

MAKALAH

TEORI MIKROSERVICE

KEAMANAN DALAM MIKROSERVICE

SEMESTER 4

DISUSUN OLEH

ESI PUTRIANI (2301081005)

JURUSAN TEKNOLOGI INFORMASI

PROGRAM STUDI TEKNIK KOMPUTER

POLITEKNIK NEGERI PADANG

2025

KATA PENGANTAR

Puji syukur saya haturkan kehadirat Allah Swt. yang telah melimpahkan rahmat dan hidayah-Nya sehingga saya bisa menyelesaikan makalah tentang "Keamanan Dala m Microservice".  
  
Tidak lupa juga saya mengucapkan terima kasih kepada semua pihak yang telah turut memberikan kontribusi dalam penyusunan karya ilmiah ini. Tentunya, tidak akan bisa maksimal jika tidak mendapat dukungan dari berbagai pihak.  
  
Sebagai penyusun, saya menyadari bahwa masih terdapat kekurangan, baik dari penyusunan maupun tata bahasa penyampaian dalam karya ilmiah ini. Oleh karena itu, saya dengan rendah hati menerima saran dan kritik dari pembaca agar saya dapat memperbaiki makalah ini.  
  
Saya berharap semoga makalah yang saya susun ini memberikan manfaat dan juga inspirasi untuk pembaca.

Padang, 08 April 2025  
  
  
Penulis

DAFTAR ISI

[KATA PENGANTAR II](#_Toc195020102)

[DAFTAR ISI III](#_Toc195020103)

[BAB I 1](#_Toc195020104)

[Latar Belakang Masalah 1](#_Toc195020105)

[Rumusan Masalah 1](#_Toc195020106)

[Tujuan 1](#_Toc195020107)

[BAB II 2](#_Toc195020108)

[Tantangan keamanan yang muncul dalam penerapan arsitektur microservice 2](#_Toc195020109)

[Jenis ancaman keamanan yang umum terjadi dalam lingkungan microservice 3](#_Toc195020110)

[Cara menerapkan mekanisme otentikasi dan otorisasi yang efektif antar layanan (service-to-service) dalam arsitektur microservice 4](#_Toc195020111)

[BAB III 6](#_Toc195020112)

[PENUTUP 6](#_Toc195020113)

[KESIMPULAN 6](#_Toc195020114)

BAB I

Pendahuluan

1. Latar Belakang Masalah

Seiring dengan kemajuan teknologi informasi, arsitektur microservice semakin banyak diterapkan dalam pengembangan sistem perangkat lunak modern. Arsitektur ini memberikan fleksibilitas, skalabilitas, serta kemudahan dalam proses pengembangan dan pemeliharaan aplikasi yang memiliki skala besar. Dalam model microservice, aplikasi dirancang sebagai sekumpulan layanan kecil yang beroperasi secara mandiri dan saling berkomunikasi melalui protokol jaringan, seperti HTTP/REST atau pesan berbasis event.Namun, meskipun ada banyak keuntungan yang ditawarkan oleh microservice, pendekatan ini juga membawa tantangan baru, terutama dalam hal keamanan. Berbeda dengan aplikasi monolitik yang biasanya memiliki satu titik masuk, sistem berbasis microservice terdiri dari banyak komponen yang saling terhubung. Setiap layanan berpotensi menjadi titik serangan yang perlu diamankan.

Beberapa ancaman yang umum ditemui dalam lingkungan microservice meliputi serangan man-in-the-middle, spoofing layanan, akses tidak sah antar layanan, serta risiko pengungkapan data sensitif melalui API. Oleh karena itu, diperlukan pendekatan keamanan yang menyeluruh, termasuk penerapan otentikasi dan otorisasi antar layanan (service-to-service authentication), enkripsi komunikasi menggunakan TLS, penggunaan gateway API, serta implementasi pola keamanan seperti Zero Trust dan Principle of Least Privilege.Lebih jauh lagi, karena microservice sering dijalankan dalam lingkungan berbasis cloud dan kontainer seperti Docker dan Kubernetes, pengelolaan keamanan kontainer, manajemen rahasia, dan pengawasan konfigurasi juga merupakan aspek penting yang harus diperhatikan.

Dengan semakin besarnya ketergantungan pada arsitektur microservice dalam sistem-sistem kritis, pemahaman dan penerapan strategi keamanan yang tepat menjadi hal yang sangat penting untuk menjaga integritas, kerahasiaan, dan ketersediaan layanan. Oleh karena itu, penelitian tentang keamanan dalam arsitektur microservice menjadi sangat relevan demi memastikan bahwa sistem yang dibangun tetap aman dan dapat diandalkan.

1. Rumusan Masalah
2. Bagaimana tantangan keamanan yang muncul dalam penerapan arsitektur microservice?
3. Apa saja jenis ancaman keamanan yang umum terjadi dalam lingkungan microservice?
4. Bagaimana cara menerapkan mekanisme otentikasi dan otorisasi yang efektif antar layanan (service-to-service) dalam arsitektur microservice?
5. Tujuan
6. Menjelaskan tantangan-tantangan keamanan yang dihadapi dalam penerapan arsitektur microservice.
7. Mengidentifikasi berbagai jenis ancaman keamanan yang umum terjadi dalam lingkungan microservice.
8. Menganalisis penerapan mekanisme otentikasi dan otorisasi yang efektif antar layanan (service-to-service) dalam arsitektur microservice.

BAB II

PEMBAHASAN MASALAH

1. Tantangan keamanan yang muncul dalam penerapan arsitektur microservice

Arsitektur microservice memungkinkan pengembangan aplikasi sebagai sekumpulan layanan kecil yang dapat beroperasi secara independen dan saling berkomunikasi melalui jaringan. Meskipun pendekatan ini menawarkan fleksibilitas, skalabilitas, dan kemudahan dalam pengembangan, ada juga kompleksitas tersendiri yang perlu dihadapi terkait keamanan.

Berikut adalah beberapa tantangan utama yang muncul dalam aspek keamanan microservice:

1. Permukaan Serangan yang Lebih Luas

Dalam sistem monolitik, aplikasi biasanya memiliki satu titik masuk (entry point) yang mempermudah penerapan kebijakan keamanan secara terpusat. Namun, dalam arsitektur microservice, terdapat banyak layanan yang dapat diakses secara individual, baik dari dalam maupun luar. Setiap layanan ini berpotensi menjadi titik serangan, sehingga memperluas permukaan serangan (attack surface) sistem secara keseluruhan.

1. Keamanan Komunikasi Antar Layanan

Karena layanan dalam microservice berkomunikasi melalui jaringan (umumnya menggunakan protokol HTTP atau gRPC), komunikasi antar layanan perlu dilindungi dari berbagai ancaman, seperti penyadapan, manipulasi data, dan serangan ulang (replay attack). Tanpa adanya enkripsi dan otentikasi yang memadai, lalu lintas data antar layanan menjadi rentan terhadap serangan man-in-the-middle (MitM).

1. Manajemen Identitas dan Akses yang Kompleks

Setiap layanan dalam arsitektur microservice harus dapat memverifikasi identitas pengguna atau layanan lain yang mengaksesnya. Ini memerlukan sistem otentikasi dan otorisasi yang skalabel dan dapat dikelola dengan konsisten di seluruh layanan. Tanpa pendekatan yang terpusat atau otomatis, pengelolaan hak akses dapat menjadi tidak efisien dan tidak aman.

d. Kurangnya Visibilitas dan Pengawasan Terpusat

Dengan adanya banyak layanan yang tersebar, pelacakan aktivitas serta deteksi anomali menjadi lebih menantang. Logging dan monitoring yang tidak terintegrasi dapat menyebabkan keterlambatan dalam mendeteksi serangan atau kesalahan konfigurasi yang membahayakan keamanan.

e. Ketergantungan pada Layanan Pihak Ketiga

Microservice sering menggunakan layanan eksternal atau pustaka pihak ketiga, seperti layanan autentikasi, database-as-a-service, atau komponen open-source. Jika tidak dipantau dan diperbarui secara berkala, komponen pihak ketiga ini dapat menjadi pintu masuk serangan melalui celah keamanan yang sudah dikenal.

f. Otomatisasi dan DevOps yang Tidak Aman

Dalam lingkungan DevOps yang bergerak cepat, perubahan kode, konfigurasi, atau penyebaran terjadi secara rutin. Apabila proses otomatisasi ini tidak dilengkapi dengan validasi keamanan yang memadai, kesalahan konfigurasi atau skrip berbahaya dapat dengan mudah masuk ke dalam lingkungan produksi tanpa terdeteksi.

1. Jenis ancaman keamanan yang umum terjadi dalam lingkungan microservice

Lingkungan microservice yang terdistribusi dan saling berkomunikasi melalui jaringan memiliki tantangan tersendiri dalam hal keamanan. Setiap layanan yang terhubung berpotensi menjadi titik serangan jika tidak dilindungi dengan mekanisme keamanan yang tepat. Berikut adalah beberapa ancaman keamanan yang umum terjadi dalam arsitektur microservice:

a. Serangan Man-in-the-Middle (MitM)

Serangan ini terjadi ketika penyerang berhasil menyusup ke dalam komunikasi antara dua layanan, lalu menyadap atau memanipulasi data yang sedang dikirim. Ancaman ini sangat berbahaya, terutama jika layanan tidak menggunakan protokol terenkripsi seperti HTTPS atau mutual TLS (mTLS). Informasi sensitif, seperti token akses atau data pengguna, dapat dicuri dan disalahgunakan.

b. Service Spoofing

Dalam serangan ini, penyerang menciptakan layanan palsu yang meniru layanan asli. Layanan palsu ini dapat digunakan untuk mengelabui layanan lain agar mengirimkan data sensitif atau memberikan akses tidak sah. Insiden ini dapat terjadi jika tidak ada validasi identitas yang efektif antara layanan.

c. Eksposur API

API adalah antarmuka utama dalam komunikasi microservice. Jika API tidak dilindungi dengan baik, penyerang dapat mengeksplorasi dan menemukan endpoint yang tidak aman atau yang tidak dilindungi oleh otorisasi. Risiko ini sering dikenal sebagai API Enumeration atau API Abuse.

d. Insecure Deserialization

Beberapa layanan microservice menerima input dalam bentuk objek yang telah diserialisasi. Jika layanan tidak memverifikasi atau menyaring data ini dengan benar, penyerang dapat mengirim payload berbahaya, yang dapat menyebabkan eksekusi kode yang tidak sah di sisi server.

e. Denial-of-Service (DoS)

Microservice yang tidak menerapkan pembatasan jumlah permintaan (rate limiting) rentan terhadap serangan DoS. Penyerang dapat mengirimkan permintaan dalam jumlah besar sehingga membuat layanan kehabisan sumber daya dan tidak dapat merespons permintaan yang sah dari pengguna lainnya.

f. Autentikasi dan Otorisasi yang Buruk

Layanan yang tidak menerapkan autentikasi dan otorisasi dengan benar berisiko memberikan akses kepada pengguna atau layanan yang tidak sah. Kesalahan semacam ini, seperti penggunaan token yang mudah dipalsukan atau ketidakmampuan dalam memverifikasi izin akses secara tepat, sangat berisiko bagi keamanan data.

g. Kerentanan pada Dependensi

Microservice sering kali bergantung pada pustaka eksternal atau layanan pihak ketiga. Jika pustaka ini memiliki celah keamanan dan tidak diperbarui secara berkala, maka microservice menjadi rentan terhadap eksploitasi melalui ketergantungan tersebut.

h. Logging dan Monitoring yang Tidak Memadai

Ketiadaan sistem pemantauan yang memadai bisa mengakibatkan serangan tidak terdeteksi. Log yang tersebar di berbagai layanan tanpa sistem pemantauan terpusat menyulitkan proses investigasi dan respons terhadap insiden keamanan.

1. Cara menerapkan mekanisme otentikasi dan otorisasi yang efektif antar layanan (service-to-service) dalam arsitektur microservice

Dalam arsitektur microservice, komunikasi antar layanan berlangsung secara terus-menerus dan dinamis. Agar sistem tetap aman, sangat penting untuk memastikan bahwa hanya layanan yang terverifikasi yang dapat saling berinteraksi dan mengakses data satu sama lain. Untuk mencapai hal ini, diperlukan mekanisme otentikasi (verifikasi identitas) dan otorisasi (pemberian hak akses) yang efisien dan aman.

Berikut adalah beberapa pendekatan dan teknologi yang umum digunakan dalam menerapkan otentikasi dan otorisasi antar layanan di dalam microservice:

a. Penggunaan Mutual TLS (mTLS)

Mutual TLS adalah metode enkripsi komunikasi dua arah yang tidak hanya mengenkripsi data, tetapi juga melakukan verifikasi identitas kedua pihak yang terlibat dalam komunikasi. Setiap layanan memiliki sertifikat digitalnya masing-masing. Ketika layanan A ingin berkomunikasi dengan layanan B, keduanya akan saling memverifikasi sertifikat yang dimiliki sebelum koneksi dilakukan. mTLS sering dipakai dalam lingkungan service mesh seperti Istio, Linkerd, atau Consul Connect untuk memastikan bahwa hanya layanan yang terverifikasi yang dapat berkomunikasi.

b. Token Berbasis JSON Web Token (JWT)

JWT adalah standar token yang umum digunakan untuk menyimpan informasi otorisasi dengan cara yang aman. Ketika satu layanan mengakses layanan lain, token JWT dikirim sebagai bagian dari header permintaan (biasanya di header Authorization: Bearer). Layanan penerima akan memverifikasi tanda tangan token tersebut dan memeriksa klaim yang ada di dalamnya, seperti identitas pengirim, hak akses, dan masa berlaku token. JWT cocok digunakan dalam arsitektur terdistribusi karena sifatnya yang stateless dan mudah untuk di-decode oleh layanan penerima.

c. OAuth 2. 0 untuk Delegasi Akses

OAuth 2. 0 menyediakan kerangka kerja otorisasi yang memungkinkan suatu layanan untuk mengakses sumber daya atas nama pengguna atau layanan lainnya. Dalam hal ini, layanan menggunakan access token yang dikeluarkan oleh server otorisasi seperti Keycloak, Auth0, atau Okta. Token ini digunakan untuk mengakses layanan lain berdasarkan ruang lingkup (scope) dan peran (roles) yang telah ditentukan sebelumnya.

d. Penggunaan API Gateway sebagai Gerbang Keamanan

API Gateway dapat berfungsi sebagai titik masuk tunggal dalam mengatur seluruh permintaan yang akan diteruskan ke layanan-layanan internal. Tugas API Gateway meliputi validasi token (JWT/OAuth), autentikasi pengguna atau layanan, serta pemeriksaan hak akses. Setelah permintaan berhasil melewati proses validasi di API Gateway, permintaan tersebut diteruskan ke layanan internal dengan disertai header keamanan atau token internal khusus.

e. Implementasi Role-Based Access Control (RBAC)

RBAC memungkinkan pengaturan hak akses berdasarkan peran (role) yang dimiliki oleh layanan atau pengguna. Setiap layanan diberikan peran tertentu, dan akses ke layanan lain dibatasi hanya untuk peran yang berwenang. RBAC dapat diintegrasikan dengan JWT, OAuth, dan service mesh untuk memberikan pengelolaan otorisasi yang terpusat dan fleksibel.

f. Service Mesh untuk Otomatisasi Keamanan

Service mesh adalah arsitektur infrastruktur yang menyediakan komunikasi antar layanan dengan fitur keamanan yang terintegrasi. Dengan menggunakan service mesh seperti Istio, administrator dapat mengatur kebijakan keamanan, termasuk otentikasi, otorisasi, enkripsi lalu lintas, dan audit log, tanpa perlu merubah kode aplikasi. Service mesh mendukung penerapan Zero Trust Security, di mana setiap layanan dianggap tidak tepercaya hingga terbukti sebaliknya.

BAB III

PENUTUP

KESIMPULAN

Penerapan arsitektur microservice dalam pengembangan perangkat lunak menawarkan sejumlah keuntungan, seperti skalabilitas, fleksibilitas, dan kemudahan dalam pemeliharaan. Namun, di balik berbagai kelebihan tersebut, terdapat tantangan signifikan terkait keamanan yang harus diperhatikan.

Keamanan di lingkungan microservice menjadi lebih rumit akibat sifatnya yang terdistribusi serta intensitas komunikasi antar layanan. Beberapa tantangan utama yang muncul adalah perluasan permukaan serangan, kesulitan dalam mengelola identitas dan akses antar layanan, serta potensi ancaman dari ketergantungan pada pustaka eksternal. Berbagai jenis serangan, seperti man-in-the-middle, spoofing, penyalahgunaan API, dan denial-of-service, dapat terjadi jika tidak ada mekanisme perlindungan yang memadai.

Untuk menghadapi tantangan tersebut, sangat penting menerapkan mekanisme otentikasi dan otorisasi yang efektif antar layanan. Penggunaan mutual TLS (mTLS), token JWT, OAuth 2. 0, serta pemanfaatan API Gateway dan service mesh adalah beberapa solusi yang dapat diimplementasikan. Strategi keamanan yang baik dalam ekosistem microservice sebaiknya mengikuti prinsip zero trust, yang menganggap setiap layanan sebagai tidak tepercaya hingga ada bukti sebaliknya.