de 15 minutos em uma distância inferior a 2 metros, há um risco alto de contágio pela COVID-19 [World Health Organization 2020], e este contato deve ser notificado para se testar e eventualmente, se isolar. Do contrário, há um baixo risco.

O rastreador de contatos é o serviço responsável por buscar todos os contatos de um identificador que reportou diagnóstico positivo e filtrar os contatos em que há risco de contágio, seguindo recomendações da OMS. Para isto, o rastreador executa uma *query* no banco de dados, filtrando os contatos enviados pelo identificador origem, no período de 15 dias antes da data do diagnóstico. Em seguida é calculado a duração total entre os contatos, isto é, somar o tempo entre notificações subsequentes entre os mesmos contatos.

Os dados de contato são agregados de forma que, se a diferença entre dois contatos entre mesmos usuários (UUIDs) for menor que 20 minutos, então pode-se assumir se trata de um mesmo um contato, ou seja, a diferença de tempo entre esses dois contatos será incluída na duração do contato, portanto, esses dois contatos são considerados o mesmo. A Figura 4 ilustra um exemplo deste procedimento.

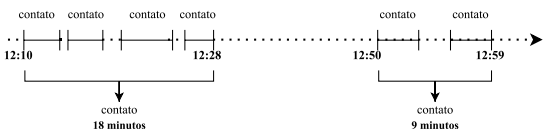


Figure 4. Agregação de contatos entre duas mesmas pessoas.

A partir da notificação de diagnóstico positivo de um usuário, a consulta retorna um conjunto de tuplas $[UUID_p, UUID_c, dist, dur]$, representando o identificador do usuário que notificou estar positivo, o identificador de um contato, a distância deste contato e a duração desta tupla de contato, respectivamente. Desta forma, o risco de contágio é calculado como:

$$Risco(UUID_p, UUID_c) = \begin{cases} 1, & \text{se } (dist \leq 2) \text{ E } (dur \geq 15) \\ 0, & \text{do contrário} \end{cases}$$

Para os casos em que $Risco = 1$, o usuário $UUID_c$ será notificado pelo serviço de Notificação de risco.

3.4.2. Notificador de contato de risco

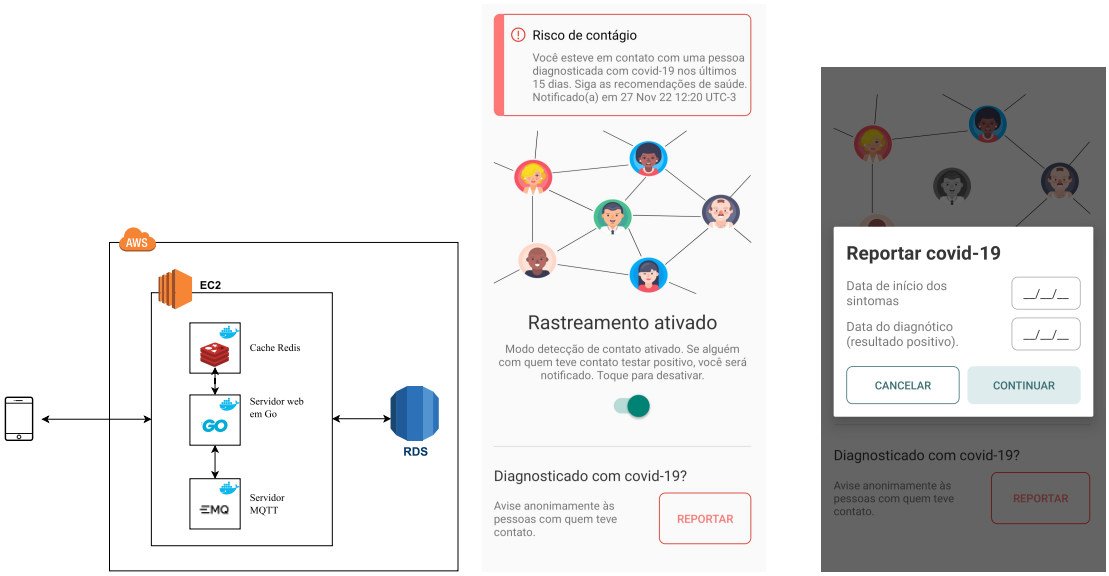
Para notificar usuários em risco de contágio, utilizamos o modelo de tópicos do MQTT para garantir a anonimidade do sistema. Conforme detalhado, ao se registrar no sistema, o dispositivo também subscreve no tópico cuja a *string* contem o próprio identificador único ($UUID$). Portanto, o serviço do notificado recebe o identificador do contato em risco ($UUID_c$) e publica uma mensagem em um tópico que segue o formato `notificacao/UUID>`. Por exemplo, o usuário com UUID `f234454c-f6d4-a0fa-df2f-4911ba9ffa6` subscreve no tópico `notificacao/f234454c-f6d4-a0fa-df2f-4911ba9ffa6` para receber notificações. Sendo assim, se este usuário está em risco, o notificador irá publicar uma mensagem de alerta neste tópico.

A cache é utilizada para evitar consultas no banco de dados a toda tarefa que for recebida, o que demandaria um tempo maior do que uma consulta direta nela. Para aumentar a eficiência da consulta e evitar que um usuário receba notificações duplicadas, o sistema utiliza uma cache para armazenar os contatos em risco por 15 dias. Após isso, outro serviço remove notificações expiradas na cache diariamente e notifica os usuários fora de risco.

3.5. Implementação e Deployment do Sistema

Para que o aplicativo móvel se comunique com servidor *web* e *broker* MQTT, é necessário que eles estejam acessíveis na internet. O *deploy* da aplicação foi realizado na plataforma de serviços de computação em nuvem *Amazon Web Services* (AWS). Os recursos utilizados foram Amazon RDS para o banco de dados PostgreSQL e uma instância Amazon EC2 para executar os servidores. Por questões econômicas, apenas uma instância EC2 foi utilizada, mas o ideal é que os servidores fiquem em máquinas diferentes. A Figura 5(a) apresenta a arquitetura da aplicação na nuvem. O servidor *web*, a cache Redis e *broker* MQTT foram virtualizados com *containers Docker*. A imagem do servidor *web* foi criada no *Dockerhub* para ser utilizada dentro instância EC2 e facilmente atualizada.

O aplicativo foi desenvolvido com a linguagem Java e um arquivo APK foi gerado para os usuários instalarem. A Figura 5(b) mostra a tela de rastreamento ativado onde o usuário foi notificado de risco de contágio. Na tela de rastreamento é possível observar o botão de reportar COVID-19. A Figura 5(c) exibe o que é solicitado do usuário ao clicar nesse botão: a data de início dos sintomas e a data em que recebeu o diagnóstico positivo.



(a) Arquitetura implantada na nuvem AWS. (b) Rastreamento ativado. (c) Reportar covid-19.

Figure 5. Deploy da aplicação Web e telas do aplicativo móvel de rastreamento.

4. Coleta e Análise dos dados

Para avaliar o sistema e validar o seu uso, um grupo de voluntários utilizou o aplicativo por um período de duas semanas. A partir desta coleta, foram analisados os grafos de contato e as propriedades destes contatos.

4.1. Metodologia da coleta de dados

Um formulário foi disponibilizado explicando os objetivos do sistema e esclarecendo quais dados são coletados durante o uso do aplicativo. Nele, os voluntários deveriam concordar com a coleta dos dados e recebiam o link para instalarem o aplicativo. Além disso, o formulário solicitava que os voluntários informassem um e-mail, não associado ao sistema, para eventual contato em um das etapas dos testes.

O formulário foi divulgado entre alunos da Universidade Federal do Espírito Santo. Os usuários foram instruídos a utilizar o sistema por algumas horas diariamente durante 2 semanas. Para simular a infecção, dois voluntários foram sorteados para reportar um diagnóstico positivo. Importante destacar que dado que os diagnósticos positivos não foram reais, as notificações não indicaram um risco de verdade.

4.2. Visualização e análise dos dados

Uma das principais contribuições do sistema é que apesar de anônimo, permite aos pesquisadores analisar a evolução do contágio e as propriedades dos contatos por meio dos grafos gerados a partir dos registros de identificadores únicos de contatos. Nestes grafos, um vértice representa uma pessoa e uma aresta, um contato entre duas pessoas. Quando um vértice da rede se infecta, a transmissão pode acontecer através dessas arestas. Um vértice com muitas arestas tem mais chances de se contaminar ou contaminar mais vértices. Arestas de um grafo podem ter pesos e, no caso do grafo de contatos, o peso pode ser a duração do contato e/ou a distância do contato, como também a quantidade de vezes que aqueles vértices tiveram contatos.

Também foi analisada a distribuição acumulada da duração e distância média entre os contatos, pois estas são informações essenciais para calcular o risco de contágio da COVID-19, além do grau médio dos vértices que compõem o grafo de contato. A Função de Distribuição Acumulada (FDA) dessas informações permite analisar a probabilidade de um indivíduo da amostra ter tido contato o vírus.

5. Resultados

Ao todo, 20 usuários se registram no sistema e 866 dados de contatos foram coletados entre os dias 06/11/2022 e 20/11/2022. Foram sorteados duas pessoas para reportarem diagnósticos positivos. A partir destas, três notificações de risco de contágio foram geradas. Dos 20 usuários registrados no sistema, apenas 17 usuários geraram dados de contatos.

Os dados dos contatos foram agregados considerando um contato constante, aquele no qual a diferença de um registro de contato e outro foi menor que 15 minutos. Portanto, doravante, um contato se refere a um conjunto de registros de contatos agregados A Tabela 1 apresenta um resumo dos dados coletados.

Table 1. Resumo dos dados coletados pelo sistema.

Usuários registrados	20
Usuários com contatos	17
Registros de contatos	866
Contatos constantes	341

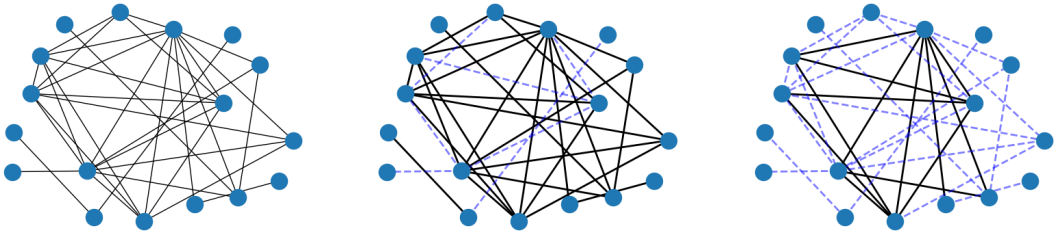
A Tabela 2 apresenta um sumário das métricas da distância, duração e também da quantidade de contatos por usuário. É possível observar que a distância média dos contatos coletados foi inferior a 200 cm, ou seja, uma distância considerada de risco. No entanto, o desvio padrão alto mostra que os valores de distância variaram bastante em relação à média. A distância mínima igual a zero indica que os usuários estavam tão próximos, que a distância foi desconsiderada. A duração dos contatos foi de cinco minutos, em média. No entanto, há uma grande diferença entre valor mínimo e valor máximo de duração. Os contatos de duração igual a zero significa que receberam apenas um *beacon* do outro. Por fim, em média, cada pessoa teve contato com quase outras 6 pessoas. Entretanto, isso inclui até aqueles contatos com duração igual a zero.

Table 2. Caracterização dos dados de contatos

Dado	**Média**	**Desvio Padrão**	**Mínimo**	**Máximo**
Distância (cm)	152,918	145,75	0,0	713.0
Duração (min)	5,085	7,933	0,0	55,203
Contatos constantes por usuário	5,879	5,127	0	21

A Figura 6 apresenta os grafos de contatos geral e considerando as arestas com peso de distância e duração de contato. A Figura 6(a) mostra o grafo de contatos geral, onde é possível ver as interações dos voluntários entre si ao longo das duas semanas. Dois indivíduos podem ter mais de um contato durante um intervalo de tempo, portanto, para

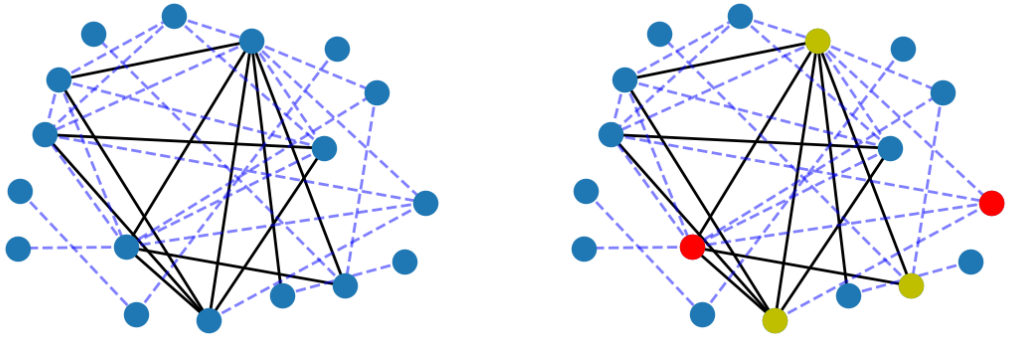
gerar os grafos foi considerada a média das distâncias e a maior duração dos contatos entre dois indivíduos. No caso da Figura 6(b), uma aresta tracejada na cor azul indica que o contato entre os dois indivíduos teve distância média superior a 200 cm, enquanto as arestas pretas indicam contatos com distância média inferior a 200 cm. Da mesma forma, para a Figura 6(c), onde as arestas tracejadas indicam contatos com duração menor que 15 minutos e as arestas na cor preta indicam contatos com duração maior que 15 minutos.



(a) Grafo de contatos geral. (b) Distância média. (c) Duração máxima.

Figure 6. Grafos de contatos geral (a) e ponderados por distância e duração.

O risco de fato aparece quando as duas variáveis, duração e distância são avaliadas, desse modo, a Figura 7(a) mostra o grafo de contato no qual as arestas pretas indicam contatos na qual a duração foi maior ou igual a 15 minutos e a distância média foi menor que 2 metros. Sendo assim, é possível visualizar os indivíduos que podem ser notificados caso um deles reporte um diagnóstico positivo.



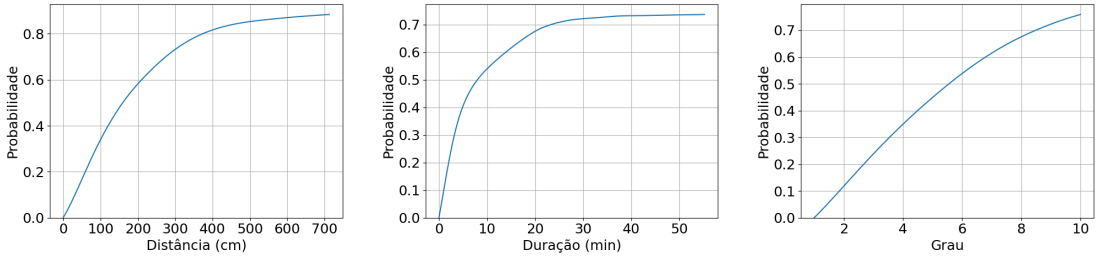
(a) Grafo de contatos ponderado por distância e duração simultaneamente. (b) nós infectados e notificados.

Figure 7. Análise dos grafos de contato.

Como citado anteriormente, dois voluntários foram selecionados para reportar um diagnóstico positivo. A Figura 7(b) apresenta os vértices infectados em vermelho e os vértices, em risco de contágio, que deverão ser notificados em amarelo. Observe que o servidor tem acesso apenas ao UUID dos nós de contato, e em um caso real, não seria possível obter a identidade dos vértices que notificaram positivo para COVID-19.

Outra forma de analisar os contatos coletados é observando as funções de distribuição acumulada das métricas. Desta forma, a Figura 8 apresenta a Função de

distribuição acumulada (FDA) da distância média, da duração dos contatos e do grau dos nós do grafo de contato, respectivamente.



(a) Distâncias dos contatos. (b) Duração dos contatos. (c) Graus dos nós do gráfico de contatos.

Figure 8. Funções de Distribuições Acumuladas (FDAs).

A Figura 8(a) mostra que por volta de 60% dos contatos registrados aconteceram em uma distância média de até 2 metros, enquanto apenas um pouco mais de 30% foram inferior a 1 metro de distância. Já a Figura 8(b) mostra que cerca de 72% dos contatos duraram até 30 minutos. Além disso, nota-se que grande parte dos contatos durou até 10 minutos, o que é evidenciado na FDA exibida na Figura 8(b). Observa-se que 54% dos contatos duraram até 10 minutos. Por fim, a Figura 8(c) diz respeito à probabilidade de um voluntário ter tido contato com uma certa quantidade de pessoas, já que mostra a FDA dos graus dos vértices do grafo de contato. Observa-se que por volta de 67% dos vértices do grafo possui até 8 arestas, ou seja, contato com até 8 pessoas distintas.

6. Conclusões e Trabalhos Futuros

Neste trabalho, foi proposto e desenvolvido um sistema de rastreamento de contatos anônimo, que teve como principal objetivo preservar a privacidade dos usuários para que se sentissem seguros em utilizá-lo. O trabalho fez uma análise das arquiteturas existentes e os problemas de privacidade que cada uma tem. Além disso, também foi levantada a questão da necessidade de poder analisar dados de contatos de usuários para estudar sobre o espalhamento do vírus, sem expor os usuários. Ao fim, voluntários foram convidados a utilizar o sistema por 2 semanas e os dados coletados foram analisados.

É importante destacar que a é anonimidade alcançada por meio do modelo de tópicos MQTT proposto pelo sistema, o qual utiliza apenas um identificador único, gerado a partir de um identificador do dispositivo do usuário. Desta forma, evita-se transmissão e armazenamento de qualquer informação que pudesse identificar os usuários ou mesmo quem realizou uma notificação de diagnóstico positivo. Por outro lado, o sistema permite aos pesquisadores analisar os tipos de contatos coletados e o potencial risco de contágio, por meio dos logs de rastreamento.

Uma limitação do sistema é a precisão da distância estimada pelos dispositivos a partir do sinal de *Bluetooth* recebido. Outro ponto é sobre como incentivar os usuários a utilizarem o aplicativo diariamente. Nos testes realizados, pode se observar que alguns usuários não utilizaram o aplicativo diariamente ao longo das semanas.

Como trabalhos futuros, é necessário avaliar a limitação a respeito da imprecisão da estimativa de distância a partir da intensidade de sinal. Uma distância mal estimada

pode ocasionar contatos de risco não identificados e com isso, usuários não notificados quando deveriam ou até mesmo o contrário, notificações para usuários que não tiveram contatos de risco. Além disso, seria importante buscar uma forma de validar o diagnóstico positivo do usuário, para evitar falsos positivos e notificações desnecessárias. Como a proposta do sistema é evitar coletar nenhuma informação pessoal, o mais adequado seria utilizar uma validação por terceiros, como uma entidade de saúde, de forma que não revele os dados pessoais do infectado, apenas se o diagnóstico é válido ou não.

Adicionalmente, o conjunto de dados gerado pode permitir a construção de um modelo de predição para identificar possíveis usuários infectados a partir do comportamento social destes. Como também identificar grupos com maior potencial de espalhamento do vírus. Por fim, o sistema proposto pode ser generalizado para novas doenças infecciosas, parametrizando dados que definem um risco de contágio.

Agradecimentos

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Código de Financiamento 001, do CNPq e da FAPES.

References

Ahmed, N., Michelin, R. A., Xue, W., Ruj, S., Malaney, R., Kanhere, S. S., Seneviratne, A., Hu, W., Janicke, H., and Jha, S. K. (2020). A survey of covid-19 contact tracing apps. *IEEE access*, 8:134577–134601.

Bay, J., Kek, J., Tan, A., Hau, C. S., Yongquan, L., Tan, J., and Quy, T. A. (2020). Bluetrace: A privacy-preserving protocol for community-driven contact tracing across borders. *Government Technology Agency-Singapore, Tech. Rep*, 18.

Castelluccia, C., Bielova, N., Boutet, A., Cunche, M., Lauradoux, C., Le Métayer, D., and Roca, V. (2020). Robert: Robust and privacy-preserving proximity tracing.

Kang, S.-J., Kim, S., Park, K.-H., Jung, S. I., Shin, M.-H., Kweon, S.-S., Park, H., Choi, S.-W., Lee, E., and Ryu, S. Y. (2021). Successful control of covid-19 outbreak through tracing, testing, and isolation: Lessons learned from the outbreak control efforts made in a metropolitan city of south korea. *Journal of Infection and Public Health*, 14(9):1151–1154.

Lee, E., Park, K., Park, D. J., Kim, J., and Jo, C. (2021). Locally testable privacy-preserving contact tracing protocol without exposing secret seed. In *2021 IEEE International Conference on Consumer Electronics (ICCE)*, pages 1–5.

Morio, K., Esiyok, I., Jackson, D., and Künnemann, R. (2023). Automated security analysis of exposure notification systems. In *USENIX Security Symposium*, pages 1–18. USENIX Association.

Rivest, R. L., Abelson, H., Callas, J., and Canetti, R. (2020). The pact protocol specification. Technical report, Instituto de Tecnologia Massachusetts.

World Health Organization (2020). Contact tracing in the context of covid-19. Disponível online em `https://apps.who.int/iris/bitstream/handle/10665/332049/WHO-2019-nCoV-Contact_Tracing-2020.1-eng.pdf`, Último acesso em 27/03/2023.