亚马逊云科技

把龙虾用到极致

软件企业 OpenClaw 云上实战进阶指南

下篇（进阶篇）

Contents.

把龙虾用到极致——软件企业 OpenClaw 云上实战进阶指南

下篇（进阶篇）

Contents.

把龙虾用到极致——软件企业 OpenClaw 云上实战进阶指南

下篇（进阶篇）

Contents.

把龙虾用到极致——软件企业 OpenClaw 云上实战进阶指南

下篇（进阶篇）

ABSTRACT

摘要

在第一期《把龙虾部署到云上：软件企业 OpenClaw 入门指南》中，我们探讨了如何将开源 AI 助手框架 OpenClaw 快速部署到亚马逊云科技，实现零密钥安全架构与多模型智能路由。然而，当 OpenClaw 从原型走向生产环境，成为服务于成千上万用户的"数字员工"时，软件企业将面临全新的挑战：如何防范恶意插件注入？如何实现多租户隔离？如何让 AI 助手拥有跨会话的"全景记忆"？如何为不同工作负载选择最合适的大模型？

本期实战进阶指南将深入探讨 OpenClaw 在生产环境中的三大核心命题：安全治理、架构扩展与认知增强。我们将结合亚马逊云科技的最佳实践，为您揭示如何构建一个安全、可扩展且具备深度业务理解能力的企业级 AI Agent 底座。

NO.1

从原型到生产的"鸿沟"——软件企业不得不面对的双重风险

OpenClaw 凭借其丰富的多渠道集成（Telegram、Slack、Discord 等 12 个以上消息通道）和强大的本地优先架构，在开发者社区中迅速走红。然而，在企业级应用场景中，它面临着开源软件与 AI Agent 的双重安全风险。

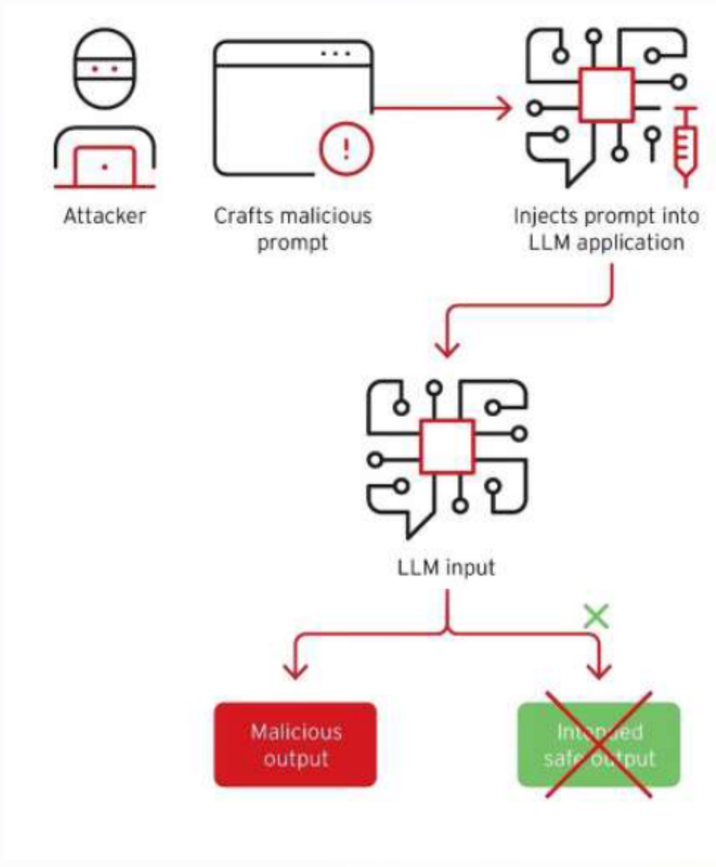
1.1 暴露在外的"数字员工"

根据 SecurityScorecard 的研究，超过数万个 OpenClaw 实例可通过互联网直接访问，且数量仍在快速增加。这些缺乏防护的实例极易成为攻击者的目标。Oasis Security 曾发现一个关键漏洞（CVE-2026-25253）：攻击者可通过跨站 WebSocket 劫持（Cross-Site WebSocket Hijacking）绕过认证，获得设备的完全控制权。

截至 2026 年 3 月，OpenClaw 在 CVE.org 中共记录 81 个 CVE 漏洞，Critical 及 High 级别漏洞占比超过半数。

1.2 恶意插件与供应链投毒

OpenClaw 的强大在于其 ClawHub 技能市场，但这也引入了严重的供应链风险。Koi Security 在 2026 年 2 月的研究中发现，ClawHub 中存在超过 820 个恶意 Skills，较几周前激增 142%。这些恶意插件可能伪装成合法的 Git 助手，暗中窃取浏览器会话 Cookie、外泄敏感文档，甚至植入后门实现持久化控制。



开源AI供应链攻击流程图
（图源：Trend Micro）

1.3 提示词注入与过度授权

AI 代理无法区分用户真实意图与恶意构造的提示词。Zenity Labs 的研究表明，攻击者可以通过在网页或文档中嵌入隐藏指令，诱导 OpenClaw 执行非授权操作。更危险的是，OpenClaw 通常以高权限运行，其默认权限涵盖文件系统、Shell 命令、浏览器和网络的完整控制权，一旦发生意图理解偏差，可能导致灾难性后果。

1.4 OWASP Agentic AI 十大安全威胁

OWASP 专门针对 Agentic AI 应用发布了十大安全威胁框架，涵盖 Agent 目标劫持、工具滥用、身份与权限滥用、供应链漏洞、非预期代码执行、记忆与上下文投毒、不安全的 Agent 间通信、连锁失败、人机信任滥用以及异常 Agent 等十个维度。软件企业在将 OpenClaw 投入生产时，必须将这一框架作为安全设计的基础参考。

NO.2

构建企业级安全防线——纵深防御策略

面对上述威胁，软件企业必须摒弃"单点防护"思维，转向构建覆盖网络、身份、数据与运行时的纵深防御体系。亚马逊云科技的安全最佳实践将这一体系分为七个层次：

防护层次	核心措施	亚马逊云科技服务
网络层	VPC 私有子网 + 7 个 VPC 端点 + NAT 网关	Amazon VPC
API 层	3 条显式路由 + 限流（50/100 请求/秒）	Amazon API Gateway
认证层	Telegram Secret Token + Slack HMAC-SHA256	Amazon Cognito
授权层	最小权限 IAM + 按资源 ARN 限定	Amazon IAM
数据层	KMS CMK 加密 + Secrets Manager 密钥管理	Amazon KMS / Secrets Manager
隔离层	每用户独立 microVM + S3 命名空间隔离	AgentCore Runtime
审计层	CloudTrail 全量审计 + CloudWatch 告警	CloudTrail / CloudWatch

2.1 运行时隔离与最小权限原则

不要让 OpenClaw 直接运行在宿主机上。亚马逊云科技建议使用隔离的环境运行 OpenClaw，通过配置 tools.allow/deny 策略，仅允许调用必要的工具。对于危险操作（如删除数据、发送邮件），必须在 OpenClaw 外部实现强制的人工确认机制，防止因上下文窗口压缩导致的"指令遗忘"问题。

2.2 统一身份与鉴权网关

当 OpenClaw 作为企业级应用提供服务时，必须解决身份传递问题。通过引入 AgentCore Gateway，结合 Amazon Cognito 或企业现有的身份提供商（Auth0、Entra ID、Okta 等），可以实现细粒度的访问控制。OpenClaw 在访问外部系统时，应传播最终用户的身份（OAuth Bearer Token），避免使用高权限的固定服务账号，从而防范混淆代理（Confused Deputy）攻击。

2.3 多层防注入与内容过滤

在提示词处理环节，引入 Amazon Bedrock Guardrails，在数据输入和模型输出阶段提供多层防护，有效拦截恶意提示词注入，并过滤敏感数据，确保 AI 代理的行为符合企业合规要求。

1.4 OWASP Agentic AI 十大安全威胁」

禁止员工直接从 ClawHub 安装未经审核的 Skills。软件企业应建立自己的私有 Skills 仓库，对每个 Skills 进行安全扫描和代码审查后再入库。对于可疑的 Skills，应在沙箱环境中运行并分析其行为，确认安全后方可推广。

NO.3

架构跃迁——OpenClaw on Amazon Bedrock AgentCore

为了彻底解决 OpenClaw 在生产环境中的扩展性与隔离性问题，亚马逊云科技推出了基于 Amazon Bedrock AgentCore 的增强架构，提供将 Agent 投入生产所需的完整能力集：Runtime、Memory、Identity、Gateway、Code Interpreter、Browser、Observability、Policy 和 Evaluations。

3.1 Serverless 弹性伸缩与会话隔离

传统的单体部署难以应对突发流量，且存在多租户数据串扰风险。AgentCore Runtime 采用与 Amazon Lambda 相同的 Firecracker microVM 技术，为每个用户会话提供独立的沙箱环境：

完全隔离	每个会话拥有独立的 CPU、内存和文件系统，用户 A 与用户 B 的数据在物理层面完全隔离。
按需伸缩	冷启动时间控制在 5 秒以内，最长运行 8 小时，会话结束后自动清理，零残留。
STS 临时凭证	每个用户会话使用独立的 STS Session Policy，实现细粒度的权限隔离。

3.2 状态持久化与生命周期管理

Serverless 容器是短暂的，而 OpenClaw 的状态（MEMORY.md 对话记忆、USER.md 用户画像、AGENTS.md 指令、SOUL.md 个性配置）需要持久化。该架构通过 Amazon S3 实现工作空间的实时同步：启动时从 S3 恢复状态，运行期间每 5 分钟周期性保存，容器收到 SIGTERM 信号时在 10 秒内完成最终保存。

同时，结合 Amazon EventBridge Scheduler，可以实现独立于容器生命周期的定时任务管理。用户只需用自然语言表达"每天早上 7 点提醒我查邮件"，系统即可自动创建持久化的定时规则，确保任务的可靠执行。

3.3 协议桥接与无缝集成

AgentCore 提供了强大的代理机制，解决了三个关键的工程难题：

协议不匹配

AgentCore 使用 HTTP 接口（GET /ping, POST /invocations），而 OpenClaw 使用控制平面原生协议。通过 `contract.js` 实现 HTTP→WebSocket 桥接与懒加载编排。

API 格式不兼容

OpenClaw 输出 OpenAI Chat Completions 格式，而 Amazon Bedrock 要求 ConverseStream 格式（结构完全不同）。通过 `proxy.js` 实现实时转译，软件企业无需修改 OpenClaw 核心代码。

启动时间与健康检查

OpenClaw 启动约需 4 分钟（插件注册），而健康检查必须秒级响应。通过懒加载策略，先返回 Healthy 状态，首次调用时才启动 OpenClaw，实现两者的兼顾。

NO.4

为每种工作负载选对模型——Amazon Bedrock 多模型路由实战

OpenClaw 的核心优势之一是支持多模型路由。在亚马逊云科技上，软件企业可以通过 Amazon Bedrock 接入丰富的模型生态，为不同类型的工作负载选择最合适的模型，在性能与成本之间取得最优平衡。

4.1 五款主力模型特性速览

模型	核心定位	关键特性	适用场景
Kimi K2.5	原生多模态 Agent 模型	视觉与语言理解，支持图像输入的 Agent 工具调用；Agent Swarm 多 Agent 协同执行	UI 自动化、视觉工作流、多 Agent 协作任务
Zhipu AI GLM 4.7	新一代编码伙伴	多语言 Agentic 编码，支持 Claude Code、Kilo Code、Cline 等主流框架；HLE 基准得分 42.8%（+12.4%）	代码生成、UI 生成与 Vibe Coding、复杂推理
MiniMax M2.5	全栈开发与 Agent 工作流	多编程语言全覆盖（Rust、Java、Go、C++、iOS、Android）；在 Claude Code、Cline、Kilo Code 等主流 Agent 框架中表现稳定	Web 与 App 全栈开发、复杂交互与高质量可视化
Qwen3 Coder Next	高效能 Agent 编码模型	仅 3B 激活参数（80B 总参数），性能媲美 10–20 倍参数量的模型；256K 上下文，支持长程推理与失败恢复	动态编码任务、复杂工具调用、主流 IDE 集成
DeepSeek-V3.2	高效能混合推理模型	DeepSeek Sparse Attention（DSA）机制，大幅降低长序列处理的计算复杂度；专为 Agentic 工作流设计	复杂推理、代码生成、Agent 多步骤工具使用、学术计算

4.2 多模型路由策略建议

软件企业在构建 OpenClaw 产品化方案时，可以参考以下路由策略：

按任务类型路由

将简单的问答类任务（如 FAQ 查询、状态查询）路由至参数量较小、成本较低的模型；将需要复杂推理或代码生成的任务路由至高性能模型。这一策略可将 AI 调用成本降低 60%–80%。

按用户等级路由

为付费高级用户分配高性能模型（如 Kimi K2.5 的多模态能力），为免费用户分配高性价比模型（如 Qwen3 Coder Next 的稀疏激活架构）。

按数据敏感度路由

涉及企业核心数据的任务，优先路由至在 Amazon Bedrock 上运行的模型，充分利用 IAM 零密钥架构和数据不离境的合规优势。

NO.5

认知增强——从"扁平记忆"到"全景记忆网"

企业级 Agent 光有安全的底座还不够，它必须足够"聪明"。传统的 OpenClaw 往往受限于"扁平记忆"：跨会话历史断裂、记忆冲突无法解决、核心线索被噪声淹没。

5.1 重塑"数字海马体"

亚马逊云科技提出了一种"全景记忆"架构，将记忆分为四个层次：工作记忆（临时会话）、短期记忆（近期行为模式）、长期记忆（深度用户理解）和核心记忆（永久身份特征）。

左脑向量抓语义 利用 Amazon OpenSearch 或 Amazon S3 Vector，采用双 LLM 架构进行过滤与决策，将自然语言精准转化为多维向量，深刻理解上下文语境与意图差异。

右脑图谱推因果 借助 Amazon Neptune Analytics 构建实体关系图谱，跳出平面思维，实现多维推理和复杂事务的因果溯源。

5.2 智能 Skill 路由：告别"Token 刺客"

当 OpenClaw 挂载了 50 个以上的 Skills 时，传统的"全量注入"方式会导致 API 账单爆炸，并严重拖慢大模型响应速度。通过基于向量相似度的动态技能路由，系统可以在运行时透明拦截用户提问，仅挑出最相关的 3–5 个 Skills 喂给模型：

Token 节省约 90%	提示词消耗从约 2000 Token 锐减至约 200 Token。
缓存命中延迟 50ms	运行时透明拦截，用户完全无感。
精准 Skill 匹配	告别 50 个以上 Skills 的盲目全量注入。

5.3 乐高式 Serverless 数据底座

软件企业可以根据自身业务场景，灵活组合亚马逊云科技的 Serverless 数据服务：

服务	适用场景	核心特点
Amazon OpenSearch	需要混合搜索（语义+关键词）、大规模向量数据	混合检索能力、丰富分析功能
Amazon S3 Vector	PB 级冷数据存储、成本敏感型 AI 应用	超低存储成本、无限扩展性
Amazon Aurora PostgreSQL+pgvector	熟悉 PG 技术栈、多租户 SaaS 应用	PostgreSQL 生态兼容、统一数据模型
Amazon Neptune Analytics	GraphRAG 应用、需要可解释性的 AI 推理	图原生推理、超高性能、一体化查询
Amazon ElastiCache / MemDB for Valkey	实时推荐、高并发向量召回（万级 QPS）	极致低延迟、实时索引更新

5.4 全景记忆实战场景

以下是一个典型的企业级 Agent 工作场景，展示了全景记忆架构的实际价值：

客户群里突然发来一条语音："上次那个电商项目现在要上价格推荐模块，预算还按老规矩，让之前那个技术负责人尽快出个方案！"

在不到一秒的时间里，OpenClaw 的底层发生了以下动作：

1. **智能 Skill 路由：**从 50 多个 Skills 中只抽出「项目文档库」「CRM 报表」「企业日程协同」3 个相关 Skill 喂给大模型。
2. **语义记忆提取：**从过去闲聊中提取客户核心记忆（"老规矩"）：客户对成本敏感，偏好 Serverless 免运维方案。
3. **图谱多跳推理：**"上次电商项目" → 当时对接负责人是 Bob → Bob 上周已调离 → Bob 走前重点引荐专家 Carol 负责推荐系统。
4. **自动执行决策：**将老项目的架构底稿和历史预算基线同步给 Carol，并自动核对 Carol 空闲档期，发送会议通知。

NO.6

总结与行动路线图

将 OpenClaw 投入生产环境是一项系统工程。亚马逊云科技为软件企业提供了从基础设施到认知增强的完整蓝图：

阶段	核心任务	关键服务
第一步：小范围试用	在受控环境中验证 OpenClaw 的核心业务价值，选定 1–2 个高价值场景	Amazon EC2 / Lightsail
第二步：构建安全底座	实施网络隔离、统一身份网关、最小权限策略和运行时监控	VPC、IAM、Cognito、GuardDuty
第三步：夯实供应链管理	建立企业私有 Skills 仓库，严格审查第三方插件，纳入信息安全治理体系	Amazon Security Hub
第四步：架构规模化	迁移至 AgentCore 多租户架构，实现 Serverless 弹性伸缩与状态持久化	Amazon Bedrock AgentCore
第五步：认知增强	引入全景记忆与智能 Skill 路由，配置多模型路由策略，持续优化成本与性能	Amazon OpenSearch、Amazon S3 Vector、Amazon Neptune

立即行动

访问亚马逊云科技官网，获取 OpenClaw on AgentCore 的完整参考架构与部署模板，或联系您的亚马逊云科技客户经理安排一对一的解决方案咨询。

GitHub 参考仓库：https://github.com/aws-samples/sample-host-openclaw-on-amazon-bedrock-agentcore