**1) نظرة عامة (Architecture)**

* **واجهة الدخول (Nginx)**: عكس عكسي (Reverse Proxy) على نفس الـ VPS، يوجّه الطلبات إلى الخدمات الداخلية.
* **المراقبة والـ Observability**:
  + **Prometheus** + **Alertmanager** للتجميع والتنبيه.
  + **Grafana** للداشبوُرد.
  + **Loki** + **Promtail** للّو́جات.
  + **Tempo** + **OTel Collector** للتتبّع (Tracing).
* **الأخطاء (Error Tracking)**: **GlitchTip** (بديل Sentry) على errors.kaleem-ai.com.
* **الخدمات الإدارية**: n8n، Mongo-Express، Redis-Commander، إلخ (مع BasicAuth/حصر الوصول).
* **الفرونت**: يرسل **Traces** إلى otel.kaleem-ai.com ويبلّغ الأخطاء إلى errors.kaleem-ai.com عبر DSN.

**2) خريطة الدومينات (DNS & Routing)**

كل السجلات من نوع **A** تُشير إلى IP الـ VPS.

| **الدومين** | **الوظيفة** | **الـ Upstream** | **ملاحظات أمنية** |
| --- | --- | --- | --- |
| api.kaleem-ai.com | API (NestJS) | 127.0.0.1:3000 | WebSocket مفعّل |
| errors.kaleem-ai.com | GlitchTip (Sentry self-hosted) | 127.0.0.1:8085 | شهادة SSL مشتركة |
| grafana.kaleem-ai.com | Grafana | 127.0.0.1:3005 | للمشرفين فقط |
| prom.kaleem-ai.com | Prometheus | 127.0.0.1:9090 | للمشرفين فقط |
| alerts.kaleem-ai.com | Alertmanager | 127.0.0.1:9093 | للمشرفين فقط |
| traces.kaleem-ai.com | Tempo UI | 127.0.0.1:3200 | للمشرفين فقط |
| otel.kaleem-ai.com | OTLP HTTP Collector | 127.0.0.1:4318 (مسار /v1/traces) | CORS مفعّل على Nginx |
| n8n.kaleem-ai.com | n8n | 127.0.0.1:5678 | WebSocket + CORS |
| n8n2.kaleem-ai.com | n8n ثانية | 127.0.0.1:5679 | WebSocket |
| mongo-express.kaleem-ai.com | Mongo Express | 127.0.0.1:8081 | BasicAuth |
| redis-cmd.kaleem-ai.com | Redis Commander | 127.0.0.1:8082 | BasicAuth |
| cdn.kaleem-ai.com | ملفات ثابتة + Proxy لـ MinIO تحت /documents/ | ملفات ثابتة من /var/www/musaidbot-cdn + 31.97.155.167:9000 | راجع ملاحظات التوقيع |

**3) الشهادات (SSL)**

* أصدرنا **شهادة واحدة متعددة الدومينات (SAN)** باستخدام Certbot **standalone**.
* موقع الشهادة المشتركة المستخدمة في جميع السيرفر بلوكس:
* ssl\_certificate /etc/letsencrypt/live/api.kaleem-ai.com/fullchain.pem;
* ssl\_certificate\_key /etc/letsencrypt/live/api.kaleem-ai.com/privkey.pem;
* certbot قام بضبط **التجديد التلقائي** (systemd timer). يمكن الاختبار:
* sudo certbot renew --dry-run

**4) إعدادات Nginx (ملخّص)**

**4.1 الملف الرئيسي /etc/nginx/nginx.conf (أهم الأسطر)**

* خرائط للـ CORS وWebSocket + تفعيل Gzip من snippet:

map $http\_origin $allow\_origin {

default "";

"~^https?://(localhost:5173|app\.kaleem-ai\.com|kaleem-ai\.com)$" $http\_origin;

}

map $http\_upgrade $connection\_upgrade { default upgrade; '' close; }

include /etc/nginx/snippets/gzip.conf;

include /etc/nginx/conf.d/\*.conf;

include /etc/nginx/sites-enabled/\*;

**4.2 سنِبتس (مرة واحدة)**

* /etc/nginx/snippets/proxy-headers.conf:

proxy\_http\_version 1.1;

proxy\_set\_header Host $host;

proxy\_set\_header X-Real-IP $remote\_addr;

proxy\_set\_header X-Forwarded-For $proxy\_add\_x\_forwarded\_for;

proxy\_set\_header X-Forwarded-Proto $scheme;

proxy\_read\_timeout 3600;

proxy\_send\_timeout 3600;

proxy\_buffering off;

* /etc/nginx/snippets/websocket.conf:

proxy\_set\_header Upgrade $http\_upgrade;

proxy\_set\_header Connection $connection\_upgrade;

* /etc/nginx/snippets/security-headers.conf:

add\_header X-Frame-Options "SAMEORIGIN" always;

add\_header X-Content-Type-Options "nosniff" always;

add\_header Referrer-Policy "strict-origin-when-cross-origin" always;

add\_header X-XSS-Protection "1; mode=block" always;

* /etc/nginx/snippets/gzip.conf:

gzip on;

gzip\_comp\_level 5;

gzip\_types text/plain text/css application/json application/javascript application/xml+rss application/xml image/svg+xml;

gzip\_min\_length 1024;

**4.3 مثال Server Block قياسي (https + redirect)**

استخدمناه لكل الدومينات مع اختلاف server\_name و proxy\_pass.

# HTTP → HTTPS

server {

listen 80;

server\_name example.kaleem-ai.com;

return 301 https://$host$request\_uri;

}

# HTTPS

server {

listen 443 ssl http2;

server\_name example.kaleem-ai.com;

ssl\_certificate /etc/letsencrypt/live/api.kaleem-ai.com/fullchain.pem;

ssl\_certificate\_key /etc/letsencrypt/live/api.kaleem-ai.com/privkey.pem;

include /etc/letsencrypt/options-ssl-nginx.conf;

ssl\_dhparam /etc/letsencrypt/ssl-dhparams.pem;

include /etc/nginx/snippets/security-headers.conf;

include /etc/nginx/snippets/proxy-headers.conf;

# (اختياري) CORS ديناميكي لخدمات محددة:

# add\_header Access-Control-Allow-Origin $allow\_origin always;

location / {

proxy\_pass http://127.0.0.1:PORT; # غيّر PORT

}

}

**4.4 حالات خاصة**

* **WebSocket** (API و n8n):
* location ^~ /api/kleem/ws {
* include /etc/nginx/snippets/proxy-headers.conf;
* include /etc/nginx/snippets/websocket.conf;
* proxy\_pass http://127.0.0.1:3000;
* }
* **OTLP Collector** (CORS مسموح لـ POST/OPTIONS فقط):
* server {
* listen 443 ssl http2;
* server\_name otel.kaleem-ai.com;
* # ... (ssl + includes)
* add\_header Access-Control-Allow-Origin $allow\_origin always;
* add\_header Access-Control-Allow-Methods "POST, OPTIONS" always;
* add\_header Access-Control-Allow-Headers "Content-Type, Accept, User-Agent, Traceparent, Baggage, Authorization" always;
* if ($request\_method = OPTIONS) { return 204; }
* location /v1/traces {
* include /etc/nginx/snippets/proxy-headers.conf;
* proxy\_pass http://127.0.0.1:4318;
* }
* }
* **Mongo-Express/Redis-Commander** (BasicAuth):
* auth\_basic "Restricted";
* auth\_basic\_user\_file /etc/nginx/.htpasswd-mongo; # أو .htpasswd-redis

**5) GlitchTip (Sentry self-hosted)**

**5.1 لماذا حدث 502 وكيف أصلحناه؟**

* كانت EMAIL\_URL غير صحيحة (console:) → سببت crash.
* أصلحناها إلى SMTP Hostinger مع **ترميز** كلمة المرور في URL (لوجود رموز خاصة).
* تبيّن أن GlitchTip يستمع داخليًا على **:8080** وليس :8000 → عدّلنا الـ port mapping إلى:
* ports: ["8085:8080"]
* بعدها أصبح curl http://127.0.0.1:8085 → **200 OK** واشتغل Nginx.

**5.2 صِيغة EMAIL\_URL (Hostinger SSL 465)**

انتبه لترميز كلمة المرور!

EMAIL\_URL="smtp://support@kaleem-ai.com:<PASSWORD\_URL\_ENCODED>@smtp.hostinger.com:465?ssl=True"

DEFAULT\_FROM\_EMAIL="support@kaleem-ai.com"

لترميز الباسورد:

python3 -c "import urllib.parse; print(urllib.parse.quote\_plus('كلمة-المرور-الأصلية'))"

**5.3 تهيئة قاعدة البيانات**

docker exec -it <glitchtip-container> ./manage.py migrate

docker exec -it <glitchtip-container> ./manage.py createsuperuser

**5.4 أخذ DSN للفرونت**

* من لوحة GlitchTip → Project → **Client Keys (DSN)**
* الشكل:
* https://<public\_key>@errors.kaleem-ai.com/<project\_id>

**6) الفرونت (Front-end)**

**6.1 متغيرات البيئة (Vite)**

ضعها في .env.production:

VITE\_API\_URL=https://api.kaleem-ai.com

VITE\_SENTRY\_DSN=https://<public\_key>@errors.kaleem-ai.com/<project\_id>

VITE\_OTEL\_COLLECTOR\_URL=https://otel.kaleem-ai.com/v1/traces

**6.2 OTel في المتصفح**

* المصدّر (OTLP HTTP) يرسل إلى VITE\_OTEL\_COLLECTOR\_URL.
* تأكد من:
* const exporter = new OTLPTraceExporter({
* url: import.meta.env.VITE\_OTEL\_COLLECTOR\_URL, // https://otel.kaleem-ai.com/v1/traces
* });
* فعلنا CORS على Nginx لـ otel.kaleem-ai.com.

**6.3 Sentry (GlitchTip)**

* استخدم DSN من GlitchTip وفعّل tracing:
* Sentry.init({
* dsn: import.meta.env.VITE\_SENTRY\_DSN,
* integrations: [Sentry.browserTracingIntegration()],
* tracesSampleRate: 0.2,
* environment: import.meta.env.MODE,
* release: "kaleem-frontend@1.0.0",
* });

**7) الـ Observability Stack**

**7.1 Prometheus**

* يغرف (scrape) من:
  + prometheus:9090, kaleem-cadvisor:8080, kaleem-redis-exporter:9121, kaleem-mongodb-exporter:9216
  + خدماتك: kaleem-api:3000, العمال (ai-reply/webhook/indexing) على مسارات /metrics
  + otel-collector:8889
* احتفظنا بقواعد تنبيه (Alerts) مثل:
  + APIHighErrorRate — 5xx > 5%
  + RabbitMQQueueBacklog
  + InstanceHighCPU
  + (تحقق من العناوين/الأسماء لتطابق خدماتك الفعلية)

**7.2 Alertmanager**

* إرسال تنبيهات إلى **Telegram Bot** (انقل التوكن خارج Git، ودوّر التوكن بعد النشر).
* تأكد من عنوان Alertmanager في prometheus.yml:
* alertmanagers:
* - static\_configs:
* - targets: ['kaleem-alertmanager:9093']

**7.3 Grafana**

* Datasources:
  + Prometheus: http://prometheus:9090
  + Loki: http://loki:3100
  + **Tempo**: http://tempo:3200 ← أضفناه ليظهر التتبّع.
* وفّر داشبوُرد:
  + لوكي Logs
  + cAdvisor CPU/Mem
  + Prometheus Alerts
  + Tempo traces (Explore)

**7.4 Loki + Promtail**

* Promtail يقرأ لوجات docker:  
  mounts: /var/lib/docker/containers, docker.sock
* تظهر في Grafana → Explore → Loki.

**7.5 Tempo + OTel Collector**

* Tempo يخزّن spans محليًا مع احتفاظ 48h (قابل للزيادة).
* OTel Collector يستقبل من الفرونت على otel.kaleem-ai.com/v1/traces ويدفع إلى Tempo.
* CORS handled على Nginx.

**8) الأمان (Security)**

* **جدار ناري**: افتح خارجيًا فقط 80/443:
* sudo ufw allow 80,443/tcp
* sudo ufw deny 3000 8085 3005 9090 9093 4318 3200 3100 5678 5679 8081 8082/tcp
* **BasicAuth** للخدمات الإدارية:
* sudo apt-get install -y apache2-utils
* sudo htpasswd -bc /etc/nginx/.htpasswd-mongo admin StrongPass123
* sudo htpasswd -bc /etc/nginx/.htpasswd-redis admin StrongPass123
* **الأسرار (Secrets)**: لا تحفظ في Git. استخدم ملفات .env على السيرفر أو مخزن أسرار.
* **توكن تيليجرام**: بدّل التوكن بعد الإطلاق.
* **تبويب IP / VPN** للخدمات الحساسة (Grafana/Prometheus/Alertmanager/Tempo UI) إذا أمكن.

**9) النسخ الاحتياطي (Backups)**

أهم الـ volumes:

* glitchtip\_pg (Postgres)
* loki\_data, tempo\_data, prom\_data, grafana\_data

مثال باك-أب Postgres يومي:

docker exec -t musad-bot-n8n-glitchtip-postgres-1 pg\_dump -U glitchtip glitchtip \

| gzip > /var/backups/glitchtip\_$(date +%F).sql.gz

**10) تشغيل/إيقاف/اختبار (Runbook سريع)**

**عند النشر الأول:**

1. أضف سجلات DNS.
2. أنشئ سنِبتس وخرائط Nginx.
3. استصدر شهادة SAN:

sudo systemctl stop nginx

sudo certbot certonly --standalone \

-d api.kaleem-ai.com -d errors.kaleem-ai.com -d grafana.kaleem-ai.com -d prom.kaleem-ai.com \

-d alerts.kaleem-ai.com -d otel.kaleem-ai.com -d traces.kaleem-ai.com \

-d n8n.kaleem-ai.com -d n8n2.kaleem-ai.com -d mongo-express.kaleem-ai.com -d redis-cmd.kaleem-ai.com -d cdn.kaleem-ai.com \

--preferred-challenges http --agree-tos -m you@example.com --no-eff-email

sudo systemctl start nginx

1. حدّث musaibot.conf لاستخدام الشهادة المشتركة.
2. sudo nginx -t && sudo systemctl reload nginx.

**GlitchTip**:

* تأكد من EMAIL\_URL الصحيح (Password URL-encoded).
* ports: ["8085:8080"]
* migrate + createsuperuser.

**اختبارات سريعة**:

curl -I http://127.0.0.1:8085 # GlitchTip 200/302

curl -I https://errors.kaleem-ai.com # عبر Nginx

curl -I -X OPTIONS https://otel.kaleem-ai.com/v1/traces -H "Origin:https://kaleem-ai.com"

**11) ماذا غيّرنا اليوم (Change Log)**

* إصلاح **EMAIL\_URL** في GlitchTip من console: إلى SMTP Hostinger مع **ترميز** كلمة المرور.
* تعديل **port mapping** لـ GlitchTip إلى 8085:8080 لأن uWSGI يستمع على 8080.
* إصدار شهادة **SAN واحدة** وتوحيد مسارات الشهادة في كل server blocks.
* إضافة خرائط map في nginx.conf (CORS/WebSocket) وإزالة تكرار Gzip.
* إزالة تكرار server\_name (تحذير n8n2 على 80).
* تصحيح WebSocket path kleem (بدلاً من klaeem).
* تمكين CORS لـ otel.kaleem-ai.com عبر Nginx.
* إضافة **Tempo** كـ Grafana datasource.
* تثبيت ممارسات أمان: BasicAuth للأدوات الإدارية + UFW.

**12) أسئلة شائعة (FAQ)**

**Q1: ما هو بريد الدخول لـ GlitchTip؟**  
A: تنشئه بنفسك عبر createsuperuser. ليس هو نفسه بريد SMTP.

**Q2: لماذا استخدمنا شهادة واحدة لكل الدومينات؟**  
A: شهادة SAN تغطي عدة أسماء، أسهل إدارة وتجديد، والمتصفحات تقبلها طالما الاسم موجود ضمن الشهادة.

**Q3: لماذا 502؟**  
A: غالبًا الـ upstream لا يرد (خدمة متوقفة/بورت خاطئ). حلّها إما بتصحيح proxy\_pass أو تشغيل الخدمة.

**Q4: كيف أربط الفرونت؟**  
A: ضع:

VITE\_SENTRY\_DSN=<dsn من GlitchTip>

VITE\_OTEL\_COLLECTOR\_URL=https://otel.kaleem-ai.com/v1/traces

**13) قوائم تحقق (Checklists)**

**نشر خدمة جديدة خلف Nginx**

* أضف سجل DNS.
* أضف server blocks (80 → https, 443).
* استخدم الشهادة المشتركة.
* حدّد proxy\_pass الصحيح.
* nginx -t && systemctl reload nginx.

**تحقيق 502**

* curl -I http://127.0.0.1:<port> على الهوست.
* docker ps وتأكد من الـ port mapping.
* docker logs <service> للأخطاء.
* تأكد من الـ firewall.

**مراقبة**

* Grafana يقرأ من Prometheus/Loki/Tempo.
* Alertmanager يرسل تيليجرام.
* Logs تظهر في Loki (Explore).
* Traces تظهر في Tempo.