

【Sentry】自架主機分析

當你要自建 Sentry 並希望個別安裝各個服務（非 all-in-one Docker 方式），需要了解 Sentry 背後實際上是一個**分散式系統**，包含多個子服務。以下是簡要說明每個服務的用途，並提供 **最小部署需求建議（幾台機器）**。

□ Sentry 核心元件說明

服務	用途	是否必要	說明
Sentry Web	Web UI + API	□ 必要	使用者操作介面與事件接入口。
Worker	背景任務處理	□ 必要	發送通知、儲存事件、標記等。Celery 背後用 Redis 當 broker。
Cron	定期任務排程	□ 必要	例如清除舊資料、定期檢查等。
Relay	邊界接收器	□ 建議	前端 SDK 最先接觸的入口，可作為 proxy、限流、預處理。
PostgreSQL	資料庫	□ 必要	儲存大部分結構化資料（專案、使用者、事件等）。
Redis	訊息佇列、快取	□ 必要	Celery 任務排程、快取使用。
Kafka	事件串流	□ 必要（新版）	訊息串流 backbone，傳輸異常事件。
Zookeeper	Kafka 依賴	□ Kafka 必要	Kafka metadata 管理工具。
ClickHouse	高速查詢	□ 建議	儲存事件 metadata，提供 Discover/Performance UI。
Symbolicator	原始碼還原符號	□ 可選	用於原生 crash report（iOS/Android 原始符號對應）。
Snuba	查詢引擎	□ 必要	Sentry 的查詢服務，轉發查詢給 ClickHouse。
nginx	Proxy	□ 建議	當作入口，轉發請求給 Sentry、Relay 等。

□ 最小分布式部署建議（分機版本）

節點	安裝項目	說明
Node 1：App Server	sentry-web + cron + worker	最好有 2 核以上，主要承擔使用者 UI 與邏輯
Node 2：資料庫	PostgreSQL + Redis	可整合在同台，需穩定，重要資料所在
Node 3：訊息處理	Kafka + Zookeeper	Kafka 可與其他服務共用機器（但建議獨立）
Node 4：查詢引擎	Snuba + ClickHouse	查詢速度核心，可以跟 Kafka 共用
Node 5：Symbolicator + Relay	可再分出獨立機器	可選，針對原生應用及 SDK proxy

□ **最小建議：**
最少可以從 **3 台機器** 起步：

- **機器 A (App)**：sentry-web、cron、worker、relay
- **機器 B (Storage)**：PostgreSQL、Redis、Symbolicator
- **機器 C (Streaming)**：Kafka、Zookeeper、ClickHouse、Snuba

這樣就可以支撐基本功能。當負載上來之後，再做橫向擴展。

□ 補充：資源建議（單位：最小）

服務	CPU	Memory	磁碟
PostgreSQL	2C	4GB+	SSD 儲存，100GB 起跳
ClickHouse	2C	4GB+	SSD 快速查詢，空間依事件量調整
Kafka	2C	4GB	10GB 起

服務	CPU	Memory	磁碟
Snuba	1C	2GB	低
Redis	1C	1GB	幾乎沒磁碟需求
Web / Worker	2C	2~4GB	看使用者數

📄 小結

想法	建議
只是要「快速玩玩看」	👉 Docker self-hosted (1 台)
內網 demo、準生產環境	👉 拆 3 台機器，參考上表
正式環境、日誌量大	👉 至少 5 台，分出 ClickHouse 與 Kafka
已有 K8s 架構	👉 用 Helm Chart 部署

🔄修訂版本 #2

★由 treeman 建立於 15 🕒🕒🕒 2025 15:07:53

🔧由 treeman 更新於 15 🕒🕒🕒 2025 15:20:09