WEARABLE HEALTH
MONITORING SYSTEM

Disusun Oleh:

1. **Azlin Niken Oktivani (2403013)**
2. **Siti Sa'adah (2403001)**

Internet Of Things

Sistem Pilih: Wearable Health

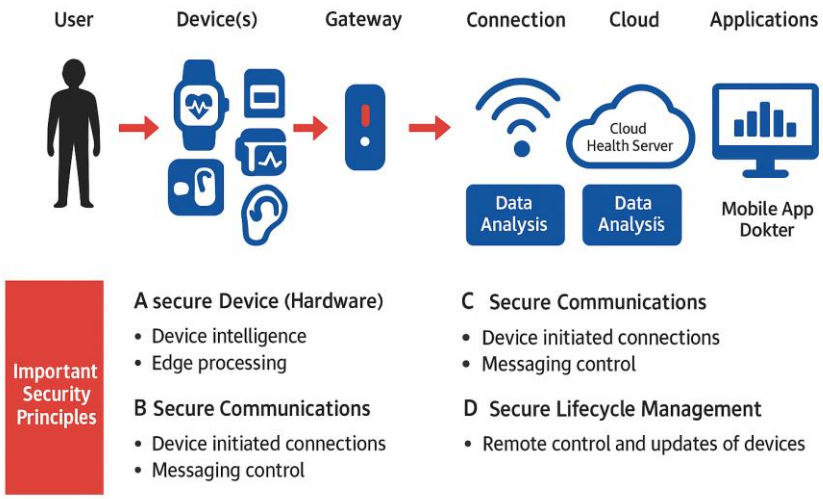
Dosen Pembimbing: Ahmad Rifa'i, S.Tr.Kom., M.Tr.Kom.

Tanggal: 04 November 2025

POLITEKNIK NEGERI INDRAMAYU

1. PENDAHULUAN

Wearable Health atau **perangkat kesehatan yang dapat dikenakan** adalah teknologi berupa alat elektronik kecil yang dipakai langsung pada tubuh manusia, biasanya dalam bentuk jam tangan pintar (smartwatch)**,** gelang kebugaran (fitness tracker)**,** patch sensor, atau bahkan pakaian pintar (smart clothing)**.** Tujuan utamanya adalah memantau kondisi kesehatan secara real-time dengan mengumpulkan data biologis pengguna, seperti: Detak Jantung (heart rate), Saturasi Oksigen(SpO2), Tekanan darah, pola tidur, aktivitas fisik & jumlah Langkah, suhu tubuh, elektrokardiogram(EKG) perangkat ini menghubungkan data ke smartphone atau sistem cloud, di mana informasi dapat dianalisis untuk memberikan insight tentang kesehatan tubuh, kebugaran, dan potensi risiko medis**.**



2. TITIK ANALISA

No	Titik (Component)	Aktor	Jenis Serangan	Trade-off	Keparahan	Solusi
1	Smart Watch	Cyber criminal	MITM	Biaya tinggi	High	Enkripsi end-to-end
2	Gelang Monitoring Jantung	Nation-state aktor	DDoS	Latency jaringan meningkat	Medium	Segmentasi jaringan
3	Sensor tekanan darah	Criminal Hacker	Bluetooth Sniffing	Konsumsi daya meningkat	Medium to High	Bluetooth Low Energy Secure Connections
4	Earphone Monitoring Oksigen	Insider Attack	Data Exfiltration	Waktu pairing lebih lama	Low-Medium	Bluetooth Secure Simple Pairing
5	Glucometer	Hacktivist	SQL Injection	Waktu pengembangan meningkat	High	Input Validation

a. Smartwatch

- Lokasi Dalam Arsitektur
  Device (smartwatch) → Gateway → Network → Cloud → App
  Smartwatch berfungsi sebagai **sensor utama** yang mendeteksi berbagai parameter kesehatan (detak jantung, tekanan darah, oksigen darah, langkah, tidur).
  Data dari smartwatch dikirim melalui **Bluetooth atau Wi-Fi** ke smartphone (gateway), lalu diteruskan ke **cloud server** untuk disimpan, diolah, dan divisualisasikan di **aplikasi mobile atau dashboard puskesmas**.

- Deskripsi Celah atau Rentan
  Firmware dan aplikasi itu tidak rutin untuk diperbarui, banyak smartwatch mengunakan system operasi atau firmware dengan patch keamanan yang tidak di update, sehingga bug lama lama tetap aktif dan bisa dimanfaatkan hacker. Transmisi datanya tidak terenskripsi jadi saat smartwatchmengirim data melalui Bluetooth klasik (tanpa BLE secure mode) data Kesehatan seperti detak jantung atau lokasi bisa disadap. Selain itu akses aplikasi juga longgar karena data yang diterima sering tidak menerapkan autentikasi kuat. Integrasi cloud tanpa proteksi kuat jadi data yang tersimpan dapat sesekali terekspos jika terjadi konfigurasi misalnya bucket cloud terbuka (misconfigured storage)

- Jenis Serangan
  MITM (Man-In-the-Middle), yang dapat menyusup ke komunikasi antara smartwatch dan gateway. Bisa juga Firmware Tampering yang memodifikasi firmware agar mengirim data palsu atau menyalin data pengguna. Selain itu, Brute Force Account Attack yang menebak kredensial akun pengguna di aplikasi monitoring.
- Serangan yang dilakukan
  1) Penyerang membuat hotspot palsu (fake Wi-Fi AP) atau menyusup pada koneksi Bluetooth.
  2) Ketika smartwatch terhubung ke gateway, penyerang menyadap paket data (sniffing) untuk menangkap data vital pengguna.
  3) Jika firmware smartwatch tidak aman, penyerang dapat memasang firmware palsu melalui proses update yang tidak diverifikasi, lalu firmware ini mengirim data pengguna ke server milik penyerang**.**
  4) Di sisi aplikasi, brute-force dilakukan untuk menebak password pengguna dan mengambil data dari cloud.
- Aktor & Motivasi Penyerangan
  1) Cyber criminal: Mencuri data medis untuk dijual atau digunakan dalam penipuan asuransi / identitas.
  2) Insider (orang dalam): Petugas atau staf yang memiliki akses ke data pasien dan menyalahgunakannya.
  3) Hacktivist: Mungkin ingin mengekspos kelemahan sistem kesehatan digital
- Dampak serangan
  1) Kerahasiaan data pasien bocor (detak jantung, pola tidur, lokasi, bahkan identitas).
  2) Risiko diagnosa keliru jika data dipalsukan melalui firmware tampering.
  3) Kepercayaan masyarakat terhadap sistem wearable health menurun.
  4) Potensi penyalahgunaan data medis untuk profiling atau diskriminasi.
- **Tingkat Keparahan**: High, karena smartwatch sangat umum digunakan dan seringkali memiliki konektivitas terus-menerus tanpa perlindungan memadai.
- **Solusi**
  1) **Enkripsi end-to-end (AES-256 / TLS 1.3)** pada setiap komunikasi Bluetooth/Wi-Fi.
  2) **Verifikasi digital signature firmware** agar update tidak bisa dimodifikasi.
  3) **Autentikasi dua faktor (2FA)** pada aplikasi pengguna.
  4) **Secure boot pada perangkat** agar hanya firmware sah yang bisa dijalankan.
  5) **Prosedur pembaruan firmware aman** (melalui server resmi, bukan file lokal).
  6) **Pelatihan staf teknis** untuk mengenali potensi serangan spoofing perangkat.
  7) **Audit keamanan rutin** pada cloud dan aplikasi mobile.

- **Trade-off atau Kendala**
  Biaya pengembangan dan pembaruan firmware yang aman cenderung tinggi, sementara keterbatasan perangkat keras pada beberapa smartwatch, seperti daya prosesor yang rendah, menghambat penerapan enkripsi yang kuat. Selain itu, diperlukan pelatihan tambahan bagi staf, terutama di puskesmas atau desa yang baru memulai proses digitalisasi.
- Cara Verifikasi Mitigasi:
  1) Uji penetrasi koneksi Bluetooth dilakukan untuk mengidentifikasi dan mencegah potensi kebocoran data saat perangkat berkomunikasi. Proses ini melibatkan simulasi serangan pada koneksi Bluetooth untuk memastikan transmisi data terlindungi dari penyadapan atau intersepsi oleh pihak tak berwenang.
  2) Tes keamanan firmware melalui validasi checksum sebelum dan sesudah pembaruan bertujuan untuk memastikan integritas perangkat lunak. Checksum digunakan untuk mendeteksi perubahan atau korupsi pada firmware, sehingga memastikan bahwa firmware yang diinstall adalah asli dan tidak dimodifikasi oleh pihak yang tidak memiliki hak.
  3) Monitoring akses cloud dan aplikasi melibatkan pencatatan aktivitas login dan penggunaan sistem secara real-time, dengan tujuan mendeteksi adanya pola aktivitas mencurigakan seperti upaya login berulang yang gagal atau akses dari lokasi yang tidak biasa. Hal ini membantu mencegah akses tidak sah dan menjaga keamanan data yang tersimpan di cloud dan aplikasi.

b. Gelang Monitoring Detak Jantung

- **Lokasi Dalam Arsitektur**
  Device (smartwatch) → Gateway → Network → Cloud → App
  Data dari gelang dikirim melalui **Bluetooth atau Wi-Fi** ke gateway (biasanya smartphone atau router lokal), lalu diteruskan ke cloud untuk disimpan dan diolah. Aplikasi pada ponsel atau komputer petugas kemudian menampilkan hasil monitoring dalam bentuk grafik dan notifikasi.
- **Deskripsi Celah atau Rentan**

  Banyak gelang kesehatan murah tidak menggunakan enkripsi seperti SSL/TLS atau AES encryption, sehingga data seperti detak jantung, suhu, dan aktivitas berisiko disadap saat dikirim melalui jaringan Wi-Fi publik atau Bluetooth. Selain itu, firmware yang jarang diperbarui membuat celah keamanan lama tetap ada dan dapat dimanfaatkan oleh penyerang. Gelang tersebut juga sering memiliki sistem autentikasi yang lemah antara perangkat dan aplikasi, sehingga memudahkan proses pembajakan atau pemalsuan perangkat.

- Jenis Serangan
  1) **MITM (Man-in-the-Middle):** Penyadapan data ketika dikirim dari gelang ke gateway.

2) **DDoS (Distributed Denial of Service):** Mengganggu koneksi ke cloud hingga layanan monitoring berhenti.
3) **Data Exfiltration:** Pengambilalihan data pengguna dari cloud server untuk dijual atau disalahgunakan.
4)

- Serangan yang dilakukan
  1) Penyerang menggunakan **Wi-Fi publik** atau hotspot palsu di area puskesmas/desa.
  2) Saat gelang pengguna mengirim data ke gateway, penyerang melakukan **MITM attack** untuk menyadap data kesehatan yang dikirim.
  3) Dalam skenario lain, penyerang bisa melakukan **overload traffic ke cloud server** sehingga sistem monitoring lumpuh dan layanan puskesmas tidak bisa menampilkan hasil pasien.
  4) Bila firmware gelang tidak aman, penyerang dapat mengirimkan **update firmware palsu** yang berisi malware.

- Aktor & Motivasi Penyerangan
  1) **Nation-state actor:** Ingin melakukan spionase atau pengumpulan data demografis kesehatan masyarakat.
  2) **Cyber criminal:** Mencuri data kesehatan untuk dijual di dark web atau untuk pemerasan (ransom).

- Dampak serangan
  Serangan siber dapat menyebabkan kebocoran data pribadi pasien yang sangat sensitif, seperti nama, informasi detak jantung, dan aktivitas fisik, yang berpotensi disalahgunakan untuk tujuan kriminal atau pelanggaran privasi. Selain itu, serangan DDoS atau tindakan sabotase dapat menghentikan layanan monitoring kesehatan digital secara tiba-tiba, sehingga mengganggu pemantauan kondisi pasien secara real-time dan berisiko pada keselamatan mereka. Dampak ini juga menimbulkan penurunan kepercayaan masyarakat terhadap teknologi kesehatan digital, yang dapat memperlambat adopsi inovasi penting dalam layanan medis. Lebih jauh lagi, manipulasi data yang diterima oleh sistem dapat menyebabkan diagnosa yang salah, sehingga mengarah pada pengambilan keputusan medis yang keliru dan berpotensi membahayakan kesehatan pasien.

- **Tingkat Keparahan**: Medium, karena perangkat relative sederhana dan tidak memiliki system pertahanan kompleks tetapi dampaknya cukup besar jika dikompromikan.

- **Solusi**
  1) **Segmentasi jaringan:** Pisahkan jaringan wearable dari jaringan administrasi puskesmas.

2) **Proteksi firewall & IDS/IPS:** Deteksi penyusupan dan penyadapan data real-time.
3) **Enkripsi komunikasi (AES-256 / TLS 1.3):** Lindungi data dari MITM.
4) **Backup data berkala:** Hindari kehilangan data akibat serangan DDoS atau ransomware.
5) **Pelatihan staf puskesmas/desa** dalam keamanan data medis digital.
6) **Kebijakan keamanan firmware:** Update wajib dan hanya dari sumber resmi.
7) **Sosialisasi keamanan kepada warga** yang menggunakan perangkat wearable.

- **Trade-off atau Kendala**
  Penggunaan enkripsi data dalam sistem jaringan menyebabkan peningkatan latency, yaitu keterlambatan waktu pengiriman data, karena proses enkripsi dan dekripsi memerlukan waktu pemrosesan tambahan. Hal ini bisa berdampak pada kecepatan respons sistem terutama dalam pengiriman data real-time seperti monitoring kesehatan. Dari sisi infrastruktur, implementasi keamanan seperti segmentasi jaringan memerlukan investasi ekstra berupa server tambahan dan perangkat firewall yang dapat memisahkan dan melindungi trafik data secara lebih efektif, sehingga menambah beban biaya operasional. Selain itu, kesiapan sumber daya manusia menjadi faktor penting, karena petugas yang menangani sistem harus memiliki pengetahuan dasar tentang keamanan siber untuk dapat menjalankan, memantau, serta merespons insiden keamanan dengan tepat, sehingga menjaga kelangsungan dan keamanan operasional layanan digital.

- Cara Verifikasi Mitigasi:
  Untuk memastikan efektivitas mitigasi serangan, pertama dilakukan simulasi serangan DDoS secara terkendali guna menguji ketahanan sistem cloud dan gateway terhadap lonjakan trafik berbahaya, sehingga dapat mengidentifikasi titik lemah dan kesiapan sistem dalam menghadapi serangan nyata. Selanjutnya, audit log server dan gateway secara rutin dilakukan untuk memeriksa pola akses yang tidak biasa atau mencurigakan, yang bisa menjadi indikasi percobaan pelanggaran keamanan atau aktivitas berbahaya dalam jaringan. Selain itu, uji penetrasi secara berkala dilakukan dengan tujuan mengevaluasi kehandalan enkripsi data, pengaturan firewall, serta pembaruan patch firmware, sehingga memastikan semua lapisan keamanan berfungsi optimal dan cepat menanggapi potensi celah yang muncul akibat perkembangan teknologi atau metode serangan baru.

c. Sensor Tekanan Darah Portabel

- **Lokasi Dalam Arsitektur**
  Sensor Tekanan Darah → Smartwatch / Smartphone (gateway) → Cloud Storage → Dashboard Medis

Sensor tekanan darah portabel menjadi bagian penting dalam sistem wearable health yang dapat beroperasi secara mandiri, seperti tensimeter digital yang menggunakan Bluetooth, atau menyatu dalam perangkat smartwatch. Sensor ini mampu mengukur tekanan darah secara detail, yaitu tekanan sistolik dan diastolik, kemudian mengirimkan hasil pengukuran secara real-time ke aplikasi di ponsel atau platform kesehatan melalui koneksi Bluetooth Low Energy (BLE) atau Wi-Fi, sehingga memudahkan pemantauan kondisi kesehatan pengguna secara terus-menerus dan akurat. Sensor ini menjadi **alat pemantauan vital** untuk pasien hipertensi, penderita jantung, atau lansia.
Dengan integrasi ke cloud, dokter dapat **melihat tren tekanan darah pasien** tanpa harus melakukan kunjungan langsung.
Data ini juga membantu dalam **deteksi dini** kondisi berbahaya seperti hipertensi kronis atau tekanan darah rendah mendadak.

- **Deskripsi Celah atau Rentan**
  Sensor tekanan darah portabel rentan karena kombinasi faktor teknologi dan operasional: pertama, banyak perangkat memakai protokol nirkabel sederhana atau konfigurasi pairing default yang tidak mengenkripsi payload secara memadai sehingga memungkinkan sniffing paket di udara; kedua, firmware pada sensor seringkali bersifat ringan dan jarang menerima patch keamanan rutin, membuka peluang firmware tampering atau supply-chain attack; ketiga, autentikasi antara sensor dan gateway sering tidak menerapkan mutual authentication sehingga mudah terjadi device spoofing; keempat, cache data sementara di gateway atau aplikasi mobile sering disimpan tanpa enkripsi penuh, sehingga jika ponsel hilang atau terinfeksi malware data sensitif mudah diekstraksi; kelima, keterbatasan sumber daya perangkat (CPU, memori, baterai) membuat implementasi proteksi kriptografis lengkap menjadi sulit pada beberapa model komersial—masalah-masalah inilah yang menjadi perhatian dalam studi-studi tentang sensor fleksibel dan validasi klinis, karena celah teknis ini tidak hanya mengancam privasi tetapi juga integritas pengukuran klinis.

- **Jenis Serangan**
  Dari sisi ancaman teknis, beberapa vektor serangan nyata yang berulang disebutkan di literatur meliputi: MITM (Man-in-the-Middle) di jalur komunikasi BLE/Wi-Fi, di mana paket data pasien dapat disadap atau dimodifikasi; device spoofing, yaitu penyerang memalsukan identitas gateway untuk menerima data sensor; firmware poisoning atau malicious firmware update yang memungkinkan perangkat mengirim data ke endpoint berbahaya atau menjalankan kode tak sah; data injection dan manipulation di titik gateway atau cloud yang mengubah nilai tekanan darah sehingga mengacaukan diagnosis; data exfiltration dari cache lokal atau cloud storage; serta DDoS terhadap gateway/cloud yang dapat

menimbulkan ketidaktersediaan layanan monitoring—semua jenis serangan ini relevan karena berdampak langsung pada privasi pasien dan keselamatan klinis.

- **Serangan yang dilakukan**
  Secara praktis, serangan MITM dapat dilakukan dengan menempatkan peralatan sniffing dan proxy di dekat korban untuk mencegat lalu lintas BLE atau Wi-Fi, menggunakan alat-alat seperti nRF Sniffer atau smartphone dengan kemampuan packet capture; device spoofing dilakukan dengan memalsukan MAC address dan nama perangkat sehingga sensor tertipu untuk pairing ke perangkat attacker; firmware poisoning seringkali memanfaatkan jalur update yang tidak tervalidasi, misalnya server update yang kompromi atau metode update manual melalui file yang di-install oleh teknisi—penyerang mengganti firmware sah dengan firmware yang menyalin atau mengirimkan data ke server mereka; data injection dapat dieksekusi melalui exploit API di gateway atau cloud, memanipulasi payload sebelum disimpan; sementara DDoS dilakukan dengan membanjiri endpoint cloud atau gateway dengan request berlebihan sehingga service degradation terjadi; skenario nyata ini mudah dieksekusi ketika pengembang atau institusi mengabaikan enkripsi end-to-end, key management, dan mekanisme validasi firmware.
- **Aktor & Motivasi Penyerangan**
  Aktor ancaman yang potensial sangat beragam, mulai dari cyber criminals yang mencari data medis bernilai tinggi untuk dijual di pasar gelap atau digunakan dalam penipuan asuransi, hingga insider (pegawai TI atau staf medis) yang menyalahgunakan akses untuk keuntungan pribadi atau penyalahgunaan data; aktor lain bisa berupa industrial spy yang ingin meniru teknologi atau strategi pasar, dan bahkan nation-state actors yang melakukan pengumpulan data biometrik skala besar untuk tujuan intelijen kesehatan atau penelitian yang tidak etis; motivasi tersebut berkisar dari finansial (jual data), politik/spionase, kompetitif (R&D), hingga vandalisme/aktivisme, dan keberadaan berbagai motivasi ini membuat skenario ancaman harus ditangani dengan pendekatan keamanan menyeluruh.

- **Dampak serangan**
  Dampak kompromi sensor mencakup aspek Confidentiality, Integrity, dan Availability: dari sisi *confidentiality*, bocornya data tekanan darah dan identitas pasien dapat menimbulkan pelanggaran privasi, stigma, penyalahgunaan data, serta sanksi hukum bila melanggar regulasi PDP/HIPAA; dari sisi *integrity*, manipulasi data (mis. nilai tekanan darah dimanipulasi naik atau turun) dapat menyebabkan diagnosa yang salah, terapi yang tidak tepat atau penghentian pengobatan yang salah, yang pada gilirannya bisa membahayakan keselamatan pasien; dari sisi *availability*, serangan DDoS atau sabotase gateway dapat menonaktifkan pemantauan vital sehingga pasien kehilangan perlindungan dini; konsekuensi nyata

meliputi salah resep obat, peningkatan mortalitas/morbiditas akibat intervensi medis yang salah, denda regulator bagi institusi, dan rusaknya kepercayaan publik terhadap teknologi wearable kesehatan.

- **Tingkat Keparahan**

  Kemungkinan terjadinya serangan pada sensor pressure portabel dinilai *sedang hingga tinggi* karena sejumlah alasan: perangkat tersebar luas, sering dipakai di lingkungan publik, dan vendor kecil cenderung mengabaikan pembaruan keamanan; sementara keparahan dampaknya dapat berkisar *menengah hingga tinggi* tergantung pada konteks klinis—misalnya pada pasien hipertensi berat atau kondisi kardiovaskular sensitif, manipulasi data kecil saja dapat menimbulkan konsekuensi klinis berat; kombinasi kemungkinan serangan yang relatif mudah dieksekusi (sniffing BLE) dengan potensi dampak yang serius menuntut mitigasi prioritas tinggi sebagaimana ditekankan oleh studi validasi klinis dan tinjauan teknologi wearable.

- **Solusi**
  Mitigasi teknis utama meliputi penerapan enkripsi end-to-end (BLE Secure Connections, TLS 1.3 untuk trafik ke cloud), implementasi mutual authentication antara sensor dan gateway menggunakan certificate-based authentication atau secure element, digital signature untuk setiap paket firmware dan secure boot agar hanya firmware bertanda tangan yang dapat berjalan, enkripsi data at rest pada gateway dan cloud (AES-256) serta penggunaan HSM/Key Management Service untuk pengelolaan kunci; di sisi backend, penggunaan API gateway, OAuth2/mTLS, WAF, rate limiting, dan SIEM untuk deteksi anomali sangat krusial; solusi operasional mencakup kebijakan patching dan update yang wajib, supply-chain security dan code signing pada proses CI/CD, pelatihan staf medis dan teknis tentang praktik pairing aman dan manajemen insiden, serta persyaratan consent & privasi yang ditaati pengguna. Gabungan langkah teknis dan operasional ini diperlukan agar mitigasi tidak hanya teori tetapi juga terimplementasi secara konsisten di lapangan.
- **Trade-off atau Kendala**
  Penerapan mitigasi menghadirkan trade-off nyata: penambahan kriptografi kuat dan secure element meningkatkan konsumsi daya dan ukuran perangkat sehingga berdampak pada masa pakai baterai dan desain produk yang ringkas; biaya produksi dan operasional juga naik karena pembelian modul keamanan, HSM, sertifikasi medis, dan audit keamanan rutin; pengalaman pengguna bisa terganggu oleh proses pairing yang lebih rumit atau autentikasi berlapis yang membuat lansia kesulitan; latency tambahan dari pemeriksaan signature atau enkripsi end-to-end dapat memengaruhi real-time monitoring; dan tidak semua perangkat legacy mendukung update keamanan modern sehingga retrofit menjadi sulit; akhirnya, proses mendapatkan sertifikasi klinis (mis. uji validasi akurasi yang memakan waktu) menjadi beban waktu dan biaya tetapi tetap wajib untuk

memastikan mitigasi keamanan tidak menurunkan kualitas pengukuran klinis.

- **Cara Verifikasi Mitigasi:**
  Verifikasi mitigasi harus melibatkan kombinasi pengujian teknis dan validasi klinis: lakukan penetration testing komprehensif (device↔gateway↔cloud) termasuk simulasi MITM untuk memastikan payload terenkripsi; gunakan BLE packet capture untuk memverifikasi tidak ada PII plaintext, dan lakukan firmware integrity testing (mis. mencoba memuat firmware palsu dan memastikan perangkat menolak berdasarkan signature check); lakukan DDoS/resilience testing pada endpoint cloud dan gateway untuk menguji skenario availability; audit supply-chain dan code signing pipeline untuk memastikan provenance firmware; lakukan clinical validation post-mitigation untuk memastikan perubahan keamanan tidak mengurangi akurasi pengukuran (uji banding vs sphygmomanometer standar seperti yang dianjurkan jurnal validasi klinis); serta adakan tabletop incident response drills dan monitoring log secara berkala melalui SIEM untuk memastikan deteksi dan respons insiden bekerja efektif—semua langkah ini harus didokumentasikan dan dijadwalkan secara berkala sebagai bukti kepatuhan dan efektivitas mitigasi.

d. Earphone Monitor Oksigen (Oxygen Monitoring Earphone)

- **Lokasi Dalam Arsitektur**
  Earphone Sensor → Smartphone (gateway) → Network → Cloud → Monitoring App
  Perangkat ini merupakan **wearable device inovatif** yang menggabungkan fungsi **pemantauan kadar oksigen darah (SpO_2)** dengan **alat audio (earphone)**. Earphone ini menggunakan **sensor optik (photoplethysmography / PPG)** untuk membaca kadar oksigen dalam darah pengguna, lalu mengirimkan hasilnya secara **real-time ke aplikasi kesehatan** di smartphone melalui koneksi **Bluetooth atau Wi-Fi**. Earphone ini membantu pengguna atau tenaga medis untuk: Memantau kadar oksigen darah tanpa alat tambahan, Mendeteksi gejala **hipoksia, stres, atau kelelahan fisik** sejak dini, Memberikan **notifikasi otomatis** ke aplikasi jika kadar oksigen menurun drastis, dan Menjadi **alternatif ringan dan mudah digunakan** dibanding alat medis rumit.
- **Deskripsi Celah atau Rentan**
  Keamanan perangkat earphone berbasis Bluetooth memiliki beberapa celah yang rentan terhadap serangan. Pertama, autentikasi Bluetooth yang lemah karena banyak perangkat tidak menerapkan mekanisme verifikasi perangkat secara ketat sebelum proses pairing, memungkinkan perangkat asing atau palsu untuk melakukan koneksi sebagai spoofed gateway. Kedua, data sensitif seperti SpO_2 yang dikirim tanpa enkripsi kuat dapat dengan mudah disadap melalui jaringan publik atau Wi-Fi tidak aman, membuka peluang pencurian data oleh pihak ketiga. Ketiga, firmware perangkat sering kali tidak dilengkapi dengan sistem verifikasi tanda tangan digital yang memadai, sehingga pembaruan yang palsu atau berbahaya dapat diinjeksikan ke dalam sistem. Selain itu, tidak adanya mekanisme pencatatan aktivitas keamanan seperti logging proses pairing

dan transfer data menyulitkan investigasi apabila terjadi insiden pelanggaran keamanan, sehingga memperburuk risiko kerentanan perangkat secara keseluruhan.

- **Jenis Serangan**
  1) **Bluetooth Sniffing:** Penyadapan data tekanan darah selama transmisi.
  2) **Device Spoofing:** Penyerang meniru smartphone pengguna untuk menerima data dari sensor.
  3) **Data Injection Attack:** Penyerang mengubah nilai tekanan darah sebelum disimpan di cloud, menyebabkan diagnosa salah.
  4) **Firmware Replacement Attack:** Firmware sensor dimodifikasi agar menyimpan atau mengirim data ke server berbahaya.

- **Serangan yang dilakukan**
  Penyerang memanfaatkan perangkat Bluetooth scanner untuk mendeteksi sensor tekanan darah yang aktif di ruang publik, sehingga mereka dapat mengidentifikasi target potensial dengan mudah. Dengan menggunakan teknik MITM (Man-in-the-Middle), penyerang menyusup di antara komunikasi sensor dan smartphone, sehingga semua data tekanan darah yang dikirimkan melewati perangkat mereka terlebih dahulu sebelum sampai ke aplikasi resmi, membuka peluang untuk penyadapan atau bahkan manipulasi data. Selain itu, jika firmware sensor tidak dilengkapi dengan mekanisme validasi seperti checksum, penyerang dapat mengganti firmware asli dengan versi modifikasi yang tersembunyi, yang secara diam-diam mengirim data hasil pengukuran atau informasi lain ke server milik penyerang, sehingga membahayakan keamanan dan keandalan sistem kesehatan digital.

- Aktor & Motivasi Penyerangan
  1) **Criminal hacker:** Mencuri data medis pengguna untuk identitas palsu atau klaim asuransi fiktif.
  2) **Insider attacker:** Pegawai teknis yang memiliki akses ke firmware atau server cloud.
  3) **Industrial spy:** Pihak kompetitor yang ingin menganalisis pola penggunaan perangkat medis digital.

- Dampak serangan
  Kebocoran data tekanan darah pasien ke pihak yang tidak berwenang tidak hanya mengancam kerahasiaan informasi medis pribadi, tetapi juga dapat menimbulkan dampak serius terhadap privasi dan keamanan pasien. Data tekanan darah yang bersifat sensitif ini bisa mengungkap kondisi kesehatan kronis seperti hipertensi atau penyakit jantung, sehingga jika jatuh ke tangan yang salah, dapat digunakan untuk tujuan diskriminasi atau penipuan. Selain itu, apabila data yang digunakan dalam sistem wearable health dimanipulasi atau palsu, dokter dapat melakukan diagnosa yang keliru, yang berakibat pada pemberian pengobatan yang tidak sesuai dan membahayakan keselamatan pasien. Situasi ini juga merusak kredibilitas

teknologi wearable health secara keseluruhan, membuat pasien dan tenaga medis menjadi ragu untuk mengandalkan perangkat tersebut dalam pemantauan kesehatan rutin. Akibatnya, kepercayaan masyarakat terhadap integritas dan keamanan sistem kesehatan digital menurun secara signifikan.

- **Tingkat Keparahan**: Medium to High, Karena perangkat ini lebih jarang diperbarui daripada smartwatch, tetapi berdampak besar pada akurasi data medis dan keselamatan pasien.

- **Solusi**
  1) Gunakan **Bluetooth Low Energy Secure Connections (BLE SC)** dengan enkripsi AES-128/256.
  2) Implementasi **mutual authentication** antara sensor dan smartphone.
  3) Terapkan **digital signature dan checksum validation** untuk setiap pembaruan firmware.
  4) Enkripsi penyimpanan data lokal di aplikasi (AES atau SQLite cipher).
  5) **Pengujian keamanan firmware** sebelum dipasarkan.
  6) **Kebijakan update berkala** dan notifikasi otomatis untuk pengguna.
  7) **Pelatihan petugas kesehatan** agar memahami risiko konektivitas Bluetooth publik.
  8) **Sertifikasi perangkat medis IoT** (misalnya ISO/IEC 80001).

- **Trade-off atau Kendala**
  Menggunakan enkripsi yang kuat pada Bluetooth Low Energy (BLE) memang membuat konsumsi daya baterai pada perangkat menjadi lebih tinggi, sehingga masa pakai baterai berkurang dan perangkat harus lebih sering diisi ulang. Selain itu, proses penyambungan atau pairing antara sensor dan perangkat menjadi lebih rumit karena mekanisme keamanan tambahan yang diterapkan, yang bisa membingungkan atau merepotkan terutama bagi pasien lansia yang kurang terbiasa dengan teknologi atau memiliki keterbatasan dalam menggunakan perangkat digital. Di sisi lain, beberapa perangkat lama punya memori firmware yang terbatas sehingga tidak bisa menyimpan fitur keamanan seperti digital signature, yang berfungsi untuk memastikan firmware asli dan mencegah firmware palsu. Hal ini membuat perangkat lama lebih rentan terhadap serangan atau pemalsuan data karena perlindungan keamanannya kurang maksimal.

- **Cara Verifikasi Mitigasi:**
  Audit koneksi BLE menggunakan packet analyzer adalah proses memonitor dan menganalisis data yang dikirim melalui koneksi Bluetooth untuk memastikan bahwa enkripsi benar-benar diterapkan dan melindungi informasi yang dikirim. Packet analyzer menangkap paket data yang lewat dan memeriksa apakah data tersebut tersandi dengan benar sehingga tidak bisa dibaca oleh pihak tidak berwenang. Untuk memastikan firmware

perangkat tetap utuh dan tidak dimodifikasi, dilakukan pengujian integritas menggunakan checksum berbasis SHA256, yaitu metode kriptografi yang menghasilkan nilai unik dari firmware; jika firmware berubah, nilai checksum juga berubah, menandakan adanya kemungkinan manipulasi. Simulasi serangan MITM dilakukan untuk menguji seberapa tahan sistem terhadap intersepsi data oleh pihak ketiga yang mencoba menyadap atau memanipulasi data selama transmisi antara sensor dan aplikasi. Simulasi ini membantu mencari celah keamanan dan memperkuat langkah perlindungan. Selain itu, monitoring akses cloud secara terus-menerus dilakukan untuk mengawasi semua aktivitas masuk dan keluar data, sehingga jika terdapat pola anomali—misalnya akses tidak biasa atau perubahan data tekanan darah yang mencurigakan—bisa segera dideteksi dan diatasi untuk mencegah potensi kebocoran atau penyalahgunaan data.

e. Glucometer (Sensor Gula Darah Portabel)

- **No / Titik (Component)**
  Pada sistem yang dijelaskan, titik utama adalah **modul pengukuran glukosa/parameter fisiologis yang menggunakan sensor MAX30100** terhubung ke **Arduino Nano** sebagai mikrokontroler pengolah sinyal, kemudian diteruskan melalui **ESP8266** sebagai modul komunikasi Wi-Fi dan akhirnya dikirimkan sebagai notifikasi atau data ke pengguna/tenaga medis lewat **bot Telegram**. Komponen-komponen ini (sensor, MCU, modul Wi-Fi, cloud/bridge Telegram, serta perangkat penerima) bersama-sama membentuk satu titik analisis integral: tiap lapisan mempunyai fungsi spesifik—pengukuran (sensor), pemrosesan awal dan format data (Arduino), konektivitas dan protokol ke jaringan (ESP8266), serta penyampaian/antarmuka (Telegram) — sehingga pengamanan harus mempertimbangkan interaksi antar komponen ini, bukan hanya proteksi pada satu bagian saja.
- **Lokasi Dalam Arsitektur**
  Sensor (MAX30100) → Arduino Nano (ADC + preprocessing) → ESP8266 (Wi-Fi → Internet) → Telegram API / Cloud Bridge → Aplikasi Pengguna (Telegram/HP Dokter)
  Sensor menangkap sinyal biologis, Arduino melakukan penghitungan atau filtering, ESP8266 membuka koneksi ke jaringan lokal/Internet dan mem-post data ke server atau bot Telegram, kemudian penerima (patient/clinician) menerima notifikasi atau log data. Karena jalur ini relatif sederhana dan menggunakan layanan publik (Telegram) sebagai layer aplikasi, tiap transisi (antar-perangkat lokal, dari ESP ke Internet, dan dari cloud ke client Telegram) merupakan titik rentan yang harus dianalisis.
- **Deskripsi Celah atau Rentan**
  Sistem berbasis Arduino + ESP8266 + Telegram rentan karena kombinasi beberapa faktor: pertama, **Arduino Nano** umumnya tidak menyediakan fitur keamanan hardware tingkat lanjut (secure element) sehingga

menyimpan kredensial atau log tanpa proteksi menjadi berisiko; kedua, **ESP8266** sering dikonfigurasi dengan firmware default, Wi-Fi tanpa enkripsi yang kuat atau tanpa otentikasi yang aman terhadap server tujuan sehingga rentan ke MITM; ketiga, penggunaan **Telegram sebagai kanal notifikasi publik** memudahkan integrasi tetapi menempatkan data medis sensitif pada layanan pihak ketiga yang memiliki model privasi tersendiri (pesan tersimpan di cloud Telegram, metadata dapat diakses jika kredensial bocor); keempat, jalur update firmware untuk Arduino/ESP mungkin tidak menggunakan signature sehingga memungkinkan firmware palsu; kelima, implementasi protokol dan parsing data di sisi bot mungkin tidak memvalidasi input sehingga berpotensi dieksploitasi (mis. injection payload). Kombinasi keterbatasan perangkat keras, konfigurasi jaringan lemah, dan ketergantungan pada layanan pihak ketiga itulah yang membuat arsitektur ini rentan.

- **Jenis Serangan**
  Dari arsitektur tersebut, vektor serangan yang paling relevan meliputi: **Man-in-the-Middle (MITM)** antara ESP8266 dan server Telegram atau MQTT broker; **credential leakage** (mis. SSID/Wi-Fi password atau API token Telegram tersimpan dalam firmware tanpa enkripsi); **device takeover** lewat exploit firmware pada ESP8266 atau bootloader Arduino; **data tampering / falsification** (mengubah nilai glukosa yang dikirim sehingga menyebabkan mis-informasi klinis); **data exfiltration** jika attacker berhasil mengunduh log/rekam medis yang tersimpan di perangkat; serta **replay attacks / injection** di level bot (mengirim perintah palsu yang menyebabkan notifikasi false alarm). Semua jenis ini potensial karena kombinasi protokol sederhana dan layanan cloud publik.

- **Serangan yang dilakukan**
  Secara praktis, MITM dapat berlangsung jika ESP8266 terhubung ke jaringan Wi-Fi publik atau tidak dipercaya, dimana attacker menempatkan rogue AP atau ARP spoofing di LAN untuk mencegat lalu lintas dan memodifikasi payload JSON yang dikirim ke Telegram API; credential leakage terjadi bila developer menyimpan token Telegram/SSID sebagai plaintext di kode Arduino yang mudah di-read dari binari firmware; device takeover dapat dilakukan dengan mengakses port serial atau OTA (jika tersedia) untuk mem-flash firmware berbahaya bila akses fisik atau akses jaringan tidak dibatasi; data falsification atau injection mudah dilakukan apabila endpoint cloud tidak memvalidasi source/format data sehingga attacker dapat mem-post nilai palsu dengan token yang dicuri atau meniru request dari ESP8266; replay attack terjadi jika tidak ada mekanisme nonce/timestamp sehingga attacker dapat mengulang pesan lama untuk memicu notifikasi lama. Skenario-skenario ini menunjukkan bahwa serangan bisa dilakukan melalui kombinasi akses fisik sederhana, kelemahan konfigurasi jaringan, dan kredensial yang terekspos.

- Aktor & Motivasi Penyerangan
  Aktor yang mungkin menyerang sistem semacam ini meliputi: **penjahat siber** yang ingin mendapatkan data medis untuk dijual atau melakukan penipuan klaim asuransi; **insider** (mis. teknisi yang memiliki akses fisik) yang menyalahgunakan akses untuk memanipulasi data pasien; **aktivis / vandals** yang ingin mengacaukan layanan telemedis; dan—meskipun kemungkinannya lebih rendah dalam skala kecil—**nation-state atau kelompok riset kompetitif** yang tertarik pada data biometrik. Motivasi meliputi keuntungan ekonomi, sabotase layanan, pengumpulan data untuk kepentingan riset/komersial tanpa izin, atau demonstrasi kelemahan keamanan. Pilihan Telegram sebagai platform publik juga membuka peluang motivasi opportunistik karena barrier to entry relatif rendah.

- **Dampak serangan**
  Jika komponen ini dikompromikan, dampaknya meliputi **Confidentiality**: bocornya data glukosa/identitas pasien akibat ekstraksi log atau interception sehingga terjadinya pelanggaran privasi dan potensi penyalahgunaan data; **Integrity**: manipulasi nilai glukosa dapat menyebabkan diagnosis atau rekomendasi terapi yang salah (mis. pemberitahuan hipoglikemia palsu atau sebaliknya), yang berakibat nyata pada keselamatan pasien seperti pemberian insulin yang tidak perlu; **Availability**: serangan DDoS atau sabotase gateway (ESP8266) dapat menghentikan aliran data sehingga tenaga medis kehilangan pemantauan real-time; konsekuensi nyata termasuk kesalahan pengobatan, penundaan intervensi medis, kerugian reputasi institusi yang mengadopsi sistem, serta potensi sanksi hukum bila terjadi pelanggaran perlindungan data.

- **Tingkat Keparahan**: Medium to High, Karena mengingat teknologi yang dipakai (Arduino Nano dan ESP8266) banyak dipakai dalam prototipe dan solusi low-cost serta sering dit-deploy di lingkungan yang kurang terkelola secara ketat, **kemungkinan serangan tergolong sedang hingga tinggi**, terlebih bila perangkat terpasang di jaringan publik atau dikonfigurasi tanpa kontrol akses. **Keparahan dampak** juga berkisar dari sedang hingga tinggi: bagi pasien non-kritis dampak mungkin berupa alarm palsu atau ketidaknyamanan, tetapi bagi pasien diabetes rentan dampak manipulasi data dapat berujung pada bahaya medis serius, sehingga skenario worst-case menuntut mitigasi yang kuat.

- **Solusi**
  Secara teknis, langkah pertama adalah **melindungi kredensial**: jangan menyimpan token Telegram atau password SSID sebagai plaintext dalam kode; gunakan mekanisme secure storage (mis. EEPROM terenkripsi atau external secure element) dan rotasi token berkala. Pastikan komunikasi ESP8266-ke-backend menggunakan **TLS/HTTPS** (jika menggunakan webhook/REST API) atau gunakan broker MQTT yang mendukung TLS

dan client certificate; implementasikan **message signing** atau HMAC pada payload sehingga server dapat memverifikasi asal data; aktifkan **mutual authentication** antar node bila memungkinkan; sediakan **mechanisme OTA aman** dengan signature verification sehingga firmware palsu ditolak; batasi akses fisik ke perangkat dan nonaktifkan debug/serial bila tidak perlu; dan jika menggunakan Telegram sebagai kanal, pastikan hanya metadata non-sensitif dikirim lewat Telegram, atau gunakan enkripsi end-to-end pada payload sensitif (mis. kirim hanya hash/ID di Telegram, sedangkan data lengkap di server yang terotentikasi). Secara operasional, terapkan kebijakan patching rutin, prosedur secure provisioning saat instalasi (unique credentials per device), pelatihan teknisi instalasi, dan kebijakan privacy consent bagi pengguna yang jelas. Jika memungkinkan, migrasikan kanal notifikasi ke layanan kesehatan yang memenuhi standar privasi nasional atau HIPAA-like untuk data sensitif.

- **Trade-off atau Kendala**
  Implementasi mitigasi di atas membawa trade-off: menambahkan enkripsi dan mekanisme signing meningkatkan kompleksitas pengembangan dan dapat menambah beban CPU/memori pada ESP8266 sehingga mempengaruhi stabilitas/throughput; secure storage atau secure element menambah biaya perangkat dan mungkin tidak tersedia pada desain low-cost; implementasi TLS pada ESP8266 memerlukan library yang cukup berat sehingga mengurangi memori heap untuk fungsi lain; penggunaan layanan notifikasi yang compliant (bukan Telegram) memerlukan biaya operasional dan integrasi tambahan serta mungkin mengurangi kemudahan penggunaan untuk pasien/petugas; pembatasan data yang dikirim ke Telegram (mengurangi sensitifitas) dapat mengurangi kegunaan aplikasi bagi tenaga medis yang menginginkan akses cepat; serta kebijakan patching dan rotasi token menuntut manajemen lifecycle yang disiplin yang kadang sulit dilakukan di lingkungan lapangan.

- **Cara Verifikasi Mitigasi:**
  Verifikasi mitigasi harus bersifat teknis dan praktis: lakukan **penetration testing** pada rantai end-to-end (sensor → Arduino → ESP8266 → Internet) termasuk simulasi MITM di jaringan Wi-Fi lokal untuk memastikan payload terenkripsi dan signature valid; uji coba **firmware integrity** dengan mencoba-coba mem-flash firmware tak bertanda tangan dan memastikan perangkat menolak; lakukan **code review** untuk memastikan tidak ada kredensial hardcoded; lakukan **capture packet** (Wireshark / TLS inspector) untuk memverifikasi tidak ada PII dikirim tanpa enkripsi; uji kebijakan rotasi token dan recovery flow jika token dikompromikan; audit penggunaan Telegram bot (pastikan token bot aman dan pesan sensitif tidak terkirim) serta lakukan **tabletop incident response drill** untuk skenario data leak atau device takeover; terakhir, verifikasi klinis pasca-mitigasi juga penting — pastikan mekanisme keamanan tidak mengganggu kualitas data pengukuran (mis. latency

enkripsi tidak mengubah sampling rate) dengan melakukan perbandingan pengukuran terhadap standar klinis.

DAFTAR PUSTAKA

Hidayani, W. R., & Santosa, A. F. (2024). Wearable IoT dalam Bidang Kesehatan: Tantangan dan Peluang. *Bincang Sains dan Teknologi*, *3*(02), 78-84.

Isyanto, H., Wahid, A. S., & Ibrahim, W. (2022). Desain Alat Monitoring Real Time Suhu Tubuh, Detak Jantung dan Tekanan Darah secara Jarak Jauh melalui Smartphone berbasis Internet of Things Smart Healthcare. *RESISTOR (Elektronika Kendali Telekomunikasi Tenaga Listrik Komputer)*, *5*(1), 39-48.

Suruso, S., Simanjuntak, L., & Hesti, E. (2023). Penerapan Internet Of Things Dalam Rancang Bangun Telemedis Kadar Glukosa. *Jurnal RESISTOR (Rekayasa Sistem Komputer)*, *6*(2), 49-56.

Utomo, M. F. (2024). Eksplorasi Peran Smartwatch Android berbasis IoT dalam Bidang Kesehatan. *Jurnal Sains Dan Teknologi 4.0*, *1*(2), 15-23.