Gustavo Daniel Cortez

Brugge, Belgium | gdcg0021@outlook.com | linkedin.com/in/gdaniel-cortez | github.com/danfromdablock | dcortez.vercel.app

SUMMARY

Hands-on engineer working across cloud infrastructure, security, and AI automation. Production experience provisioning cloudspaces and VMs on commercial IaaS, deploying and securing on-premise LLMs on dedicated GPU infrastructure, and building end-to-end N8N workflows with prompt-injection hardening. Cybersecurity background covers full-scope penetration testing, ELK-based threat monitoring, and vulnerability assessment. Comfortable bridging offensive security knowledge with operational engineering work. Open to roles in DevOps, Cloud, DevSecOps, AI Engineering, or Cybersecurity.

TECHNICAL SKILLS

DevOps & Cloud: Docker, Linux (Ubuntu, Kali), VMware, OPNsense, WireGuard VPN, CI/CD, Bash scripting, Infrastructure as Code, IaaS provisioning (whitesky.cloud)

Kubernetes: Foundational hands-on experience with container orchestration concepts and tooling

AI & ML Operations: On-premise LLM deployment (GPUStack, Ollama), N8N workflow automation, RAG pipelines, Qdrant vector database, prompt engineering, AI agent security testing

Networking & Security: Firewall rule design, network segmentation, VPN administration, Metasploit, Nmap, Burp Suite, OWASP ZAP, Wireshark, Pass-the-Hash, ARP spoofing, RCE exploitation

Monitoring & Forensics: Elasticsearch (ELK Stack), Kibana, Filebeat, ModSecurity log analysis, honeypot deployment (Cowrie)

Programming: Python, JavaScript, Bash, SQL, Flask, Flask-SocketIO; familiarity with C# (.NET), Java, PHP

Cloud Platforms: whitesky.cloud (production daily); AWS familiarity (academic / project level)

EXPERIENCE

AI Automation and Cloud Operations Intern

whitesky.cloud | February 2026 – May 2026 | Belgium

- Provisioned and ran cloudspaces on whitesky.cloud's IaaS platform: multi-VM Linux setups, internal networking, storage allocation, and Docker container deployment for production AI workloads.
- Wrote Python automation scripts against whitesky.cloud's API to provision cloudspaces, spin up VMs, create users, and manage infrastructure resources without manual console work.
- Integrated third-party APIs into N8N workflows: Rocket.Chat for team notifications and AI-assisted reply drafting, Zammad for ticket ingestion and updates, and Meneja for pulling G8 platform healthcheck data. Built the request/response handling, error checks, and authentication flow for each integration.
- Built a production AI ticketing workflow in N8N that connects Zammad, Rocket.Chat, PostgreSQL, and Qdrant. The agent reads incoming tickets, drafts a customer reply, and routes the response based on a confidence score. Estimated 40% reduction in manual L1 support workload.
- Built a second N8N workflow that polls the Meneja API every 5 hours, pulls failed healthchecks across managed G8 platforms, runs each one through an AI agent against a GitLab knowledge base of fix procedures, and posts severity-sorted reports to Rocket.Chat.
- Managed a fleet of Docker containers (n8n, Ollama, Qdrant, PostgreSQL, pgAdmin) on Linux VMs. Day-to-day work included container health checks, log inspection, restart procedures, and disk-usage cleanup.
- Deployed Qwen 3 30B on GPUStack on a dedicated GPU VM, kept separate from the application VMs. All inference runs internally so ticket data never leaves the company network.
- Set up an OPNsense firewall with a first-match LAN ruleset and a whitelist-based isolation model. The AI VM only accepts inbound traffic from N8N and cannot make outbound connections to the public internet.
- Configured WireGuard VPN on OPNsense with per-user /32 tunnel IPs and split-tunnel options. Backend services (Ollama, Qdrant, PostgreSQL, pgAdmin) are reachable only through VPN. The N8N web interface is the single hardened entrypoint exposed to the public internet.

- Hardened the AI ticketing agent against prompt injection: restricted system-prompt leakage, cross-ticket data access, and destructive command generation through prompt rules and strict JSON output schemas. Ran adversarial test payloads against the agent (jailbreak prompts, instruction overrides, payloads embedded in ticket bodies and image attachments) to verify it refused unsafe actions. Added magic-byte validation on file attachments to confirm declared file types.
- Wrote Python and Bash scripts for deployment, log correlation, and incident response. Documented system architecture, maintenance procedures, and troubleshooting playbooks for the operations team.

PROJECTS

Full-Scope Penetration Test – NDL Vulnerability Assessment

- Ran a penetration test across VPN, DMZ, and internal network segments. Identified 5 Critical, 4 High, and 4 Medium vulnerabilities.
- Exploited XML-RPC deserialization RCE, Elasticsearch MVEL script RCE, Pass-the-Hash, SNMPv1 misconfiguration, and RDP credential interception to reach SYSTEM-level access.
- Performed lateral movement and subnet pivoting using Metasploit, Impacket, and credential harvesting. Demonstrated a full attack chain from initial access to domain persistence.
- Delivered an audit report with CVSS scoring, CWE references, network topology maps, and a prioritized remediation roadmap.

Web Security Monitoring and Honeypot Analysis

- Reviewed 2,000+ ModSecurity alerts and categorized them by attack type (SQL injection, XSS, RCE). Identified the top 3 patterns and used them to tighten WAF rules.
- Correlated Filebeat severity data to surface critical system errors. Mean time to detection dropped by around 30%.
- Deployed a Cowrie SSH honeypot that captured 500+ brute-force login attempts and SQLMap scans. Wired it up to Elasticsearch and Kibana for live threat dashboards.

Advanced Nmap Scanner – Python CLI Tool

- Built a Python CLI wrapper around Nmap. Supports OS fingerprinting, custom NSE scripts, decoy scanning, and normalized output (JSON, XML, grepable) for use in downstream security tools.

EncryptedChatBackEnd – Python, Flask-SocketIO

- Real-time encrypted messaging backend supporting 100 concurrent users. Uses AES-256 encryption, Eventlet for async I/O, and environment-based secret management.

ProductionPetrolForecast – Python, Flask

- Petroleum production forecasting models (Exponential, Hyperbolic, Harmonic decline analysis) wrapped in a Flask web interface for dataset upload and Excel report generation.

EDUCATION

Bachelor of Applied Computer Science, Cybersecurity Professional

Howest University of Applied Sciences

High School Diploma

British School Quito

LANGUAGES

English (Proficient) · Spanish (Proficient)