ЛОЦИЯ
ИТ-КОМПАНИЯ

Руководство администратора и пользователя программного обеспечения «Continuous Advanced Management Penetration Organisation Testing (CAMPOT)»

Москва – 2024

СОДЕРЖАНИЕ

ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ

Ниже приведены термины, определения и сокращения, используемые в Инструкции администратора и пользователя программного обеспечения «Continuous Advanced Management Penetration Organisation Testing (CAMPOT)».

Сокращения:

ID	Идентификатор
API	Application programming interface — программный интерфейс приложения
CVE	Common Vulnerabilities and Exposures - база данных общеизвестных уязвимостей информационной безопасности.
CWE	Common Weakness Enumeration - система категорий слабых мест и уязвимостей аппаратного и программного обеспечения
TCP/UDP	Transmission Control Protocol/ User Datagram Protocol
URL	Uniform Resource Locator — единообразный указатель местонахождения ресурса
ОС	Операционная система

Термины и определения:

Дедупликация	Специализированный метод сжатия массива данных, использующий в качестве алгоритма сжатия исключение дублирующих копий повторяющихся данных.
Иммутабельность	Неизменяемость (объекта)
ИТ-инфраструктура	Совокупность программных, аппаратных и иных средств, используемых в организации для осуществления информационного обмена
Кластер	Группа компьютеров, серверов или процессоров, объединённых высокоскоростными каналами связи, представляющая с точки зрения пользователя единый аппаратный ресурс.
Митигация (уязвимости)	Снижение влияния потенциальных последствий, вызванных эксплуатацией уязвимости
Нода (кластера)	Сервер (хост), соединенный с другими узлами (серверами) в рамках Кластера
Порт	Сетевой порт - число, которое идентифицирует назначение сетевых потоков данных в пределах одного компьютера
Парсинг (данных)	Извлечение структурированной информации из неструктурированных или полуструктурированных данных.
Пространство имён	Некоторое множество каким-либо образом взаимосвязанных имён или терминов, логический контейнер, в котором все имена уникальны.
Система CAMPOT	«Continuous Advanced Management Penetration Organisation Testing (CAMPOT)». Решение класса CPT (Continuous Penetration Testing) для непрерывного контроля защищенности IT-инфраструктуры компании.
Тег	Ассоциированное ключевое слово, относящееся к какой-либо информации
Конечный узел	Любое устройство, предоставляющее сервисы формата «клиент-сервер» в режиме сервера по каким-либо интерфейсам и уникально определённое на этих интерфейсах.
ElasticSearch	Elasticsearch — тиражируемая программная поисковая система
Kafka-топик	Способ группировки и распределения потоков Big Data сообщений по категориям.

I. ИНСТРУКЦИЯ АДМИНИСТРАТОРА

1. ОБЩЕЕ ОПИСАНИЕ ИНСТРУКЦИИ АДМИНИСТРАТОРА

1.1. Инструкция администратора программного обеспечения «Continuous Advanced Management Penetration Organisation Testing (CAMPOT)» (далее – «система CAMPOT») предназначена для администраторов, осуществляющих установку и настройку системы CAMPOT.

1.2. Инструкция содержит описание взаимодействия Администратора с системой CAMPOT, предназначенной для непрерывного контроля защищенности ИТ-инфраструктуры организации и позволяющей выявлять уязвимости на этапе их появления.

2. ОПИСАНИЕ И ФУНКЦИИ СИСТЕМЫ CAMPOT

2.1. Система CAMPOT представляет из себя оркестратор для запуска различных сканеров уязвимостей.

2.2. Система CAMPOT построена на микросервисной архитектуре и включает в себя 3 сервиса:

- Сервис «CamPot Watcher» - отвечает за постановку задач и мониторинг их выполнения;
- Сервис «CamPot Data Render» - отвечает за обработку и сохранение результата сканирования;
- Сервис «CamPot Trigger» - отвечает за запуск сканирования «по запросу».

2.3. Для хранения и организации потоков данных программный комплекс использует:

- Elastic Search;
- Apache Kafka.

2.4. Для запуска сканеров используются следующие окружения:

- Kubernetes;
- Приложения совместимые по API с Tenable Nessus;
- Приложения совместимые по API с Acunetix.

3. ЗАПУСК И КОНФИГУРИРОВАНИЕ СИСТЕМЫ

3.1. Каждый сервис системы CAMPOT представляет из себя бинарный файл и является самодостаточным для запуска.

3.2. Порядок запуска сервисов не имеет значения. Во время запуска, каждый сервис инициализирует базу данных, при этом выполнение дополнительных миграций не требуется.

3.3. Система «CamPot Data Render» при запуске инициализирует Kafka-топики.

3.4. Сервисы работают в системах под управлением ОС Linux. Также возможен запуск в контейнеризированных средах, в том числе с применением оркестраторов.

3.5. Конфигурирование системы осуществляется посредством переменных окружения. Переменные окружения указываются в файле «.env», располагающемся в одной директории вместе с бинарным файлом сервиса.

4. ОПИСАНИЕ ПЕРЕМЕННЫХ ОКРУЖЕНИЯ

4.1. Переменные окружения для Сервиса «CamPot Watcher»:

- `ELASTIC_URL` - URL-адрес ElasticSearch;
- `ELASTIC_USERNAME` - имя пользователя ElasticSearch;
- `ELASTIC_PASSWORD` - пароль пользователя ElasticSearch;
- `KAFKA_URL` - URL-адрес Kafka;
- `KAFKA_CA_PATH` - путь до CA сертификата Kafka;
- `KAFKA_CERT_PATH` - путь до клиентского сертификата Kafka;
- `KAFKA_KEY_PATH` - путь до клиентского ключа Kafka;
- `NESSUS_USER` - имя пользователя nessus (или альтернативы);
- `NESSUS_PASSWORD`- пароль пользователя nessus (или альтернативы);
- `NESSUS_BASE_URL` - URL адрес nessus (или альтернативы);
- `NESSUS_API_TOKEN` - X-Api-Token для nessus (или альтернативы);
- `ACUNETIX_TOKEN` - Api-Token для acunetix (или альтернативы);
- `ACUNETIX_BASE_URL` - URL-адрес acunetix (или альтернативы).

4.2. Переменные окружения для Сервиса «CamPot Data Render»:

- `ELASTIC_URL` - URL-адрес ElasticSearch;
- `ELASTIC_USERNAME` - имя пользователя ElasticSearch;

- `ELASTIC_PASSWORD` -пароль пользователя ElasticSearch;
- `KAFKA_URL` - URL адрес Kafka;
- `KAFKA_CA_PATH` - путь до CA сертификата Kafka;
- `KAFKA_CERT_PATH` - путь до клиентского сертификата Kafka;
- `KAFKA_KEY_PATH` - путь до клиентского ключа Kafka.

4.3. Переменные окружения для Сервиса «CamPot Trigger»:

- `ELASTIC_URL` - URL адрес ElasticSearch;
- `ELASTIC_USERNAME` - имя пользователя ElasticSearch;
- `ELASTIC_PASSWORD` - пароль пользователя ElasticSearch.

II. ИНСТРУКЦИЯ ПОЛЬЗОВАТЕЛЯ

1. ОБЩЕЕ ОПИСАНИЕ ИНСТРУКЦИИ ПОЛЬЗОВАТЕЛЯ

1.1. Инструкция по эксплуатации системы «Continuous Advanced Management Penetration Organisation Testing (CAMPOT)» (далее – «система CAMPOT») предназначена для пользователей, осуществляющих эксплуатацию и управление системой. Иных функциональных ролей в системе не предусмотрено.

1.2. Система CAMPOT представляет собой серверную часть программного обеспечения (далее – «ПО») для работы со сканированием внешнего периметра ИТ-инфраструктуры. Эксплуатация ПО не предполагает наличие графического пользовательского интерфейса, все функции программного обеспечения могут быть реализованы Пользователем с помощью шагов, описанных в настоящей Инструкции.

1.3. Инструкция содержит описание взаимодействия Пользователя с системой CAMPOT, предназначенной для непрерывного контроля защищенности ИТ-инфраструктуры организации и позволяющей выявлять уязвимости на этапе их появления.

2. ФУНКЦИИ СИСТЕМЫ CAMPOT

2.1. Объекты в ElasticSearch, используемые для взаимодействия с системой делятся на две группы:

- **Собираемые объекты** - сканируемые объекты и результаты сканирования;
- **Объекты управления** - объекты управления сканерами.

2.2. Все взаимодействия с системой осуществляются посредством манипуляции объектами в ElasticSearch.

2.3. Разграничение доступа и интеграция со сторонними сервисами осуществляется средствами ElasticSearch.

2.4. Каждая группа объектов хранится в своем индексе. Для собираемых объектов ежедневно создается новый индекс с периодом ротации длительностью 3 (три) месяца.

3. ОПИСАНИЕ СОБИРАЕМЫХ ОБЪЕКТОВ

3.1. **Собираемые объекты** – объекты, получаемые в результате работы сканеров, являются неизменяемыми. Создание/изменение объекта происходит путем добавления нового, объект с максимальным значением «created_at» является актуальным состоянием объекта. Объект идентифицируется по полю «id».

3.2. **Конечный узел** – сканируемый объект, идентифицируется IP-адресом, шаблон индекса: «compot-host-*». Перечень полей объекта:

- ID (id, uuid) - идентификатор объекта;
- CreatedAt (created_at, data) - время создания объекта;
- Name (name, text) – человекочитаемый идентификатор объекта. Составляется из организации и IP-адреса;
- Organization (organization, text) – организация, которой принадлежит объект;
- Deleted (deleted, bool) - объекты иммутабельны, удаление происходит путем добавления соответствующего флага;
- LastUpdatedBy (last_updated_by, uuid) – ID сканера, внесшего изменения последним;
- CreatedBy (created_by, uuid) - ID сканера, в результате работы которого объект был создан (может быть пустым, актуально для объектов созданных резолверами (resolver);
- Fqdn (fqdn, text) - FQDN хоста;
- Ip (ip, text) – IP-адрес хоста;
- Os (os, text) - операционная система хоста;
- Ports (ports, list) - открытые порты на хосте:

– Number (number, num) - номер порта;

– ProtocolL4 (protocol_l4, text) - протокол L4 (TCP/UDP);
– ProtocolL7 (protocol_l7, text) - протокол L7;
– Service (service, text) - название сервиса, занимающего выбранный порт;
• About (about, list) - информация о хосте (severity - 0, в контексте работы сканеров):
– Name (name, text) - имя записи, используется для идентификации и дедупликации;
– Data (data, text) – строка с информацией;
– Recommendation (recommendation, text) – рекомендации (при наличии);
– RawDataId (raw_data_id, uuid) – ID записи с «сырыми» (необработанными) данными;
• Vulnerability (vulnerability, list) – Список уязвимостей:
– ScannedBy (scanned_by, uuid) – ID сканера, выявившего уязвимость;
– VulnerabilityId (vulnerability_id, uuid) – ID уязвимости;
– Name (name, text) – имя записи, используется для идентификации и дедупликации по умолчанию;
– Status (status, num) –статус уязвимости:
0 - актуальная
1 - передана в обработку
2 - не воспроизводится
3 - митигирована

3.3. Сервис - Сканируемый объект. Идентифицируется URL адресом. Шаблон индекса: «compot-service-*». Перечень полей объекта:

• ID (id, uuid) - идентификатор объекта;
• CreatedAt (created_at, data) - время создания объекта;
• Name (name, text) – человекочитаемый идентификатор объекта, составляется из организации и URL;
• Organization (organization, text) – организация, которой принадлежит объект;
• Deleted (deleted, bool) - объекты иммутабельны, удаление происходит путем добавления соответствующего флага;
• LastUpdatedBy (last_updated_by, uuid) - ID сканера, внесшего изменения последним;
• CreatedBy (created_by, uuid) - ID сканера, в результате работы которого объект был создан (может быть пустым, актуально для объектов созданных резолверами (resolver);
• Url (url, text) - URL сервиса (в наиболее частых случаях подразумевается базовый URL).
• Port (port, text) – порт, на котором работает сервис.
• About (about, list) - информация о сервисе (severity - 0, в контексте работы сканеров):
– Name (name, text) - имя записи, используется для идентификации и дедупликации;
– Data (data, text) - строка с информацией;
– Recommendation (recommendation, text) – рекомендации (при наличии);
– RawDataId (raw_data_id, uuid) - ID записи с «сырыми» (необработанными) данными.
• Vulnerability (vulnerability, list) - список уязвимостей:
– ScannedBy (scanned_by, uuid) – ID сканера, выявившего уязвимость;
– VulnerabilityId (vulnerability_id, uuid) – ID уязвимости;
– Name (name, text) – имя записи, используется для идентификации и дедупликации по умолчанию;
– Status (status, num) - статус уязвимости:
0 – актуальная
1 – передана в обработку
2 – не воспроизводится
3 – митигирована

3.4. Уязвимость – результат сканирования. Шаблон индекса «compot-vulnerability-*». Перечень полей объекта:

• ID (id, uuid) - идентификатор объекта;
• CreatedAt (created_at, data) - время создания объекта;

- Name (name, text) – человекочитаемый идентификатор объекта. Составляется из организации и URL;
- Organization (organization, text) – организация, которой принадлежит объект;
- Deleted (deleted, bool) - объекты иммутабельны, удаление происходит путем добавления соответствующего флага;
- LastUpdatedBy (last_updated_by, uuid) - ID сканера, внесшего изменения последним;
- CreatedBy (created_by, uuid) - ID сканера, в результате работы которого объект был создан (может быть пустым, актуально для объектов созданных резолверами (resolver);
- Synopsis (synopsis, text) - краткое описание уязвимости;
- Description (description, text) -полное описание выявленной уязвимости;
- Solution (solution, text) - рекомендации по устранению/митигации уязвимости;
- Severity (severity, num) - приоритет уязвимости;
- Cvss (cvss, text) - приоритет по CVSS;
- ExtRefs (ext_refs, list) - внешние ссылки (CVE/CWE номера, ссылки на доп. информацию):
  – Name (name, text) - имя записи, используется для идентификации и дедупликации;
  – Url (url, text) - URL внешнего источника.
  – Type (type, text) - Тип внешней ссылки (например: CVE, Info, CWE etc.)
- Ports (ports, list) – необработанные («сырые») результаты запросов сканеров к цели. Данные привязаны к сканируемому порту:
  – Number (number, num) - номер порта;
  – ScannerDataId (scanner_data_id, uuid) - ID записи данными.
- Target (target, text) – информация, передаваемая непосредственно сканеру, для определения цели сканирования (объекта сканирования);
- TargetId (target_id, uuid) - ID объекта, в котором выявлена уязвимость.
- Status (status, num) - статус уязвимости:

0 – актуальная
1 – передана в обработку
2 – не воспроизводится
3 – митигирована

3.5. Исходные данные - Результат сканирования. Шаблон индекса: «compot-raw-*». Перечень полей объекта:

- ID (id, uuid) - идентификатор объекта;
- CreatedAt (created_at, data) - время создания объекта;
- Name (name, text) – человекочитаемый идентификатор объекта. Составляется из организации и URL;
- Organization (organization, text) – организация, которой принадлежит объект;
- Deleted (deleted, bool) - объекты иммутабельны, удаление происходит путем добавления соответствующего флага;
- LastUpdatedBy (last_updated_by, uuid) - ID сканера, внесшего изменения последним;
- CreatedBy (created_by, uuid) - ID сканера, в результате работы которого объект был создан (может быть пустым, актуально для объектов созданных резолверами (resolver);
- Data (data, text) – необработанные («сырые») данные.

4. ОПИСАНИЕ ОБЪЕКТОВ УПРАВЛЕНИЯ

4.1. Объекты управления – объекты, описывающие логику запуска сканеров. Являются изменяемыми, удаляются перманентно.

4.2. Задача – объект, описывающий задачу сканирования и содержащий сведения о текущем состоянии задачи. Индекс: «compot-job». Перечень полей объекта:

- ID (id, uuid) - идентификатор объекта;
- CreatedAt (created_at, data) - время создания объекта;

- Name (name, text) – человекочитаемый идентификатор объекта. Составляется из организации и URL;
- Organization (organization, text) – организация, которой принадлежит объект;
- Frequency (frequency, num) -период между запусками (в минутах);
- Delay (delay, num) -диапазон случайной задержки (от 0), в минутах (наступает после Frequency);
- EndTask (ent_task, date) - время завершения последней задачи;
- Status (status, num) - статус выполнения задачи;

– 0 - создана
– 1 - завершена
– 2 - выполняется
– 3 - ошибка выполнения
– 4 - завершилась с ошибкой
– 5 – осуществляется парсинг
– 6 - хост/сервис не доступен
– 7 - ошибка объекта

- Target (target, text) - цель (Url или IP адрес);
- TargetId (target_id, text) - ID объекта, который является целью;
- TargetType (target_type, text) - Тип цели (Host или Service):

– asset – необработанные («сырые») данные, предоставляемые сканеру. Объект может отсутствовать;
– host - объект типа «Host»;
– service - объект типа «Service»;

- Scanner (scanner, object) - объект описывающий сканер:

– Image (image, text) - имя образа (путь до реестра и тег указываются в «K8sCtx»);
– Secrets (secrets, list) - список секретов, доступных сканеру (указываются их имена, секреты необходимо предварительно загрузить в кластер, в объект с именем «secrets»).

- State (state, text:text) - KV хранилище (text:text) для сохранения состояния задачи.
- K8sCtx (k8s_context, object) - контекст запуска задачи в K8S:

– Addr (addr, text) - IP:port до API кластера;
– Token (token, text) - токен аутентификации в кластере (bearer);
– CaPath (ca_path, text) - путь до CA k8s api;
– KafkaUrl (kafka_url, text) - IP:port Kafka, в который будет отправляться результат работы сканера.
– NameSpace (namespace, text) – пространство имён кластера k8s, в котором запускается сканер.
– Tolerations (tolerations, text) - возможность запуска сканера, на определенных нодах кластера.
– RegistryPath (registry_path, text) - путь до реестра.
– RegistryTag (registry_tag, text) - тег для имеджа.
– MemoryLimit (memory_limit, text) - Ограничения по памяти (в формате подходящем для k8s).

- FailCounter (fail_counter, num) - счётчик количества неудачных повторных запусков задачи.
- Irregular (irregular, bool) – указывает, является ли задача регулярной.

4.3. Исключение – объект, описывающий исключения для запуска сканеров. Индекс: «compot-exclusion». Перечень полей объекта:

- ID (id, uuid) – идентификатор объекта;
- CreatedAt (created_at, data) – время создания объекта;
- Name (name, text) – человекочитаемый идентификатор объекта. Составляется из организации и URL;

- Organization (organization, text) – организация, которой принадлежит объект;
- Scanner (scanner, text:object) – Правило для исключения запуска сканера. Ключ - имя сканера, значение – объект, описывающий исключение:

– Targets (targets, list) - исключения по цели. Список объектов типа «Диапазон» (описание в пункте 4.4.);

– TargetsIds (targets_ids, list) - исключения по ID цели. Список объектов типа «Диапазон» (описание в пункте 4.4.);

– Organizations (organizations, list) - исключения по организации. Список объектов типа «Диапазон» (описание в пункте 4.4.).

4.4. Объекты типа «Диапазон»:

- Value (value, text) – значение;
- TimeRanges (value, list) - список объектов, описывающих временные окна (периоды:

– WeekDay (week_day, num) - день недели;

– LeftHour (left_hour, num) - левое значение часа;

– LeftMinute (left_minute, num) - левое значение минуты;

– RightHour (right_hour, num) - правое значение часа;

– RightMinute (right_minute, num) - правое значение минуты.

5. РАБОТА С ЗАДАЧАМИ

5.1. Добавление задачи:

- Для добавления задачи необходимо создать объект «Задача» в индексе «compot-job»;
- Если сканируемый объект не был в системе, то поле «TargetId» можно оставить пустым, тогда система автоматически создаст сканируемый объект.

5.2. Просмотр статуса задачи:

- Статус задачи записывается в поле статуса объектов из индекса «compot-job».

5.3. Удаление задачи:

- Для удаления задачи необходимо удалить объект из индекса «compot-job» посредством операции DELETE.

5.4. Просмотр результата:

- Для получения актуального состояния объекта необходимо выполнить поисковый запрос к интересуемому шаблону индексов. Тело запроса:

```
{
   "aggs": {
      "unique_id": {
         "aggs": {
            "by_created": {
               "top_hits": {
                  "size": 1,
                  "sort": [
                     {
                        "created_at": {
                           "order": "desc"
                        }
                     }
                  ]
               }
            }
         },
         "terms": {
            "field": "id",
            "size": 2000000000
         }
      }
   },
   "size": 0
}
```