

# Содержание

<b>VAS Experts ePDG Monitoring System</b> .....	3
<b>Комплексная система мониторинга шлюза VoWiFi (ePDG)</b> .....	3
<b>1. Обзор решения</b> .....	3
Ключевые преимущества .....	3
<b>2. Архитектура системы мониторинга</b> .....	3
Четырёхуровневая архитектура наблюдаемости .....	4
<b>3. Компоненты и метрики</b> .....	4
Покрытие мониторингом .....	4
Количественный обзор по категориям .....	4
Принципы именования .....	5
<b>4. Перечень метрик</b> .....	5
4.1 Config (2) .....	5
4.2 Network (1) .....	6
4.3 IKEv2 SWu (3) .....	6
4.4 GTPv2-C S2b (4) .....	6
4.5 GTP-U data plane (3) .....	6
4.6 Diameter SWm/SWx/S6b (5) .....	6
4.7 Service KPI (4) .....	7
4.8 Session State (4) .....	7
4.9 Application (3) .....	7
4.10 System (4) .....	7
Типы метрик (напоминание) .....	7
<b>5. Интерфейсы интеграции</b> .....	7
5.1 Prometheus (CNCF Standard) .....	8
5.2 SNMP v2c — EPDG-MIB .....	8
5.3 Grafana .....	9
5.4 Alertmanager Webhooks .....	9
<b>6. Система алармов</b> .....	9
Категории алармов .....	9
Полный перечень алармов (20+ правил) .....	9
Процесс обработки алармов .....	10
Особенности .....	10
<b>7. Визуализация и операционные дашборды</b> .....	10
Состав дашбордов .....	10
Дизайн для NOC .....	11
<b>8. Интеграция в единый стек EPC Monitoring</b> .....	11
<b>9. Покрытие метрик по уровням OSI</b> .....	11
Детализация метрик по уровням .....	12
Уровень 9: Качество восприятия сервиса VoWiFi .....	12
<b>10. Стандарты и совместимость</b> .....	13
<b>11. Модель развёртывания</b> .....	13
Характеристики развёртывания .....	13
Варианты размещения .....	14
<b>12. Конфигурация экспортёра метрик</b> .....	14
<b>13. План развития системы мониторинга</b> .....	14
13.1 Расширение перечня метрик .....	14
13.2 Планируемые функциональные расширения .....	15
13.3 Интеграционный roadmap .....	16

<b>14. Заключение</b> .....	16
Следующие шаги .....	16

# VAS Experts ePDG Monitoring System

## Комплексная система мониторинга шлюза VoWiFi (ePDG)

### 1. Обзор решения

Система мониторинга VAS Experts ePDG Monitoring обеспечивает полный операционный контроль компонента **fast-epdg** — шлюза VoWiFi (Voice over WiFi), реализующего эволюционированный пакетный шлюз данных (Evolved Packet Data Gateway) согласно 3GPP TS 29.273 и TS 24.302. Шлюз обеспечивает защищённую передачу голосового и пакетного трафика через нелицензируемые частоты Wi-Fi с IPSec/IKEv2 туннелированием и интеграцию с EPC-ядром через интерфейсы SWu, SWm, SWx, S2b, S6b.

Решение предоставляет единую платформу наблюдаемости для оперативных служб мобильного оператора — от уровня IPSec SA (L3 security) до KPI абонентского опыта VoWiFi (L9 QoE).

### Ключевые преимущества

- **Мониторинг в реальном времени** — обновление метрик каждые 10-15 секунд, непосредственное отображение состояния IKE SA / Child SA и GTP-туннелей в NOC-дашбордах без отложенной агрегации.
- **Проактивное обнаружение аномалий** — 20+ алармов с автоматической эскалацией по severity. Недоступность PGW/AAA, деградация IKEv2-латентности, рост ошибок EAP-AKA' — детектируются до того, как абоненты заметят проблемы со звонками.
- **Открытые интерфейсы интеграции** — Prometheus, SNMP v2c (EPDG-MIB под официальным enterprise OID VAS Experts .1.3.6.1.4.1.43823 / поддерево ePDG .43823.1), Alertmanager webhooks, Grafana provisioning. Интеграция в существующую NMS/OSS инфраструктуру без vendor lock-in.
- **Zero external dependencies на plugin-уровне** — встроенный /metrics endpoint в fast-epdg, без Java, без JMX, без внешних агентов. Publish в формате Prometheus text exposition поверх HTTP.
- **Покрытие всего стека SWu→S2b** — IKEv2 (SWu), Diameter SWm/SWx/S6b, GTPv2-C (S2b) и GTP-U data plane — в одном endpoint. 33 метрики суммарно, покрывают control plane и data plane.

### 2. Архитектура системы мониторинга

```
flowchart TB
    subgraph DataPlane ["Data Plane"]
        IPSEC["IPSec ESP IKEv2 SA / Child SA Kernel xfrm"]
        GTPU["GTP-U Tunneller S2b Data"]
    end
    subgraph ControlPlane ["Control Plane"]
        EPDG_PGW["ePDG ↔ PGW"]
        IKEv2_SWu["IKEv2 SWu"]
    end
```

```

EAP-AKA' auth"] DIAM["Diameter Client
SWx/SWm/S6b"] GTPC["GTPv2-C S2b
to PGW/SMF"] CTRL["ePDG Controller
Attach/Detach FSM"] end subgraph Collection["Сбор метрик"] PROMEXP["fast-epdg
/metrics endpoint
:9817"] end subgraph Storage["Хранение"] PROM["Prometheus
TSDB
15-day retention"] end subgraph Visualization["Визуализация"] GRAF["Grafana
4 дашборда, 35+ панелей"] end subgraph Alerting["Алармирование"] AM["Alertmanager
Routing / Inhibition"] EMAIL["Email SMTP"] SNMPGW["SNMP Trap Sender
Webhook → Trap gateway"] NMS["Внешняя NMS
SNMP v2c UDP/162"] WH["Webhooks
Telegram / PagerDuty"] end IKE --> PROMEXP IPSEC --> PROMEXP GTPC --> PROMEXP GTPU -->
PROMEXP DIAM --> PROMEXP CTRL --> PROMEXP PROMEXP --> PROM PROM --> GRAF PROM --> AM
AM --> EMAIL AM --> SNMPGW SNMPGW --> NMS AM --> WH

```

### Четырёхуровневая архитектура наблюдаемости

Уровень	Компонент	Технология
Сбор (Collection)	Встроенный /metrics endpoint fast-epdg	Prometheus text exposition поверх HTTP
Хранение (Storage)	Prometheus TSDB	Локальное хранение, 15-дневная ретенция по умолчанию
Визуализация (Visualization)	Grafana + JSON provisioning	Автозагрузка 4 дашбордов
Алармирование (Alerting)	Alertmanager + SNMP Trap Sender	PromQL rules → webhook → SNMP v2c trap

## 3. Компоненты и метрики

### Покрытие мониторингом

```

flowchart LR
  EXP["fast-epdg
/metrics :9817"] --> CFG["Config
2 метрики"]
  EXP --> NET["Network
1 метрика"]
  EXP --> PROTO["Protocols L5-L7
15 метрик"]
  EXP --> SVC["Service KPI
4 метрики"]
  EXP --> SESS["Session State
4 метрики"]
  EXP --> APP["Application
3 метрики"]
  EXP --> SYS["System
4 метрики"]
  PROTO --> IKEV2["IKEv2
SWu — 3"]
  PROTO --> GTPC["GTPv2-C
S2b — 4"]
  PROTO --> GTPU["GTP-U
S2b data — 3"]
  PROTO --> DIA["Diameter
SWm/SWx/S6b — 5"]

```

### Количественный обзор по категориям

Категория	Кол-во метрик	Интервал опроса	Ключевые показатели
-----------	---------------	-----------------	---------------------

Категория	Кол-во метрик	Интервал опроса	Ключевые показатели
<b>Config</b>	2	10 с	Статус конфигурации, счётчик reload
<b>Network</b>	1	10 с	Статус соединений к peers (PGW/AAA/HSS)
<b>IKEv2 (SWu)</b>	3	10 с	Сообщения по типам (IKE_SA_INIT, IKE_AUTH, CREATE_CHILD_SA), latency histogram, ошибки
<b>GTPv2-C (S2b)</b>	4	10 с	Сообщения (Create/Modify/Delete Session), latency, ошибки, retransmissions
<b>GTP-U data plane</b>	3	10 с	Packets/bytes, tunnel ошибки
<b>Diameter (SWm/SWx/S6b)</b>	5	10 с	Сообщения по command code (DER/DEA, MAR/MAA, AAR/AAA), latency, ошибки, watchdog, connection status
<b>Service KPI</b>	4	10 с	Attach success rate, attach duration histogram, service availability, uptime
<b>Session State</b>	4	10 с	IKE SA, Child SA, GTP sessions, total subscribers
<b>Application</b>	3	10 с	Threads total, memory, лог-сообщения по уровням
<b>System</b>	4	10 с	CPU usage, memory, disk usage, открытые FD
<b>Итого</b>	<b>33 метрики</b>		

## Принципы именования

Все метрики имеют префикс `epdg_` и организованы по иерархии:

```
epdg_
├── config_*           # Конфигурация
├── network_*         # Сетевой уровень
├── ikev2_*           # SWu (IKEv2/IPSec)
├── gtp_*             # S2b control-plane GTPv2-C
├── gtpu_*            # S2b data-plane GTP-U
├── diameter_*        # SWm/SWx/S6b
├── service_*         # KPI сервисов (attach, availability, uptime)
├── session_*         # Состояние сессий (IKE SA, Child SA, GTP,
subscribers)
├── app_*             # Метрики приложения (memory, threads, logs)
└── system_*         # Системные метрики (CPU, disk, network)
```

## 4. Перечень метрик

Все метрики экспортируются через единый `/metrics` endpoint в формате Prometheus text exposition. Именование следует конвенции Prometheus: `epdg_<группа>_<имя>[_unit]`, тип Counter имеет суффикс `_total`, Histogram — суффикс `_seconds` / `_bytes`.

### 4.1 Config (2)

Имя	Тип	Назначение
epdg_config_status	Gauge	Статус конфигурации компонента (0=error, 1=ok)
epdg_config_reload_total	Counter	Счётчик конфигурационных reload (success/failure)

#### 4.2 Network (1)

Имя	Тип	Назначение
epdg_network_connection_status	Gauge	Статус TCP/UDP-соединения к peer (0=down, 1=up) — применяется к PGW (S2b), AAA (SWm), HSS (SWx)

#### 4.3 IKEv2 SWu (3)

Имя	Тип	Назначение
epdg_ikev2_messages_total	Counter	Счётчик IKEv2 сообщений (IKE_SA_INIT / IKE_AUTH / CREATE_CHILD_SA / INFORMATIONAL)
epdg_ikev2_request_duration_seconds	Histogram	Распределение времени ответа на IKEv2 request
epdg_ikev2_errors_total	Counter	IKEv2 ошибки (NO_PROPOSAL_CHOSEN, AUTHENTICATION_FAILED, INVALID_SYNTAX и т.д.)

#### 4.4 GTPv2-C S2b (4)

Имя	Тип	Назначение
epdg_gtp_messages_total	Counter	Сообщения GTPv2-C (Create/Modify/Delete Session, Echo)
epdg_gtp_request_duration_seconds	Histogram	Latency request→response
epdg_gtp_errors_total	Counter	GTP-C ошибки по Cause Code
epdg_gtp_retransmissions_total	Counter	Ретрансмиссии GTP-C запросов

#### 4.5 GTP-U data plane (3)

Имя	Тип	Назначение
epdg_gtpu_packets_total	Counter	Пакеты через GTP-U tunnel (uplink/downlink)
epdg_gtpu_bytes_total	Counter	Байты через GTP-U tunnel
epdg_gtpu_errors_total	Counter	Ошибки туннелирования (TEID mismatch, decap fail)

#### 4.6 Diameter SWm/SWx/S6b (5)

Имя	Тип	Назначение
epdg_diameter_messages_total	Counter	DER/DEA (SWm), MAR/MAA (SWx), AAR/AAA (S6b), STR/STA
epdg_diameter_request_duration_seconds	Histogram	Latency Diameter request→answer
epdg_diameter_errors_total	Counter	Ошибки по Experimental-Result-Code
epdg_diameter_watchdog_status	Gauge	Статус DWR/DWA watchdog per peer (0=timeout, 1=ok)
epdg_diameter_connection_status	Gauge	Статус Diameter connection per peer (0=disconnected, 1=connected)

## 4.7 Service KPI (4)

Имя	Тип	Назначение
epdg_service_attach_total	Counter	Попытки attach (success/failure) по APN
epdg_service_attach_duration_seconds	Histogram	Длительность attach (IKE_SA_INIT → session ready)
epdg_service_availability	Gauge	Availability flag (0=down, 1=up)
epdg_service_uptime_seconds	Gauge	Service uptime

## 4.8 Session State (4)

Имя	Тип	Назначение
epdg_session_ike_sa_total	Gauge	Активные IKE SA
epdg_session_child_sa_total	Gauge	Активные Child SA (IPSec tunnels)
epdg_session_gtp_sessions_total	Gauge	Активные GTP-C sessions на S2b
epdg_session_subscribers_total	Gauge	Уникальные абоненты (подключённые UE)

## 4.9 Application (3)

Имя	Тип	Назначение
epdg_app_threads_total	Gauge	Общее число рабочих потоков
epdg_app_memory_bytes	Gauge	Память процесса по типам
epdg_app_log_messages_total	Counter	Лог-сообщения по уровням (debug/info/warn/error/fatal)

## 4.10 System (4)

Имя	Тип	Назначение
epdg_system_cpu_usage_percent	Gauge	CPU usage процесса
epdg_system_memory_bytes	Gauge	Системная память
epdg_system_disk_bytes	Gauge	Дисковое пространство
epdg_system_open_fds	Gauge	Открытые file descriptors

## Типы метрик (напоминание)

Тип	Назначение
<b>Counter</b>	Монотонно растущий счётчик (сообщения, ошибки, reload-ы)
<b>Gauge</b>	Текущее значение (активные сессии, память, статус)
<b>Histogram</b>	Распределение значений с автоматическими percentile bucket'ами (latency, lifetime)

## 5. Интерфейсы интеграции

flowchart LR  
CORE["VAS Experts  
ePDG Monitoring"] CORE --> P["Prometheus  
CNCF / OpenMetrics"]  
CORE --> S["SNMP v2c  
EPDG-MIB"]  
CORE --> G["Grafana  
JSON Provisioning"]  
CORE --> W["Webhooks  
ChatOps"]  
CORE --> AM["Alertmanager"]

Routing"] P --> P1["Cloud-native NMS  
Thanos / Cortex / Mimir"] S --> S1["Legacy NMS  
HP OpenView, NetAct  
IBM Tivoli"] G --> G1["NOC Wall Displays  
Drill-down Analytics"] W --> W1["Telegram / Slack  
PagerDuty / OpsGenie"] AM --> AM1["Smart routing  
Severity-based"]

## 5.1 Prometheus (CNCF Standard)

Нативный `/metrics` endpoint на порту **9817** встроен в `fast-epdg`. Формат — стандартный Prometheus text exposition v0.0.4 (OpenMetrics compatible). Поддерживается federation для агрегации с центральным Prometheus оператора; `remote_write` для долгосрочного хранения в Thanos, Cortex, Grafana Mimir.

## 5.2 SNMP v2c — EPDG-MIB

Enterprise OID VAS Experts (`vas.expert`): **.1.3.6.1.4.1.43823** (регистрация IANA PEN). ePDG MIB живёт поддеревом **.1.3.6.1.4.1.43823.1**. Другие продукты VAS Experts используют соседние поддеревья (например EPC Monitoring на `.43823.100`).

```
flowchart TB
  IANA["IANA PEN enterprises  
.1.3.6.1.4.1"] --> VAS["VAS VAS Experts  
.1.3.6.1.4.1.43823"]
  VAS --> EPDG["EPDG EPDG-MIB  
.43823.1"]
  EPDG --> EPC["EPC EPC Monitoring  
.43823.100"]
  EPDG --> NOTIF["NOTIF epdgNotifications  
.43823.1.1"]
  EPDG --> CONF["CONF epdgConformance  
.43823.1.3"]
  EPDG --> SERVICE["SERVICE service .1.1.1"]
  EPDG --> IKE["IKE ikev2 .1.1.2"]
  EPDG --> GTP["GTP gtp .1.1.3"]
  EPDG --> DIAM["DIAM diameter .1.1.4"]
  EPDG --> SESS["SESS sessions .1.1.5"]
  EPDG --> SYS["SYS system .1.1.6"]
  EPDG --> NET["NET network .1.1.7"]
  NOTIF --> TRAPAGR["TRAPAGR 7 raise / 7 clear pairs"]
```

Итого **47 OID** покрывают SMI-аналог Prometheus-метрик + **14 trap notifications** (с парами `raise/clear` согласно RFC 3877 ALARM-MIB). Совместимость с HP OpenView, IBM Tivoli NetCool, Nokia NetAct, Huawei U2000.

Примеры SNMP-запросов:

```
# Все дерево ePDG
snmpwalk -v2c -c public <host> .1.3.6.1.4.1.43823.1
```

```
# Service availability (Gauge 0..1)
snmpget -v2c -c public <host> .1.3.6.1.4.1.43823.1.1.1.1.0
```

## 5.3 Grafana

4 JSON-провизионированных дашборда (35+ панелей суммарно):

- **ePDG Overview** — availability, attach KPI, sessions, interfaces status
- **IKEv2 Details** — messages, performance, errors, IKE SA lifecycle
- **GTP Details** — GTPv2-C + GTP-U данные по PGW peers
- **Diameter Details** — messages по applications, latency, watchdog

Автоматическая установка через Grafana provisioning API. Адаптивный дизайн для NOC wall displays с авто-refresh 15 секунд.

## 5.4 Alertmanager Webhooks

Webhook-интерфейс для интеграции с любой системой оповещений: Telegram Bot, Slack, PagerDuty Events API v2, OpsGenie, Microsoft Teams. Отдельный **SNMP Trap Sender** service конвертирует Alertmanager webhooks в SNMP v2c traps с Enterprise OID.

# 6. Система алармов

## Категории алармов

Severity	Алармы	Описание	Реакция
<b>Critical</b>	ePDG_Service_Down, ePDG_High_Attach_Failure_Rate, ePDG_PGW_Unreachable, ePDG_AAA_Unreachable, ePDG_Diameter_Watchdog_Timeout	Компонент недоступен, массовый отказ attach, peer unreachable	Немедленная эскалация: Email + SNMP Trap + Webhook. Repeat 1 час
<b>Warning</b>	ePDG_High_IKEv2_Latency, ePDG_High_GTP_Latency, ePDG_High_IKEv2_Error_Rate, ePDG_High_GTP_Error_Rate, ePDG_High_Memory_Usage, ePDG_High_CPU_Usage, ePDG_Low_Disk_Space, ePDG_High_Error_Log_Rate	Деградация производительности, аномалии ресурсов	Email. Repeat 4 часа. Подавляется при наличии Critical на том же компоненте

## Полный перечень алармов (20+ правил)

flowchart LR  
AL["ePDG Alert Rules  
20+"] AL --> CR["Critical  
5 rules"]  
AL --> WR["Warning"]

8 rules"] AL --> INFO["Recording  
 34 rules"] CR --> C1["Service\_Down  
 availability == 0"] CR --> C2["Attach\_Failure\_Rate  
 > 10%"] CR --> C3["PGW\_Unreachable  
 connection\_status{s2b} == 0"] CR --> C4["AAA\_Unreachable  
 connection\_status{swm} == 0"] CR --> C5["Diameter\_Watchdog\_Timeout  
 watchdog\_status == 0"] WR --> W1["High\_IKEv2\_Latency  
 p95 > 1.0 s"] WR --> W2["High\_GTP\_Latency  
 p95 > 0.5 s"] WR --> W3["High\_IKEv2\_Error\_Rate  
 > 5%"] WR --> W4["High\_GTP\_Error\_Rate  
 > 5%"] WR --> W5["High\_Memory\_Usage  
 > 80%"] WR --> W6["High\_CPU\_Usage  
 > 80%"] WR --> W7["Low\_Disk\_Space  
 < 10%"] WR --> W8["High\_Error\_Log\_Rate  
 > 10/s"] INFO --> I1["attach\_success\_rate  
 preaggregated"] INFO --> I2["p95\_p99\_latency  
 preaggregated"] INFO --> I3["throughput  
 preaggregated"]

## Процесс обработки алармов

sequenceDiagram participant M as Метрика (Prometheus) participant R as Alert Rule (PromQL)  
 participant AM as Alertmanager participant E as Email (SMTP) participant SG as SNMP Trap Gateway  
 participant NMS as Внешняя NMS participant W as Webhook (ChatOps) M->>R: Значение  
 превышает порог R->>R: Ожидание (for: 1-10 мин) R->>AM: Alert FIRING AM->>AM: Group by  
 [alertname, component] AM->>AM: Inhibition check (critical подавляет warning) alt severity =  
 critical AM->>E: Email [CRITICAL] AM->>SG: Webhook → SNMP Trap SG->>NMS: SNMP v2c Trap (OID  
 .1.3.6.1.4.1.43823.1.2.X) AM->>W: Webhook (Telegram / PagerDuty) else severity = warning  
 AM->>E: Email [WARNING] end Note over M,R: Метрика возвращается в норму R->>AM: Alert  
 RESOLVED R->>SG: clear-trap (paired notification) AM->>E: Email [RESOLVED]

## Особенности

- **Inhibition:** Critical-алармы автоматически подавляют Warning для того же компонента
- **Grouping:** Алармы группируются по alertname + component с 30-секундным окном
- **Dead time / Hysteresis:** for parameter от 1 до 10 минут предотвращает ложные срабатывания
- **Trap pairing:** raise/clear parallel notifications для соответствия RFC 3877 ALARM-MIB

## 7. Визуализация и операционные дашборды

### Состав дашбордов

Дашборд	Панели	Назначение
ePDG Overview	10	Service availability, attach success rate, active sessions, SWu/SWm/S2b status, throughput per-interface
IKEv2 Details	10	Messages/sec по типам, request duration histogram, 95th percentile latency, errors by type, IKE SA lifecycle

Дашборд	Панели	Назначение
<b>GTP Details</b>	8	GTPv2-C messages по PGW, retransmissions, errors by Cause Code, GTP-U uplink/downlink, bearers
<b>Diameter Details</b>	7	Messages per application (SWm/SWx/S6b), request duration, watchdog status, Result-Code distribution, connection status timeline

## Дизайн для NOC

flowchart TB
 NOC["NOC Dashboard Layer"] --> OVER["ePDG Overview  
KPI Summary"]
 NOC --> IKE["IKEv2 Details  
Drill-down"]
 NOC --> GTP["GTP Details  
Drill-down"]
 NOC --> DIA["Diameter Details  
Drill-down"]
 OVER -->|Click attach KPI| IKE
 OVER -->|Click session count| GTP
 OVER -->|Click peer status| DIA

- **Автообновление:** 15-секундный refresh rate
- **Адаптивная цветовая схема:** зелёный → жёлтый → красный по пороговым значениям
- **Drill-down:** от Overview к детализации per-component
- **Time-range selector:** от 5 минут до 30 дней истории
- **JSON provisioning:** дашборды разворачиваются автоматически

## 8. Интеграция в единый стек EPC Monitoring

ePDG мониторинг полностью интегрирован в общий мониторинг пакетного ядра:

```

flowchart TB
  subgraph Common ["Единый Monitoring Stack"]
    PROM["Prometheus"]
    GRAF["Grafana"]
    AM["Alertmanager"]
  end
  subgraph Sources ["Источники метрик EPC"]
    DPI["FastDPI :9110"]
    SMF["SMF /metrics :9090"]
    PCEF["fast-pcef /metrics :9090"]
    PCRF["FastPCRF"]
    EPDG["fast-epdg :9817"]
  end
  DPI --> PROM
  SMF --> PROM
  PCEF --> PROM
  PCRF --> PROM
  EPDG --> PROM
  PROM --> GRAF
  PROM --> AM

```

Оператор NOC видит **все компоненты EPC** (DPI, SMF, PCEF, FastPCRF, ePDG) в одном интерфейсе Grafana, с единой системой алармов и маршрутизации уведомлений через один Alertmanager.

## 9. Покрытие метрик по уровням OSI

```

graph LR
  L1["L1 Physical  
NIC counters via system"]
  L2["L2 Data Link  
MAC, VLAN"]
  L3["L3 Network  
IP, IPsec ESP, GTP-U"]
  L4["L4 Transport  
TCP/UDP/SCTP"]
  L5["L5 Session  
GTPv2-C, IKEv2"]
  L6["L6 Presentation"]

```

IKEv2/IPSec encryption, EAP-AKA"] L7["L7 Application Diameter, service bearer ops"] Operations["Operations KPI, SLA, Capacity"] CX["CX Level Subscriber Experience"] L1 --> L2 --> L3 --> L4 --> L5 --> L6 --> L7 --> Operations --> CX style L1 fill:#e74c3c,color:#fff style L2 fill:#e67e22,color:#fff style L3 fill:#f39c12,color:#fff style L4 fill:#2ecc71,color:#fff style L5 fill:#1abc9c,color:#fff style L6 fill:#3498db,color:#fff style L7 fill:#9b59b6,color:#fff style Operations fill:#34495e,color:#fff style CX fill:#2c3e50,color:#fff

## Детализация метрик по уровням

Модель OSI:

Уровень	Метрики	Примеры
<b>L1/L2 Physical / Data Link</b>	—	Покрывается отдельным node_exporter / аналогом на уровне ОС (не входит в перечень метрик ePDG)
<b>L3 Network / IPSec tunnels</b>	3	epdg_gtpu_packets_total, epdg_gtpu_bytes_total, epdg_gtpu_errors_total — GTP-U data plane
<b>L4 Transport</b>	1	epdg_network_connection_status — TCP к peers (PGW/AAA/HSS)
<b>L5 Session</b>	3	epdg_session_ike_sa_total, epdg_session_child_sa_total, epdg_session_gtp_sessions_total
<b>L6 Presentation/Security</b>	3	epdg_ikev2_messages_total, epdg_ikev2_request_duration_seconds, epdg_ikev2_errors_total — IKEv2/IPSec шифрование и EAP-AKA' аутентификация
<b>L7 Application</b>	9	epdg_diameter_* (SWm/SWx/S6b, 5 метрик), epdg_gtp_* (GTPv2-C, 4 метрики)

Операторский уровень:

Уровень	Метрики	Примеры
<b>Operations</b>	11	epdg_service_availability, epdg_service_uptime_seconds, epdg_app_* (3), epdg_system_* (4), epdg_config_* (2)
<b>Customer Experience</b>	3	epdg_service_attach_duration_seconds p95, epdg_service_attach_total (success rate), epdg_ikev2_request_duration_seconds p99

## Уровень 9: Качество восприятия сервиса VoWiFi

QoE-индикатор	Метрики-источники	Интерпретация
<b>Время подключения VoWiFi</b>	epdg_service_attach_duration_seconds p95	> 3 сек — абонент замечает задержку при переключении на WiFi
<b>Непрерывность сервиса</b>	epdg_session_ike_sa_total delta	Массовый сброс > 50 IKE SA = проблема доступности
<b>Успешность аутентификации</b>	ePDG_High_Attach_Failure_Rate alert rate	> 5% = проблема с HSS/AAA peers
<b>Задержка назначения bearer</b>	epdg_gtp_request_duration_seconds{msg=create-session} p99	> 500 мс — задержка готовности voice path

QoE-индикатор	Метрики-источники	Интерпретация
Ошибки GTP-U tunnel	epdg_gtpu_errors_total rate / epdg_gtpu_packets_total	> 0.1% = деградация качества голоса
IKEv2-надёжность	epdg_ikev2_errors_total по типам	NO_PROPOSAL_CHOSEN / AUTHENTICATION_FAILED — проблемы с certs / UE

## 10. Стандарты и совместимость

Стандарт	Область	Применение
3GPP TS 29.273	SWx/S6b/SWm	Методология учёта Diameter-сообщений и результирующих кодов
3GPP TS 24.302	SWu (IKEv2)	Определение IKEv2 message types и error codes
3GPP TS 33.402	3GPP security for non-3GPP access	EAP-AKA' / IKEv2 security parameters
3GPP TS 23.402	Non-3GPP access architecture	Структура interfaces (SWu/SWm/SWx/S6b/S2b)
3GPP TS 32.421	Performance Measurement	Методология KPI сбора
3GPP TS 32.409	Performance measurement charging	Структура счётчиков
IETF RFC 7296	IKEv2	Message types, error notifications, SA states
IETF RFC 6733	Diameter	Command codes, Result-Codes
IETF RFC 4187	EAP-AKA	Аутентификация через SIM
IETF RFC 3877	ALARM MIB	Структура Enterprise MIB для алармов
IETF RFC 3418	SNMPv2 MIB	Совместимость SNMP v2c
Prometheus Exposition Format	Metrics (v0.0.4)	Формат экспорта метрик
OpenMetrics	CNCF Standard	Перспективная совместимость

## 11. Модель развёртывания

```

flowchart TB
    subgraph Host1 ["Сервер ePDG"]
        EPDG["fast-epdg (VoWiFi gateway)"]
        PLUGIN["/metrics endpoint :9817"]
        EPDG --> PLUGIN
    end
    subgraph Host2 ["Сервер мониторинга"]
        PROM["Prometheus"]
        GRAF["Grafana"]
        AM["Alertmanager"]
        SNMPTRAP["SNMP Trap Sender (webhook gateway)"]
        PROM --> GRAF
        PROM --> AM
        AM --> SNMPTRAP
    end
    subgraph Host3 ["Внешние системы"]
        NMS["Операторская NMS (HP OpenView / NetAct / Tivoli)"]
        CHAT["ChatOps (Telegram / PagerDuty)"]
    end
    PLUGIN --> |HTTP :9817/metrics| PROM
    PROM --> |UDP 162| SNMPTRAP
    AM --> |Webhook| CHAT

```

### Характеристики развёртывания

Параметр	Значение
Metrics footprint	In-process (~2 MB memory overhead)
Внешние зависимости	Самодостаточный пакет fast-epdg (rpm)

Параметр	Значение
Управление	fast-epdg.service systemd
Конфигурация	Секция monitoring в fast-epdg.conf
Обновление	Hot reload конфигурации без даунтайма
ОС	Linux (RHEL/CentOS 8+, Ubuntu 22.04+)
Порт	9817 TCP (listen 0.0.0.0, настраивается)
Время развёртывания	< 5 минут (enable plugin в config + restart)

## Варианты размещения

- **On-premise** — plugin работает в адресном пространстве fast-epdg, нулевой footprint
- **Co-located Prometheus** — Prometheus рядом с ePDG, scrape через localhost
- **Централизованный** — единый Prometheus собирает со всех ePDG узлов (federation)

## 12. Конфигурация экспортёра метрик

Секция monitoring в fast-epdg.conf:

```
monitoring {
    enabled = yes
    listen_port = 9817
    listen_address = 0.0.0.0
    update_interval = 10
    metrics {
        ikev2 = yes
        gtp = yes
        diameter = yes
        service = yes
        session = yes
        app = yes
        system = yes
    }
}
```

Каждая группа метрик может быть независимо включена/выключена без перекомпиляции.

## 13. План развития системы мониторинга

Следующие направления развития запланированы в ближайших релизах для расширения наблюдаемости и упрощения интеграции с операторскими NOC-процессами. Приоритизация определяется запросами клиентов и incident-ретроспективами.

### 13.1 Расширение перечня метрик

flowchart TB
 ROADMAP["Roadmap: Metrics Expansion"] --> NET["Network L3/L4 расширение"]
 ROADMAP --> SESS["Session State расширение"]
 ROADMAP --> KPI["Service KPI расширение"]
 ROADMAP --> APP["Application расширение"]
 ROADMAP --> SYS["System расширение"]
 NET --> N1["packets/bytes/errors per-interface counters"]
 SESS --> S1["ike\_sa\_lifetime histogram"]
 SESS --> S2["gtp\_bearers по QCI"]
 SESS --> S3["diameter\_sessions per-app"]
 SESS --> S4["subscribers\_by\_apn"]
 KPI --> K1["detach reasons"]
 KPI --> K2["bearer operations"]
 KPI --> K3["throughput per-direction"]
 KPI --> K4["packets\_lost counters"]
 APP --> A1["memory allocations breakdown"]
 APP --> A2["IO operations / wait time"]
 APP --> A3["event loop lag histogram"]
 APP --> A4["file descriptor limits"]
 SYS --> Y1["CPU load average"]
 SYS --> Y2["disk IO operations/time"]
 SYS --> Y3["system network counters"]

Блок	Планируемые метрики	Задача
<b>Network</b>	epdg_network_packets_total, epdg_network_bytes_total, epdg_network_errors_total (по interface/protocol/direction)	Детализация L3/L4 трафика до уровня peer-interface
<b>Session Extended</b>	epdg_session_ike_sa_lifetime_seconds (Histogram), epdg_session_gtp_bearers_total (по apn/qci), epdg_session_diameter_sessions_total (per-application), epdg_session_subscribers_by_apn	Subscriber-level analytics для capacity planning
<b>Service KPI Extended</b>	epdg_service_detach_total (по reason/initiator), epdg_service_bearer_operations_total, epdg_service_throughput_bytes_per_second, epdg_service_packets_lost_total	Полноценный set KPI уровня QoE
<b>Application Extended</b>	epdg_app_threads_active, epdg_app_memory_allocations_total, epdg_app_io_operations_total, epdg_app_io_bytes_total, epdg_app_io_wait_seconds_total, epdg_app_event_loop_lag_seconds, epdg_app_file_descriptors_limit	Глубокая app-level observability для performance tuning
<b>System Extended</b>	epdg_system_cpu_load_average (1m/5m/15m), epdg_system_disk_io_operations_total, epdg_system_disk_io_time_seconds_total, epdg_system_network_bytes_total, epdg_system_network_errors_total	Комплексная картина ресурсов без внешних node_exporter'ов

## 13.2 Планируемые функциональные расширения

Направление	Описание
<b>REST API для просмотра сессий</b>	OpenAPI 3.0-совместимый API с поиском по IMSI/MSISDN/IP, детализацией IKE SA / Child SA / GTP-bearer; Swagger UI для интерактивного использования
<b>Web UI Session Browser (SPA)</b>	Встроенный web-интерфейс для техподдержки: real-time поиск сессий, фильтрация по APN/state, визуализация IPSec SA и bearer lifetime
<b>Zabbix Template pack</b>	Готовые YAML-шаблоны Zabbix 7.0 с HTTP Agent + Prometheus Pattern preprocessing, auto-discovery items, dependent items, trigger expressions
<b>Alert Config UI</b>	Web-интерфейс создания/редактирования Prometheus alert rules без ручного YAML-редактирования, с validation и preview
<b>Расширение EPDG-MIB</b>	Доведение SNMP OID покрытия до 60+ (включая плановые метрики из §13.1), generation MIB-subtree на лету при изменении экспортера

Направление	Описание
<b>Distributed tracing hooks</b>	OpenTelemetry export для корреляции IKEv2 → Diameter → GTP-C latency в одном trace (отладка end-to-end attach задержек)
<b>QoE MOS proxy</b>	Косвенная оценка MOS / R-factor через packet loss + jitter estimation на GTP-U data plane (без polling probe-решений)

### 13.3 Интеграционный roadmap

timeline title Roadmap этапы расширения мониторинга  
 Расширение метрик : Network + Session + KPI + App + System REST API + SPA : Session Browser UI + OpenAPI 3.0  
 Ops-tooling : Zabbix templates + Alert Config UI  
 Advanced observability : OpenTelemetry + Distributed tracing  
 QoE probe : MOS/R-factor proxy на GTP-U

Конкретные сроки реализации определяются дорожной картой релизов VAS Experts и согласуются с клиентскими проектами.

## 14. Заключение

**VAS Experts ePDG Monitoring** — встроенный мониторинг шлюза VoWiFi с полным покрытием всех интерфейсов 3GPP EPC non-3GPP access:

1. **Сквозное покрытие SWu → S2b** — IKEv2 (SWu), Diameter SWm/SWx/S6b, GTPv2-C + GTP-U (S2b), на всех уровнях L2-L9. **33 метрики 47 SNMP OID 14 trap notifications**
2. **Скорость реагирования** — 10-секундный цикл сбора, 13+ алармов с hysteresis, автоэскалация через Email/SNMP Trap/Webhook.
3. **Открытость интеграции** — Prometheus, SNMP v2c Enterprise MIB, Grafana, Alertmanager webhooks. Совместимость с Nokia NetAct, HP OpenView, IBM Tivoli, Zabbix, PRTG.
4. **Полнота визуализации** — 4 дашборда Grafana, 35+ панелей, **34 recording rules** для pre-aggregated KPI (attach success rate, p95/p99 latency, throughput).
5. **Минимальная стоимость владения** — самодостаточный пакет fast-epdg, без внешних агентов / runtime-зависимостей, развёртывание за 5 минут, нативный /metrics endpoint.
6. **Единый стек с EPC Monitoring** — полная интеграция в общий Prometheus/Grafana/Alertmanager оператора вместе с DPI/SMF/PCEF/FastPCRF.

### Следующие шаги

Для получения демонстрации, технической спецификации или коммерческого предложения:

- Сайт: [vasexperts.ru](http://vasexperts.ru)
- Техническая поддержка: [support@vasexperts.ru](mailto:support@vasexperts.ru)
- Отдел продаж: [sales@vasexperts.ru](mailto:sales@vasexperts.ru)

---

VAS Experts — российский разработчик решений глубокой инспекции трафика (DPI), управления политиками и тарификации, шлюзов VoWiFi для операторов мобильной и фиксированной связи. Продукты VAS Experts используются операторами в России и странах

*СНГ для СОРМ, BRAS, NAT/CG-NAT, QoS, пакетного ядра LTE и VoWiFi.*