



Universidad Nacional Autónoma de México
Facultad de Ingeniería



Carrera: Ingeniería en Computación

Curso: Sistemas Operativos

Profesor: Ing. Gunnar Eyal Wolf Iszaevich

Alumn@:

Montiel Juarez Oscar Ivan

Torres Delgadillo Samuel Mixcoatl

Fecha de entrega: 1 de Abril del 2025

Semestre: 2025-2

1. Introducción al Edge Computing

Edge Computing: Ventajas y Desafíos

Ventajas:

- Baja latencia: Procesamiento local reduce latencia hasta 84.1% vs cloud logrando <10ms en servidores cercanos (<30 km) [1,2,3].
- Ahorro de ancho de banda: Solo envía datos relevantes a la nube [4,5]. Para 2025, la mayoría de datos se procesarán en edge [6].
- Seguridad: Menor exposición de datos sensibles vs cloud [7,8,9].

Desafíos:

- Escalabilidad: Complejo gestionar miles de dispositivos heterogéneos [8,10]. Balanceo de carga tradicional no siempre funciona [11].
- Recursos limitados: Dispositivos edge tienen poca potencia y almacenamiento [12]. Asignación óptima de recursos es NP-hard [13].
- Seguridad: Más vulnerabilidades por distribución (ej. malware, ataques físicos) y dificultad para actualizar dispositivos [14].

2. Componentes Clave

1. Dispositivos Edge:

- a. Sensores, cámaras y dispositivos IoT con capacidad limitada de procesamiento [15]
- b. Ejecutan sistemas operativos ligeros (Ej: Raspbian, Ubuntu Core) para procesamiento inicial

2. Nodos Edge:

- a. Gateways o micro centros de datos con mayor potencia computacional [15,16]
- b. Sistemas operativos especializados (Ej: K3s, Fedora IoT) para:
 - i. Filtrado de datos
 - ii. Agregación de múltiples fuentes
 - iii. Ejecución de contenedores ligeros (Docker, Kubernetes)

3. Cloud Central:

- a. Gestión centralizada del sistema [8]
- b. Compatible con sistemas operativos tradicionales (Linux/Windows Server)

3. Flujo de Datos y Requerimientos del SO

1. Captura: Dispositivos edge generan datos usando SOs en tiempo real (RTOS)

como FreeRTOS o Zephyr)

2. Procesamiento Local:
 - a. SOs optimizados para baja latencia (Ej: Azure Sphere OS)
 - b. Protocolos ligeros: MQTT/CoAP [17]
3. Transmisión a Cloud:
 - a. TCP/IP sobre SOs convencionales [8]
 - b. Encriptación obligatoria (TLS/SSL integrado en el SO)

Aspectos Críticos para SOs Edge:

1. Baja latencia: Kernels optimizados (Ej: Linux con parches PREEMPT_RT)
2. Seguridad: Actualizaciones OTA y sandboxing (Ej: SELinux en nodos edge) [8]
3. Eficiencia energética: Gestión de recursos en SOs embebidos

4. Requisitos Clave para SOs Edge

1. Eficiencia de recursos:
 - a. Mínima huella en memoria y CPU (critical en dispositivos limitados) [12]
 - b. Ejemplos: Kernels modulares (microkernels) y soporte para hibernación profunda
2. Escalabilidad distribuida:
 - a. Gestión centralizada de flotas de dispositivos heterogéneos [8]
 - b. Soporte nativo para redes mesh y actualizaciones OTA
3. Tolerancia a fallos:
 - a. Mecanismos de autoreparación y operación offline [8]
 - b. Transaccionalidad en updates (ej: Ubuntu Core)
4. Seguridad reforzada:
 - a. Sandboxing obligatorio (snaps/flatpaks) [14]
 - b. Encriptación hardware-aware (TPM/HSM)
5. Soporte a heterogeneidad:
 - a. Compatibilidad multi-architecture (ARM/x86/RISC-V) [7]
 - b. Drivers unificados para protocolos diversos (MQTT, LoRaWAN)

5. SOs Edge vs Tradicionales

| característica | SOs Edge | SOs Tradicionales |
|----------------|-----------------------|----------------------------|
| Arquitectura | Microkernel/Unikernel | Monolítica (linux/windows) |
| Footprint | 10-100MB (Riot OS) | 1GB+ |

| | | |
|-----------------|-----------------------|-------------------------|
| Latencia | us-ms (RTOS) | ms-s |
| Actualizaciones | Atómicas(a/B updates) | Paquetes convencionales |

Ejemplos Técnicos:

1. Azure IoT Edge: Containers asilados + offline operation
2. RIOT OS: RTOS para dispositivos <64KB RAM
3. Ubuntu Core: Snapcraft + updates transaccionales

6. Gestión de Recursos en Edge

Desafíos:

1. Asignación dinámica en hardware heterogéneo [12,18]
2. Balanceo de carga en redes inestables [11]

Soluciones Técnicas:

1. Kubernetes Edge: K3s para clusters distribuidos
2. Unikernels: Compilación estática de apps + solo librerías esenciales
3. Virtualización Ligera: Firecracker (AWS) para microVMs seguras

7. Seguridad en Edge Computing

Riesgos Clave:

1. Ataques físicos: Dispositivos en ubicaciones inseguras (ej: manipulación de sensores industriales) [8]
2. Vulnerabilidades distribuidas:
 - a. DDoS contra nodos edge [14]
 - b. Side-channel attacks en hardware limitado
 - c. Inyección de malware por protocolos IoT

Estrategias a Nivel de SO:

1. Cifrado eficiente:
 - a. Algoritmos lightweight (ChaCha20-Poly1305) para datos en tránsito/reposo [8,14]
 - b. Soporte hardware para aceleración AES/ECC
2. Autenticación robusta:
 - a. Protocolos basados en ECC para dispositivos limitados [14]

- b. Zero-trust architecture con MFA
- 3. Actualizaciones OTA seguras:
 - a. Firmware firmado criptográficamente
 - b. Particiones A/B para rollback automático [8]
- 4. Hardening del kernel:
 - a. Syscall filtering (seccomp-bpf)
 - b. Memory-safe languages (Rust) para drivers

8. Casos de Uso Reales

| Aplicación | Requisitos Técnicos | Sos Utilizados |
|--------------------|--------------------------------------|------------------------------|
| IIoT | RTOS para predic maintenance | FreeRTOS, Zephyr |
| Vehículo Autónomos | Latencia <10ms, soporte GPU/TPU | QNX, ROS 2 |
| Telemedicina | HIPAA-compliance, cifrado end to end | Azure sphere os, ubuntu core |
| Smart cities | Escalabilidad masiva | RIOT OS, OpenWrt |

9. Tendencias Futuras

- 1. 5G + Edge:
 - a. SOs con stack de red optimizado (kernel bypass con DPDK) [4]
- 2. Edge AI:
 - a. Kernels con soporte nativo para TPUs (TensorFlow Lite Micro) [7]
- 3. SOs especializados:
 - a. Unikernels para cargas de trabajo específicas (ej: inferencia ML)
 - b. Microkernels certificados (seL4 en entornos críticos)
- 4. Blockchain en Edge:
 - a. Light clients para consenso distribuido [18]
 - b. Smart contracts ejecutados en nodos edge

10. Conclusiones

Los SOs edge deben equilibrar:

- 1. Baja latencia (<1ms en casos críticos)

2. Seguridad by design (TEEs, enclaves SGX)
3. Gestión eficiente de recursos (memory ballooning, CPU pinning)

Áreas de Oportunidad:

1. Schedulers para heterogeneidad hardware
2. Debugging distribuido en entornos edge
3. Standardización de APIs para gestión remota

Fuentes citadas

1. Akamai, "Edge Computing Versus Cloud Computing: Key Similarities and Differences," Akamai Blog, Mar. 26, 2025. [Online]. Available: <https://www.akamai.com/blog/edge/edge-computing-versus-cloud-computing-key-similarities-differences>
2. X. Mengwei, "IWQoS23-edge-latency.pdf," xumengwei.github.io, Mar. 26, 2025. [Online]. Available: <https://xumengwei.github.io/files/IWQoS23-edge-latency.pdf>
3. "Measuring Latency Reduction and the Digital Divide of Cloud Edge Datacenters," ResearchGate, Mar. 26, 2025. [Online]. Available: https://www.researchgate.net/publication/369974733_Measuring_Latency_Reduction_and_the_Digital_Divide_of_Cloud_Edge_Datacenters
4. "IJSRA-2024-2082.pdf," ijsra.net, Mar. 26, 2025. [Online]. Available: <https://ijsra.net/sites/default/files/IJSRA-2024-2082.pdf>
5. Arm, "What Is Edge Computing (Versus Cloud Computing)?," Arm Glossary, Mar. 26, 2025. [Online]. Available: <https://www.arm.com/glossary/edge-computing-vs-cloud-computing>
6. "The Role of Edge Computing in Manufacturing: Enhancing Network Performance and Decision-Making," coevolve.com, Mar. 26, 2025. [Online]. Available: <https://www.coevolve.com/insights-the-role-of-edge-computing-in-improving-network-performance-and-business-decisions/>
7. "Edge computing," Wikipedia, Mar. 25, 2025. [Online]. Available: https://en.wikipedia.org/wiki/Edge_computing
8. Stratus Technologies, "What is Edge Computing | Why We Need Edge," [stratus.com](https://www.stratus.com/edge-computing/), Mar. 26, 2025. [Online]. Available: <https://www.stratus.com/edge-computing/>
9. "Location Privacy Protection in Edge Computing: Co-Design of Differential Privacy and Offloading Mode," MDPI, vol. 13, no. 13, 2025. [Online]. Available: <https://www.mdpi.com/2079-9292/13/13/2668>
10. S. A. Khan, et al., "ScalEdge: A framework for scalable edge computing in Internet of...," SAGE Journals, vol. 15501477211035332, Mar. 26, 2025. [Online]. Available: <https://journals.sagepub.com/doi/10.1177/15501477211035332?icid=int.sj-fu>
I-text.similar-articles.7
11. "Revolutionizing load harmony in edge computing networks with...," PMC, Mar. 26, 2025. [Online]. Available: <https://pmc.ncbi.nlm.nih.gov/articles/PMC11779838/>
12. "EDGE COMPUTING: EVOLUTION, CHALLENGES, AND FUTURE DIRECTIONS," ResearchGate, Mar. 26, 2025. [Online]. Available: https://www.researchgate.net/publication/380341600_EDGE_COMPUTING_EVOLUTIO

N CHALLENGES AND FUTURE DIRECTIONS

13. "Joint Task Offloading and Resource Allocation in Heterogeneous....," NSF, Mar. 26, 2025. [Online]. Available: <https://par.nsf.gov/servlets/purl/10450429>
14. "A Survey on Edge Computing (Ec) Security Challenges....," Preprints.org, vol. 202502.1500, Mar. 26, 2025. [Online]. Available: <https://www.preprints.org/manuscript/202502.1500/v1>
15. "Edge Computing vs. Cloud Computing: Key Differences in 2024....," Jessup University Blog, Mar. 26, 2025. [Online]. Available: <https://jessup.edu/blog/engineering-technology/edge-computing-vs-cloud-computing/>
16. "Edge Computing," Microsoft Research, Mar. 26, 2025. [Online]. Available: <https://www.microsoft.com/en-us/research/project/edge-computing/>
17. "What Is Edge Computing? | Microsoft Azure," Azure, Mar. 26, 2025. [Online]. Available: <https://azure.microsoft.com/en-us/resources/cloud-computing-dictionary/what-is-edge-computing>
18. "Optimizing Resource Management in 5G Heterogeneous Edge....," ResearchGate, Mar. 26, 2025. [Online]. Available: https://www.researchgate.net/publication/389415591_Optimizing_Resource_Management_in_5G_Heterogeneous_Edge_Computing_with_Blockchain_and_Deep_Learning