GS LABS
GS GROUP

Публичный - L0

Сервис авторизации и проверки доступа

Руководство администратора

Индекс	Shield.Int-AG
Конфиденциальность	Публичный - L0
Ревизия	1.0
Статус	Согласован

Содержание

1. Введение

1.1. Назначение

Настоящий документ содержит описание способов мониторинга, ведения логов, диагностики состояний, а также описание конфигурационных параметров **Сервиса авторизации и проверки доступа** (далее в документе для краткости используется условное наименование данного сервиса - **Shield**).

1.2. Справочная документация

№	Наименование документа	Индекс документа (если есть)
1	Messages.Auth.Internal	2004-Shield.Messages.Auth.Internal-1.1.0-FS

2. Мониторинг

Shield не имеет пользовательского интерфейса для мониторинга, поэтому наблюдение происходит через внешние инструменты.

> ⚠ Инструменты Prometheus и Grafana, упоминаемые в данной главе, являются сторонними по отношению к Shield продуктами. Описание их структуры, принципов действия, процедур установки и т. п. выходит за рамки настоящего документа.

2.1. Prometheus

Shield не имеет особых метрик для Prometheus, поэтому сбор этих метрик настраивать не требуется. Однако Kubernetes компоненты предоставляют большой набор метрик, который можно использовать для мониторинга, например, метрики nginx ingress.

2.2. Grafana

Grafana выполняет несколько задач в Shield:

1. Отображение Prometheus метрик.
2. Отображение логов компонентов.

2.2.1. Логи компонентов

Логи компонентов отображаются в Grafana путём подключения как DataSource системы Grafana Loki или Elasticsearch. Настройка сбора логов остаётся на усмотрение администратора, но т.к. на тестовых зонах используется Grafana Loki ввиду его простоты, то здесь представлен пример конфигурации дэшборда для Grafana Loki:

logs.json

```
{
  "annotations": {
    "list": [
      {
        "builtIn": 1,
        "datasource": "-- Grafana --",
        "enable": true,
        "hide": true,
        "iconColor": "rgba(0, 211, 255, 1)",
        "name": "Annotations & Alerts",
        "type": "dashboard"
      }
    ]
  },
  "editable": true,
```

```
  "gnetId": null,
  "graphTooltip": 0,
  "id": 2,
  "iteration": 1594219172374,
  "links": [],
  "panels": [
    {
      "datasource": "Loki",
      "fieldConfig": {
        "defaults": {
          "custom": {}
        },
        "overrides": []
      },
      "gridPos": {
        "h": 14,
        "w": 12,
        "x": 0,
        "y": 0
      },
      "id": 2,
      "options": {
        "showLabels": false,
        "showTime": true,
        "sortOrder": "Descending",
        "wrapLogMessage": true
      },
      "targets": [
        {
          "expr": "{namespace=\"$namespace\", app=\"api-gateway-api-gateway\", container=\"api-gateway\"}",
          "refId": "A"
        }
      ],
      "timeFrom": null,
      "timeShift": null,
      "title": "API Gateway",
      "type": "logs"
    },
    {
      "datasource": "Loki",
      "fieldConfig": {
        "defaults": {
          "custom": {}
        },
        "overrides": []
      },
      "gridPos": {
        "h": 10,
        "w": 12,
        "x": 12,
        "y": 0
      },
      "id": 4,
      "options": {
        "showLabels": false,
        "showTime": true,
        "sortOrder": "Descending",
        "wrapLogMessage": true
      },
      "targets": [
        {
          "expr": "{namespace=\"$namespace\", app=\"pechkin-pechkin\", container=\"pechkin\"} |= \"sending
message\"",
          "refId": "A"
        }
      ],
      "timeFrom": null,
      "timeShift": null,
```

```
    "title": "Pechkin SMS messages",
    "type": "logs"
  },
  {
    "datasource": "Loki",
    "fieldConfig": {
      "defaults": {
        "custom": {}
      },
      "overrides": []
    },
    "gridPos": {
      "h": 13,
      "w": 12,
      "x": 12,
      "y": 10
    },
    "id": 6,
    "options": {
      "showLabels": false,
      "showTime": true,
      "sortOrder": "Descending",
      "wrapLogMessage": false
    },
    "targets": [
      {
        "expr": "{namespace=\"$namespace\", app=\"id-provider-id-provider\", container=\"id-provider\"}",
        "refId": "A"
      }
    ],
    "timeFrom": null,
    "timeShift": null,
    "title": "ID Provider logs",
    "type": "logs"
  },
  {
    "datasource": "Loki",
    "fieldConfig": {
      "defaults": {
        "custom": {}
      },
      "overrides": []
    },
    "gridPos": {
      "h": 13,
      "w": 12,
      "x": 0,
      "y": 14
    },
    "id": 12,
    "options": {
      "showLabels": false,
      "showTime": true,
      "sortOrder": "Descending",
      "wrapLogMessage": false
    },
    "targets": [
      {
        "expr": "{namespace=\"$namespace\", app=\"domains-domains\", container=\"domains\"}",
        "refId": "A"
      }
    ],
    "timeFrom": null,
    "timeShift": null,
    "title": "Domains logs",
    "type": "logs"
  },
  {
```

```
    "datasource": "Loki",
    "fieldConfig": {
      "defaults": {
        "custom": {}
      },
      "overrides": []
    },
    "gridPos": {
      "h": 13,
      "w": 12,
      "x": 12,
      "y": 23
    },
    "id": 15,
    "options": {
      "showLabels": false,
      "showTime": true,
      "sortOrder": "Descending",
      "wrapLogMessage": false
    },
    "targets": [
      {
        "expr": "{namespace=\"$namespace\", app=\"policies-policies\", container=\"policies\"}",
        "refId": "A"
      }
    ],
    "timeFrom": null,
    "timeShift": null,
    "title": "Policies logs",
    "type": "logs"
  },
  {
    "datasource": "Loki",
    "fieldConfig": {
      "defaults": {
        "custom": {}
      },
      "overrides": []
    },
    "gridPos": {
      "h": 13,
      "w": 12,
      "x": 0,
      "y": 27
    },
    "id": 14,
    "options": {
      "showLabels": false,
      "showTime": true,
      "sortOrder": "Descending",
      "wrapLogMessage": false
    },
    "targets": [
      {
        "expr": "{namespace=\"$namespace\", app=\"accounts-accounts\", container=\"accounts\"}",
        "refId": "A"
      }
    ],
    "timeFrom": null,
    "timeShift": null,
    "title": "Accounts logs",
    "type": "logs"
  },
  {
    "datasource": "Loki",
    "fieldConfig": {
      "defaults": {
        "custom": {}
```

```
      },
      "overrides": []
    },
    "gridPos": {
      "h": 10,
      "w": 12,
      "x": 12,
      "y": 36
    },
    "id": 10,
    "options": {
      "showLabels": false,
      "showTime": true,
      "sortOrder": "Descending",
      "wrapLogMessage": false
    },
    "targets": [
      {
        "expr": "{namespace=\"$namespace\", app=\"rpc-rpc\"}",
        "refId": "A"
      }
    ],
    "timeFrom": null,
    "timeShift": null,
    "title": "Remote Control Logs",
    "type": "logs"
  },
  {
    "datasource": "Loki",
    "fieldConfig": {
      "defaults": {
        "custom": {}
      },
      "overrides": []
    },
    "gridPos": {
      "h": 19,
      "w": 12,
      "x": 0,
      "y": 40
    },
    "id": 13,
    "options": {
      "showLabels": false,
      "showTime": true,
      "sortOrder": "Descending",
      "wrapLogMessage": false
    },
    "targets": [
      {
        "expr": "{namespace=\"$namespace\", app=\"devices-devices\", container=\"devices\"}",
        "refId": "A"
      }
    ],
    "timeFrom": null,
    "timeShift": null,
    "title": "Devices logs",
    "type": "logs"
  },
  {
    "datasource": "Loki",
    "description": "Tokens engine",
    "fieldConfig": {
      "defaults": {
        "custom": {}
      },
      "overrides": []
    },
```

```
    "gridPos": {
      "h": 10,
      "w": 12,
      "x": 12,
      "y": 46
    },
    "id": 11,
    "options": {
      "showLabels": false,
      "showTime": true,
      "sortOrder": "Descending",
      "wrapLogMessage": false
    },
    "targets": [
      {
        "expr": "{namespace=\"$namespace\", app=\"shield-shield\"}",
        "refId": "A"
      }
    ],
    "timeFrom": null,
    "timeShift": null,
    "title": "Shield",
    "type": "logs"
  }
],
"refresh": "30s",
"schemaVersion": 25,
"style": "dark",
"tags": [],
"templating": {
  "list": [
    {
      "allValue": null,
      "current": {
        "selected": false,
        "tags": [],
        "text": "smart-home-cloud",
        "value": "smart-home-cloud"
      },
      "hide": 0,
      "includeAll": false,
      "label": null,
      "multi": false,
      "name": "namespace",
      "options": [
        {
          "selected": false,
          "text": "smart-home-cloud",
          "value": "smart-home-cloud"
        },
        {
          "selected": false,
          "text": "smart-home-cloud-dev",
          "value": "smart-home-cloud-dev"
        },
        {
          "selected": true,
          "text": "smart-home-cloud-int",
          "value": "smart-home-cloud-int"
        }
      ],
      "query": "smart-home-cloud,smart-home-cloud-dev,smart-home-cloud-int",
      "skipUrlSync": false,
      "type": "custom"
    }
  ]
},
"time": {
```

```
    "from": "now-15m",
    "to": "now"
  },
  "timepicker": {
    "refresh_intervals": [
      "10s",
      "30s",
      "1m",
      "5m",
      "15m",
      "30m",
      "1h",
      "2h",
      "1d"
    ]
  },
  "timezone": "",
  "title": "Debug Logs",
  "uid": "e6YuM81Wk",
  "version": 18
}
```

3. Ведение Логов

3.1. Режимы Ведения Логов

Shield ведет логи, информация из которых может быть использована для решения возникающих проблем. Логи могут вестись с разной степенью подробности.

Доступны следующие режимы ведения логов:

- 0 - trace: подробная информация по любым действиям;
- 1 - debug: конфигурационные данные (при запуске системы), другая информация, необходимая для отладки, + сообщения уровня Info;
- 2 - info (значение по умолчанию): базовая информация (сообщения о запуске, работе, выключении системы) + сообщения уровня Warning;
- 3 - warning: системные предупреждения + сообщения уровня Error;
- 4 - error: все ошибки, возникающие в процессе работы, в том числе ошибки уровня Fatal;
- 5 - fatal: критические ошибки, приводящие к сбоям системы.

3.2. Формат записей

Все логи представлены в формате JSON. Каждое сообщение лога представлено в формате отдельной JSON-структуры с исчерпывающим набором полей.

В зависимости от уровня логирования изменяется набор данных внутри JSON либо добавляются новые элементы (записи лога).

Примеры логов:

```
{"details":{"cid":"64cd192d-2fb9-479f-862f-dd47e0b57d77","method":"/gs_labs.lambda.shield.id.IdProvider
/Verify"},"level":"info","msg":"Incoming request","time":"2021-02-25T14:27:22.719544896Z"}
{"details":{"cid":"64cd192d-2fb9-479f-862f-dd47e0b57d77","duration":"0.0002","method":"/gs_labs.lambda.shield.id.
IdProvider/Verify"},"level":"info","msg":"Request time","time":"2021-02-25T14:27:22.719751343Z"}
{"details":{"cid":"64cd192d-2fb9-479f-862f-dd47e0b57d77","duration":"0.0002","err":"wrong requisites","method":"
/gs_labs.lambda.shield.id.IdProvider/Verify"},"level":"error","msg":"error handling request","time":"2021-02-
25T14:27:22.719777128Z"}
```

Примечание. Все логи начинаются с CID (идентификатор запроса), благодаря которому можно отследить один и тот же запрос в разных компонентах.

4. Диагностика

Ввиду того, что продукт Shield разработан для Kubernetes, диагностика системы происходит путём анализа логов и состояния Kubernetes pods. В ситуациях, когда возникают неожиданные ошибки, необходимо обратиться к разработчику для консультации.

5. Обработка ошибок

При возникновении ошибочной ситуации, компонент генерирует сообщение об ошибке и текст, который будет отправлен в логи.

Сообщения об ошибках обрабатываются и передаются клиентскому приложению. В частности ошибки, возвращаемые из сервисов *Accounts, Domains, Devices, Id-provider* через сервис *Id-Provider* передаются клиентскому приложению.

Сообщения и их параметры (обозначение ошибки, её код, описание возможной причины) описаны в документе [1] (не является публичной информацией).

6. Параметры конфигурации

Shield построен на микросервисной архитектуре и функционирует на основе взаимодействия между собой следующих компонентов (сервисов) системы:

- Accounts
- API Gateway
- Devices
- Domains
- Id-Provider
- Operations
- Policies
- Shield
- Error-mapper
- Notify server

В данной главе для каждого компонента системы приведены конфигурационные параметры из проекта для развертывания Shield в kubernetes с описанием на русском языке и указанием значений по умолчанию.

Конфигурацию сервисов по умолчанию можно посмотреть в файле *default.yaml* проекта. Ссылка на проект предоставляется заказчику по запросу.

6.1. Accounts

Сервис Accounts предназначен для управления учетными записями пользователей Умного дома.
Параметры переменных окружения сервиса предоставляются заказчику по требованию.

6.2. API Gateway

API Gateway - сервис роутинга запросов, проксирует API от всех внешних подсистем нужному провайдеру.
Параметры переменных окружения сервиса предоставляются заказчику по требованию.

6.3. Devices

Сервис Devices предназначен для управления устройствами пользователей умного дома.
Параметры переменных окружения сервиса предоставляются заказчику по требованию.

6.4. Domains

Сервис Domains предназначен для объединения нескольких аккаунтов в один домен и привязки устройств умного дома к домену.
Параметры переменных окружения сервиса предоставляются заказчику по требованию.

6.5. API Gateway

API Gateway - точка входа в Shield, проксирует API от всех внешних подсистем нужному Provider.
Параметры переменных окружения сервиса предоставляются заказчику по требованию.

6.6. Id-Provider

Сервис ID Provider осуществляет проверку данных для авторизации и формирует ответ. Данный сервис является обёрткой для работы с сервисами Accounts, Domains, Devices для пользователей умного дома (УД). Имеет отдельные БД: PostgreSQL - выполняет роль хранилища данных сервиса, Redis - хранилище токенов и кэшируемой информации.
Параметры переменных окружения сервиса предоставляются заказчику по требованию.

6.7. Operations

Сервис Operations предназначен для управления аккаунтами пользователей. Разделяются два вида аккаунтов - customers и admins. Аккаунтами customers считаются аккаунты, заведенные в сервисе Accounts. Аккаунтами admins считаются аккаунты, заведенные в сервисе Admin-Accounts.
Параметры переменных окружения сервиса предоставляются заказчику по требованию.

6.8. Policies

Сервис Policies предназначен для управления правами доступа пользователей к ресурсам. Сервис имеет две функции: публикация прав доступа к ресурсу (выдача субъекту авторизации прав на управление ресурсами) и проверка прав доступа к ресурсу. Права доступа хранятся в key-value хранилище etcd.
Параметры переменных окружения сервиса предоставляются заказчику по требованию.

6.9. Shield

Сервис Shield отвечает за прием запросов на авторизацию и выдачу ответов пользователям (предназначен для получения, хранения, валидации и отзыва токенов доступа - токена доступа к аккаунту и токена доступа к устройству). Имеет отдельную БД Redis - хранилище токенов и кэшируемой информации.
Параметры переменных окружения сервиса предоставляются заказчику по требованию.

6.10. Notify Server

Сервис Notify server предназначен для направления в Okydrom запросов по созданию задач о нотификации BSS об операциях с аккаунтами, доменами и контроллерами в Shield.
Параметры переменных окружения сервиса предоставляются заказчику по требованию.

6.11. error_maps_i18n_shield

Kubernetes Jobs реализует начальное наполнение error-mapper.
Параметры переменных окружения сервиса предоставляются заказчику по требованию.

6.12. error_mapper_db_sch

Kubernetes Jobs реализует установку схем error-mapper.

6.13. error_mapper_db_api

Kubernetes Jobs реализует установку api error-mapper.
Параметры переменных окружения сервиса предоставляются заказчику по требованию.

6.14. Error-mapper

Сервис реализует маппинг внутренних ошибок на внешние.
Параметры переменных окружения сервиса предоставляются заказчику по требованию.