Proposta de análise de desempenho entre OpenID Connect e Keycloak usando OpenID Connect

Carlos D. S. Bunn [1], **Charles C. Miers**[1]

[1]Departamento de Ciência da Computação (DCC)
Universidade do Estado de Santa Catarina (UDESC)

carlos.bunn@edu.udesc.br,

charles.miers@udesc.br

Resumo. *A adoção de técnicas seguras para confiança digital é essencial para a eficiência e segurança. O Keycloak, ao atuar como serviço centralizado entre aplicações web e usuários, gerencia autenticação e autorização, mas introduz latência e potencial aumento no payload e dados trafegados. Este artigo propõe uma análise de desempenho do Keycloak, focando na delegação de identidades e tokens, visando identificar impactos significativos.*

1. Introdução

A autenticação é o processo de confirmar a identidade de uma entidade. Um processo de autenticação normalmente depende de alguma forma de comprovação. Enquanto autorização refere-se ao processo de verificar quais entidades podem acessar ou quais ações elas podem realizar [Krebs 2019]. Cada vez se torna mais complexa a tarefa de assegurar a identidade de indivíduos em um cenário virtual, bem como a de garantir a confiabilidade de suas informações nesse contexto.

O Keycloak é uma solução *open-source* de gestão de identidade e acesso que fornece funcionalidades, tais como a autenticação de utilizadores, autorização, *Single Sign-On* (SSO) e segurança para aplicações e serviços, podendo ser integrada em várias estruturas e plataformas [Secretary 2019]. Entre as diversas configurações do Keycloak, é possível implementar a tecnologia SSO, possibilitando a cada utilizador acesso a todos os recursos pretendidos com uma autenticação e credenciais únicas [Secretary 2019]. Assim, o Keycloak evita que uma aplicação precise interagir diretamente com vários sistemas de SSO (e.g., OpenID Connect e SAML), fazendo que acesse somente a interface do Keycloak. Porém, ao se tornar um intermediário entre usuário e serviços de autenticação, pode-se gerar latência e mudança nos *payload* de rede. O objetivo deste artigo é uma proposta para análise do desempenho do Keycloak, utilizando requisições padrões diretas do OpenID Connect e através do Keycloak para mensurar e comparar métricas de latência, *payload* e tráfego de rede.

A organização do artigo segue da seguinte forma. A fundamentação do Keycloak, extensão de seus documentos de identificação e métodos de rastreamento na Seção 2. A Seção 3 a proposta de análise do impacto no desempenho, acompanhada dos critérios utilizados na análise.

2. Fundamentação

Mantida pela empresa Red Hat e sendo de código aberto, o Keycloak é uma ferramenta de Gerenciamento de Identidade dos usuários. O mote da solução está em não ser necessário

ao usuário lidar com diferentes mecanismos de autenticação, e.g., armazenar senhas com a devida segurança necessária (outros métodos de segurança). As aplicações não passam a ter acesso direto às credenciais do usuário; em vez disso, recebem *tokens* de segurança que concedem acesso apenas ao necessário [Thorgersen and Silva 2021].

Podendo gerenciar diversas autorizações a diferentes aplicação, o Keycloak oferece recursos como SSO, autenticação multi-fator e gestão de utilizadores [Thorgersen and Silva 2021].

O Keycloak fornece uma console de administração fácil de utilizar para gerir utilizadores, funções e permissões e suporta padrões como OAuth 2.0 e OpenID Connect [Krebs 2019]. Visando fornecer meios para que o cliente adquira informações acerca da autenticação dos usuários, o OpenID é uma camada de identidade sobre a segunda versão do OAuth e permite que aplicações autentifiquem usuários e/ou obtenham informações básicas acerca de seus perfis.

As informações perante os atributos dos usuários e autenticação, são recebidos pelo aplicativo por meio de *tokens*, os quais são emitidos pelo servidor podendo ou não estar assinados e/ou cifrados [Krebs 2019].

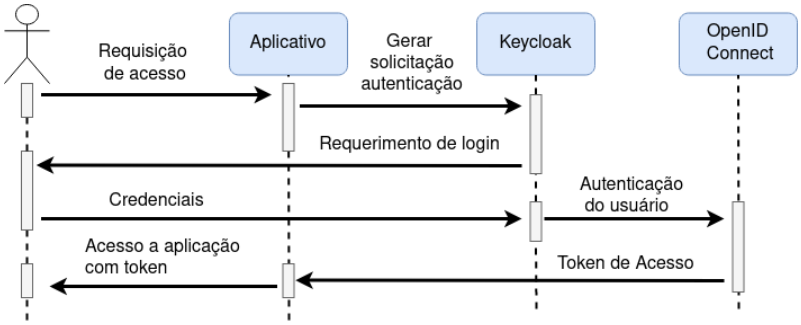

Figura 1. Sequência de sistema com Keycloak para acessar o OpenID Connect.

A Figura 1 representa o diagrama de sequência do Keycloak, ilustrando o fluxo de interação entre o usuário, a aplicação, o Keycloak e o *OpenID Connect*. Inicialmente, o usuário interage com a aplicação desejada, que então solicita a autorização de acesso ao Keycloak. O Keycloak redireciona o usuário para uma página de *login*, na qual serão inseridas as devidas credenciais. Com as credenciais corretas, o Keycloak delega a validação da identidade do usuário ao *OpenID Connect* (por exemplo, pode ser outro SSO). Caso as credenciais sejam válidas, o *OpenID Connect* gera um *token* de acesso destinado à aplicação, o qual posteriormente é recebido pela aplicação. Essas etapas compõem o processo de autenticação e autorização, assegurando que apenas os usuários autenticados tenham acesso aos recursos protegidos. Com a validação estabelecida pelo OpenID Connect a aplicação passa a ter salvo que o usuário possui acesso ao serviço. Desse modo o SSO funciona mantendo um estado de autenticação válido após o primeiro *login* bem-sucedido. Quando o usuário tenta acessar outra aplicação protegida pelo Keycloak, o sistema verifica se o estado de autenticação persiste e, se for o caso, concede automaticamente o acesso sem solicitar credenciais adicionais, resultando em navegação facilitada para o usuário entre diferentes aplicações sem interrupções desnecessárias de autenticação.

3. Proposta e critérios

A busca desta proposta é mensurar aspectos de desempenho do Keycloak por meio de requisições direcionadas ao *OpenID Connect*, durante a transferência de afirmações no gerenciamento de identidade e acesso em seu processo de execução. A análise envolve a mensuração e comparação de métricas de latência, *payload* e tráfego de rede. É comum que a utilização do Keycloak e do OpenID Connect tenha impactos, uma vez que a adição de um elemento na infraestrutura possa gerar tais efeitos. Além disso, a avaliação do desempenho do sistema também considera a integração e funcionamento do SSO. Identificar e quantificar as diferenças de desempenho proporciona uma compreensão sobre o funcionamento do sistema em termos de adequação, capacidade, eficiência e escalabilidade.

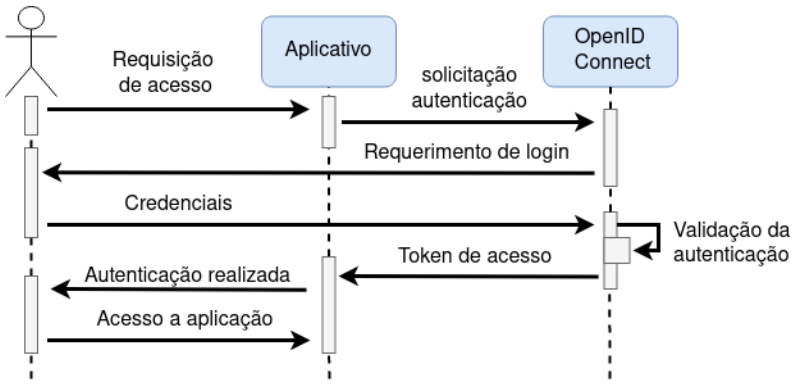

Figura 2. Diagrama de Sequência sistema apenas com OpenID Connect.

A Figura 2 representa a autenticação apenas com o uso do OpenID Connect como meio seguro para garantir o acesso dos usuários aos recursos protegidos da aplicação. O processo inicia-se com a requisição de acesso do usuário à aplicação, desencadeando assim uma requisição de acesso. O aplicativo solicita a autenticação ao OpenID Connect, que requer as credenciais do usuário. Após o usuário enviar suas credenciais ao OpenID Connect e estas serem verificadas, o OpenID Connect emite um *token* de acesso ao aplicativo. O aplicativo, então, envia a confirmação de autenticação ao usuário, concedendo-lhe acesso à aplicação. Ao utilizar o Keycloak como intermediário entre um aplicativo e o OpenID Connect, o processo de integração é simplificado, fornecendo uma camada de abstração que facilita a implementação e o gerenciamento de autenticação e autorização. Centralizando o controle de acesso com base em funções e políticas, o Keycloak oferece uma solução abrangente que permite uma administração eficaz dos privilégios dos usuários.

O *OpenID Connect* é um componente integrável ao sistema Keycloak, capaz de se comunicar diretamente com o Keycloak por meio de configurações predefinidas. As conexões entre os componentes são protegidas por Transport Layer Security (TLS), no qual o Keycloak valida e gerencia as credenciais do usuário em conjunto com o Lightweight Directory Access Protocol (LDAP). O processo envolve a emissão de um *token* para a aplicação seja liberando, que pode ser utilizando *OpenID Connect* ou direto pelo Keycloak. É importante que o Keycloak gerencie essa comunicação para evitar possíveis ataques de roubo de *tokens*, caso o *OpenID Connect* esteja fora da gerência do Keycloak.

Visando propor um *benchmark* do OpenID Conncet através do Keycloak (Figura 1) em relação ao uso puro do OpenID Connect (Figura 2) sobre o consumo rede, propondo capturar informações como: (i) latência; e (ii) *payload* de rede. Os seguintes cenários de experimentação são definidos para o plano de testes:

1. Cenário 2: comunicação utilizando Keycloak com OpenID Connect como gerencia de autenticação.
2. Cenário 1: comunicação utilizando OpenID como gerencia de autenticação.

Ao analisar a latência e o tráfego entre o usuário e o Keycloak em comparação com o acesso direto ao OpenID Connect, as diferenças surgem na complexidade e eficiência da comunicação. A conexão via Keycloak pode gerar um acréscimo na latência devido ao encaminhamento das solicitações de autenticação e autorização. Por outro lado, o acesso direto ao OpenID Connect pode diminuir a latência, mas aumenta o tráfego na rede, o que pode sobrecarregar o servidor em ambientes de alta demanda. Embora o Keycloak possa gerar uma sobrecarga no sistema, este simplifica o controle e monitoramento das credenciais dos usuários.

Empregando o uso de *shell scripts* e ferramentas como Prometheus e TCPDump para a obtenção dos dados, tem-se um arcabouço adequado à coleta de informações. Os *scripts* serão responsáveis pela automação dos procedimentos experimentais, permitindo a análise dos dados padronizada. Essa análise proporcionará uma compreensão detalhada do comportamento do sistema, avaliando os parâmetros por meio de ferramentas padrão disponíveis no GNU/Linux. Assim, será viável identificar as implicações no uso do Keycloak e aspectos de desempenho relacionados ao seu uso.

4. Considerações

O Keycloak oferece recursos para segurança dos aplicativos e serviços; no entanto, medidas como criptografia de dados, autenticação e verificações de segurança adicionais podem afetar a velocidade do sistema, assim impactando no desempenho, seja na rede, latência ou no processamento. Este trabalho não tem como objetivo se aprofundar no uso específico do processador, mas sim focar no o impacto do Keycloak no desempenho da rede e na latência das requisições, identificando possíveis gargalos que podem surgir na interação de diferentes elementos no sistema.

Agradecimentos: Os autores agradecem o apoio da LabP2D/UDESC e a FAPESC.

Referências

Krebs, B. (2019). *The OpenID Connect Handbook*. Auth0 by Okta.

Secretary, S. C. (2019). *IT Security and Privacy—A Framework for Identity Management-Part 1: Terminology and Concepts*. International Organization for Standardization, S.l.

Thorgersen, S. and Silva, P. I. (2021). *Keycloak-identity and access management for modern applications: harness the power of Keycloak, OpenID Connect, and OAuth 2.0 protocols to secure applications*. Packt Publishing Ltd.