Internet Engineering Task Force (IETF)
Request for Comments: 9244
Category: Standards Track
ISSN: 2070-1721

M. Boucadair, Ed.
Orange
T. Reddy.K, Ed.
Akamai
E. Doron
Radware Ltd.
M. Chen
CMCC
J. Shallow
June 2022

Distributed Denial-of-Service Open Threat Signaling (DOTS) Telemetry

Телеметрия распределенной сигнализации об угрозах DoS-атак

Аннотация

Этот документ нацелен на обогащение протокола сигнального канала распределенной сигнализации об угрозе отказа в обслуживании (Distributed Denial-of-Service Open Threat Signaling или DOTS) различными атрибутами телеметрии для оптимального смягчения распределенных атак на службы (Distributed Denial-of-Service или DDoS). Документ задаёт базовый уровень обычного трафика, атрибуты телеметрии трафика атаки, которые DOTS-клиент может передать своему серверу DOTS в запросе на смягчение последствий атаки, атрибуты смягчения атаки, которые сервер DOTS может передать клиенту, атрибуты телеметрии эффективности, которые клиент может передать серверу DOTS. Атрибуты телеметрии могут способствовать системе смягчения атак при выборе методов защиты от DDoS и эффективном смягчении DDoS-атак.

Документ задаёт два модуля YANG для представления типов сообщений телеметрии DOTS и обмена данными данными об отображениях атак по каналу данных DOTS.

Статус документа

Документ относится к категории Internet Standards Track.

Документ является результатом работы IETF[1] и представляет согласованный взгляд сообщества IETF. Документ прошёл открытое обсуждение и был одобрен для публикации IESG[2]. Дополнительную информацию о стандартах Internet можно найти в разделе 2 в RFC 7841.

Информацию о текущем статусе документа, ошибках и способах обратной связи можно найти по ссылке https://www.rfc-editor.org/info/rfc9244.

Оглавление

[1]Internet Engineering Task Force - комиссия по решению инженерных задач Internet.
[2]Internet Engineering Steering Group - комиссия по инженерным разработкам Internet.

1. Введение

IT-организации и сервис-провайдеры сталкиваются с распределёнными атаками на службы (Distributed Denial-of-Service или DDoS), которые делятся на две большие категории.

1. Атаки на сетевом и транспортном уровне, нацеленные на инфраструктуру жертвы. Такие атаки не обязательно нацелены на подавление фактически предоставляемых услуг и чаще препятствуют пересылке трафика легитимных пользователей различными элементами сети (маршрутизаторами, коммутаторами, межсетевыми экранами, транзитными каналами и т. п.).

   Основой таких атак является передача большого объёма трафика в направлении инфраструктуры жертвы. Обычно объем трафика атаки может составлять от нескольких сотен Мбит/с до сотен Гбит/с и даже Тбит/с. Атаки обычно организуются с использованием бот-сетей (botnet) и рефлекторов для усиления атаки (параграф 3.1 в [RFC4732]), таких как NTP (Network Time Protocol), DNS (Domain Name System), SNMP (Simple Network Management Protocol), SSDP (Simple Service Discovery Protocol).

2. Атаки прикладного уровня нацелены на разные приложения. Типичными примерами служат атаки на HTTP/HTTPS, DNS, SIP (Session Initiation Protocol), SMTP (Simple Mail Transfer Protocol). Однако для таких атак открыты и все прочие приложения на периметре сети, для которых известны номера применяемых портов.

   Атаки на прикладном уровне считаются более сложными и их труднее классифицировать, следовательно - обнаруживать и эффективно ослаблять.

Для усугубления проблем злоумышленники применяют многовекторные атаки, которые состоят из динамических атак сетевого и прикладного уровня и могут включать иную тактику. Таким способом формируется многовекторная атака с разными типами и объёмами, одновременно нацеленными на жертву. Многовекторные атаки сложнее в обнаружении и защите от них. Для защиты от таких атак требуется применять одновременно множество методов ослабления атак. Злоумышленники часто меняют векторы атак сразу после успешного ослабления атаки, заставляя менять используемые методы защиты.

Из сказанного выше следует, что обнаружение и смягчение указанных атак является сложной и запутанной задачей. Для этого нужны всесторонние знания об атрибутах атак и обычном поведении систем, на которые атаки нацелены

(включая обычные картины трафика), а также сведения о текущей и прошлых атаках злоумышленника. Ещё более сложной задачей является получение всей аналитики, требуемой для обнаружения таких атак, с учётом текущих возможностей сбора отчётов.

Протокол сигнального канала Distributed Denial-of-Service Open Threat Signaling (DOTS) [RFC9132] служит для передачи сведений о сетевом ресурсе или сети (части сети), подвергающейся DDoS-атаке. Такая информация передаётся клиентами DOTS одному или нескольким серверам DOTS для принятия соответствующих действий по смягчению последствий со стороны трафика, сочтённого подозрительным. Примеры использования представлены в [RFC8903].

Клиенты DOTS могут интегрироваться с детекторами атак DDoS или элементами сети или системы защиты, которые активно вовлечены в текущие атаки. Среда смягчения атак клиента DOTS определяет, что она больше не способна ослаблять атаки самостоятельно. Это может быть обусловлено нехваткой ресурсов или средств защиты с учётом сложности и интенсивности атак. В таких обстоятельствах клиент DOTS обладает бесценными сведениями о фактических атаках, которые должны быть обработаны его серверами DOTS. Предоставляя клиенту DOTS возможность делиться всеобъемлющими сведениями о происходящей атаке, сервер DOTS может значительно повысить свои возможности в части смягчения атаки. Когда атака обрабатывается средствами смягчения, связанными с сервером DOTS, этот сервер знает способы смягчения атаки. Сервер DOTS может делиться такой информацией с клиентом DOTS, чтобы тот мог лучше оценить используемые методы и средства смягчения атаки.

Клиент DOTS могут передавать серверам советы по смягчению атаки, полученные на основе сведений об атаке, отдавая себе отчёт в том, что серверы могут игнорировать эти советы, как описано в [RFC8612] (Gen-004). Советы передаются по сигнальному каналу DOTS, поскольку каналы данных могут быть недоступны во время атаки. Способы обработки серверами DOTS атрибутов обычного трафика и трафика атак, а также советов по смягчению зависят от реализации.

Клиенты и серверы DOTS могут извлечь выгоду из этих сведений, представляя различные детали информации в соответствующие системы управления, отчётов и порталов.

Этот документ определяет атрибуты телеметрии DOTS, которые могут передаваться между клиентами и серверами DOTS. Эти атрибуты не являются обязательными для протокола сигнального канала DOTS [RFC9132]. Если для агента DOTS не заданы ограничения, он может передать своему партнёру доступные атрибуты телеметрии для оптимизации службы смягчения атак, предоставляемой DOTS. Упомянутая политика может быть согласована, например, при подписке на сервис (это выходит за рамки документа) для указания набора клиентов DOTS, развёрнутых в клиентском домене DOTS, которым разрешено передавать или принимать данные телеметрии.

В параграфе 11.2 задан модуль YANG, дополняющий канал данных DOTS [RFC8783] сведениями, относящимися к деталям атаки. Совместное использование таких деталей в период «бездействия» предназначено для оптимизации обмена данными по сигнальному каналу DOTS.

2. Терминология

Ключевые слова **должно** (MUST), **недопустимо** (MUST NOT), **требуется** (REQUIRED), **нужно** (SHALL), **не следует** (SHALL NOT), **следует** (SHOULD), **не нужно** (SHOULD NOT), **рекомендуется** (RECOMMENDED), **не рекомендуется** (NOT RECOMMENDED), **возможно** (MAY), **необязательно** (OPTIONAL) в данном документе интерпретируются в соответствии с BCP 14 [RFC2119] [RFC8174] тогда и только тогда, когда они выделены шрифтом, как показано здесь.

Читателю следует ознакомиться с терминами, определёнными в [RFC8612].

Телеметрия DOTS определена как набор атрибутов, используемых для характеризации базового уровня обычного трафика, трафика атак и мер по их смягчению, а также других сведений, которые могут помочь в применении контрмер. Телеметрия DOTS - это необязательный набор атрибутов, которые могут передаваться по сигнальному каналу DOTS.

Идентификатор установки телеметрии (Telemetry Setup Identifier или tsid) создаётся клиентом DOTS для однозначного указания данных конфигурации телеметрии DOTS (см. 7.1.2. Передача конфигурации телеметрии DOTS).

Идентификатор телеметрии (Telemetry Identifier или tmid) создаётся клиентом DOTS для однозначного указания данных телеметрии DOTS, передаваемыми до или в процессе смягчения атаки (см. 8.2. От клиентов к серверам DOTS).

«Перекрытие» младшей части tsid (или tmid) указывает младшую часть перекрывающихся запросов телеметрии.

Термин «труба» (pipe) обозначает максимальный уровень трафика, который может получать клиентский домен DOTS. Сопоставление трубы с одним или группой сетевых интерфейсов зависит от развёртывания. Например, каждый межсетевой канал может рассматриваться как отдельная труба, если сервер DOTS размещается у каждого восходящего (upstream) провайдера или все соединения с восходящими провайдарами могут считаться клиентским доменом DOTS одной трубой, если серверы DOTS не размещаются у этих провайдеров.

В документе используются выделенные IANA Enterprise Number, известные как Private Enterprise Numbers и SMI (Structure of Management Information) Network Management Private Enterprise Codes [Private-Enterprise-Numbers].

Значение символов на диаграммах деревьев YANG определено в [RFC8340] и [RFC8791].

В соответствии с соглашениями раздела 2 в [RFC8783] примеры параграфа 8.1.6 используют /restconf как обнаруженный корневой путь RESTCONF API. В этих примерах символ \ в конце строки означает перенос длинной строки в целях форматирования. Предполагается, что такие строки объединяются с удалением символов \, перевода строки и начального пробела в следующей строке.

3. Телеметрия DOTS - обзор и назначение

Своевременная и эффективная передача актуальных данных телеметрии DDoS всем элементам, вовлеченным в процесс смягчения атак, важна и повышает общую эффективность систем смягчения DDoS-атак. Двухсторонний обмен между агентами DOTS необходим для повышения уровня осведомлённости каждой стороны об атаках и усилиях по их смягчению в поддержке высокоэффективных услуг по ослаблению атак.

3.1. Улучшение видимости атак

При передаче запроса на смягчение атаки для клиентов DOTS безусловно полезно сообщить серверам DOTS все сведения о происходящих атаках. Это может происходить в случаях, когда клиенты DOTS запрашивают поддержку у серверов DOTS для защиты от атак, которые уже обнаружены и/или (частично) смягчены.

Если атаки уже обнаружены и классифицированы в клиентском домене DOTS, сервер DOTS и связанные с ним службы смягчения могут проактивно использовать эти сведения для оптимизации предоставления услуг. Важно отметить, что клиентские и серверные домены DOTS используют разные подходы к обнаружению и смягчению атак, что может приводить к разным результатам обнаружения и классификации атак. Служба смягчения DDoS рассматривает сведения об имеющейся атаке от клиентов DOTS как советы и не может полностью полагаться или доверять им.

В дополнение к тому, что сервер DOTS напрямую использует данные телеметрии в качестве оперативных советов, команда обеспечения безопасности сервера DOTS также может получить пользу от данных телеметрии. Основным требованием групп по обеспечению безопасности является осведомлённость об атаках и видимость атак, с которыми нужно работать. Это особенно актуально для происходящих атак, где телеметрия DOTS предоставляет данные о текущем статусе атаки. Даже при возможности автоматического смягчения команды защиты могут применять данные телеметрии DOTS для подготовки смягчения атаки и выделения нужных ресурсов (например, специалистов, ресурсов среды и средств смягчения) для конкретного сервиса. Точно так же персонал на стороне клиента DOTS запрашивает обратную связь для своих запросов защиты, поэтому для серверов DOTS важно делиться с клиентами данными телеметрии DOTS.

Таким образом, взаимный обмен информацией имеет решающее значение для «замыкания контура смягчения» между клиентами и серверами DOTS. Для команд на стороне сервера важно понимать, что атаки, видимые ресурсами сервера по смягчению атак, - это те же атаки, для которых клиент DOTS запрашивает смягчение. Команде на стороне клиента DOTS важно понимать, что предоставляется именно требуемая услуга, например, «я запросил смягчение двух атак, но мой сервер DOTS обнаружил и смягчает лишь одну из них». Несоответствие в классификации атак между клиентами и серверами DOTS можно выявить и, может быть, обработать с помощью атрибутов телеметрии DOTS.

Кроме того, системы управления и оркестровки на серверной и клиентской стороне могут применять телеметрию DOTS в качестве обратной связи для автоматизации действий по управлению и поддержке на основе полученных данных телеметрии.

Если ресурсы защиты сервера DOTS позволяют способствовать телеметрии DOTS, сервер адаптирует свою стратегию защиты и активирует требуемые меры противодействия (включена автоматизация) для принятия оптимизированных решений и действий. Обсуждение интерфейса от сервера DOTS к системе смягчения для передачи данных телеметрии выходит за рамки этого документа.

3.2. Улучшенное обнаружение

Телеметрия DOTS может также служить входными данными для определения значений при настройке параметров, доступных на ресурсах смягчения атак. За последние несколько лет технологии обнаружения DdoS-атак развились от детектирования по порогу (когда весь трафик или его определённые части превышают заданный порог) до обнаружения аномалий. В последнем случае требуется поддерживать строгое изучение обычного поведения, а аномалия (или атака) идентифицируется и классифицируется но основе сведений о нормальном трафике и отклонений от него. Статистические алгоритмы и искусственный интеллект (например, машинное обучение) применяются так, что пороги рассчитываются автоматически путём изучения нормального трафика в период «бездействия» (смягчение не применяется). Полученные характеристики обычного трафика называют также базовым уровнем (normal traffic baseline), а атакой считаются случаи, когда фактический трафик жертвы отличается от базового уровня.

Кроме того, последующие действия по смягчению атак являются значительно более сложными. Способность различать легитимный трафик и атаки на уровне пакетов является сложной задачей. Например, пакет может казаться легитимным, а сигнатура атаки не может быть обнаружена. Аномалии можно выявить лишь после детального статистического анализа. Средства смягчения атак DDoS применяют базовый уровень трафика в процессе смягчения атак для идентификации и классификации ожидаемого появления определённой картины трафика. В частности, средства смягчения используют базовый уровень для определения «степени нормальности», которой нужно достичь в разных процессах смягчения атак.

Расчёт базового уровня выполняется на основе непрерывного изучения нормального поведения защищаемых объектов. Минимальный период изучения варьируется от часов до дней и даже недель, в зависимости от поведения защищаемых приложений. Базовый уровень не может изучаться во время атак, поскольку в этом случае поведение защищаемого объекта отличается от обычного.

Если клиент DOTS рассчитал базовый уровень для защищаемых объектов, передача такой информации серверу DOTS вместе с параметрами трафика атаки представляет ценность. Сервер DOTS с учётом этой телеметрии настраивает свои ресурсы смягчения атаки. Системы смягчения серверов DOTS используют базовый уровень для понимания обычного поведения жертвы атаки и стремятся восстановить нормальную работу жертвы. Предполагается, что смягчение атаки будет в результате более эффективным, точным и без ложных срабатываний или пропусков.

Смягчение атак без знания обычного трафика будет в лучшем случае неточным. Это особенно верно для рекурсивной сигнализации (см. параграф 3.2.3 в [RFC8811]). С учётом возможности интеграции клиентов DOTS в самые разные системы и варианты применения, это повышает важность сведений о поведении каждого клиентского домена DOTS, особенно из-за того, что глобальные пороги обнаружения атак практически не реализуются. Каждый клиентский домен DOTS имеет свой уровень трафика и своё поведение. Без сведений о базовых уровнях серверам DOTS в некоторых случаях может быть очень трудно обнаружить и эффективно смягчить атаки.

> Важно подчеркнуть, что системы смягчения атак серверов DOTS практически не способны рассчитать обычный базовый уровень без полученных заранее сведений о трафике.

Конечно, такие сведения можно предоставить по отдельному каналу (out-of-band) или настроить вручную с риском того, что информация станет неточно результате изменения сети и обычной картины трафика. Применение динамических и кооперативных мер между клиентами и серверами DOTS для идентификации и обмена ключевыми параметрами обеспечит более эффективную защиту от DDoS-атак.

3.3. Эффективное ослабление атак

Во время сильной атаки каналы клиента DOTS могут быть загружены полностью. Клиенты DOTS запрашивают у своих серверов DOTS обработку атаки в восходящем направлении, чтобы вернуть каналам клиента приемлемый уровень загрузки (в идеале, нормальный). На этом этапе важно обеспечить, чтобы система смягчения не перегрузила каналы клиента DOTS, отправляя тому большие объёмы «чистого» или кажущегося чистым трафика. Такое может случиться, если система смягчения не может обнаружить и ослабить все атаки, направленные в клиентский домен DOTS.

В таких случаях для клиентов DOTS важно сообщить серверам DOTS общую пропускную способность каналов, определяющую уровень трафика, который клиентский домен DOTS может воспринять из восходящей сети (провайдера). Необходимо динамически обновлять состояния каналов между агентами DOTS, находящимися в условиях DDoS-атаки (например, при использовании несколькими клиентами DOTS общих физических каналов). Серверу DOTS следует активировать другие механизмы, чтобы гарантированно предотвратить ненамеренное насыщение каналов клиентского домена. Для этого разумным решением будет применение ограничения скорости, описанного в [RFC8783]. Клиент DOTS может указать типы трафика (например, ICMP, UDP, TCP через порт 80), который он предпочитает ограничить. Действиями по ограничению скорости можно управлять по каналу сигнализации [RFC9133] даже при перегрузке «трубы».

4. Внутреннее устройство

4.1. Обзор операций телеметрии

Стек протоколов DOTS делится на две логических части: сигнальный канал [RFC9132] и канал данных [RFC8783]. Это разделение обусловлено совершенно разными требованиями к передаваемому по этим каналам трафику. Сигнальный канал DOTS должен оставаться доступным и пригодным для использования даже в случае атаки, которая, например, может перегрузить одно из направлений охваченных каналов, что делает ненадёжными механизмы на основе подтверждений и настоятельно требует делать сообщения достаточно малыми, чтобы передать в одном пакете IP (параграф 2.2 в [RFC8612]). Напротив, канал данных DOTS доступен для высокоскоростной передачи данных до или после атаки с использованием более традиционных методов доставки (параграф 2.3 в [RFC8612]). Обычно предпочтительно заранее настраивать конфигурацию по каналу данных DOTS, включая настройку псевдонимов для статических или почти статических наборов данных, таких как множество сетевых адресов/префиксов, которые могут быть атакованы. Это помогает оптимизировать использование сигнального канала DOTS для небольших сообщений, которые важно доставить даже во время атаки. Напомним, что для сигнализации и данных DOTS требуются защищённые каналы (раздел 11 в [RFC9132] и раздел 10 в [RFC8783]).

Данные телеметрии имеют аспекты, соответствующие обоим режимам работы (сигнализация и данные). Безусловно необходимо передавать обновляемые сведения о трафике происходящей атаки и целях атаки, чтобы иметь детальные сведения о статусе смягчения и обновлять стратегию защиты от адаптивных атак. Однако полезно также предоставлять службам смягчения картину нормального (базового уровня) трафика в направлении возможных целей атак, чтобы помочь в обнаружении отклонений в поведении трафика при возникновении атак. Кроме того, можно поддерживать «базу данных» с классификацией известных типов атак, чтобы можно было применять краткий идентификатор атаки в период её действия для описания данной атаки. Эта спецификация предусматривает использование канала данных DOTS для последней функции (параграф 8.1.6), но большая часть функций телеметрии реализуется по сигнальному каналу DOTS.

Отметим, что передача сведений о трафике происходящей атаки является функциональным требованием, а сведения о базовом уровне трафика представляют идентичную структуру данных, которая естественным образом определена вместе с описанием атаки. Связанные с этим данные о настройке телеметрии служат для параметризации сведений о фактическом трафике данных, также передаются по каналу управления из соображений целесообразности.

Этот документ задаёт расширение протокола сигнального канала DOTS, установка, поддержка и использование которого заданы в [RFC9132].

После организации сигнального канала DOTS клиенты, поддерживающие телеметрию DOTS, выполняют настройку конфигурации телеметрии (например, интервалы измерения и уведомления, ёмкость трубы, базовый уровень трафика), как описано в разделе 7. Затем агенты DOTS могут включать атрибуты телеметрии DOTS с использованием сигнального канала DOTS (параграф 8.1). Клиент DOTS может использовать раздельные сообщения для обмена со своими серверами DOTS набором данных телеметрии, связанных с текущими мерами смягчения атак (параграф 8.2). Заинтересованный в телеметрических уведомлениях, связанных с некоторыми из его ресурсов, клиент следует процедуре, описанной в параграфе 8.3. Клиент DOTS, получающий такие уведомления, может принять решение об отправке запроса на смягчение атаки, если он не может смягчить её локально внутри клиентского домена DOTS.

Совокупные данные телеметрии DOTS могут также включаться в сообщения об обновлении эффективности (параграф 9.1) или смягчения (параграф 9.2).

4.2. Поблочная передача

Клиенты DOTS могут использовать поблочную передачу [RFC7959] в соответствии с рекомендациями параграфа 4.4.2 в [RFC9132] для управления размером отклика, когда возвращаемые данные не помещаются в одну дейтаграмму.

Клиенты DOTS могут также применять опцию Block1 протокола CoAP (Constrained Application Protocol) в запросах PUT (параграф 2.5 в [RFC7959]) для инициирования больших передач, но эти передачи Block1 скорее всего приведут к отказу, если входная «труба» заполнена, поскольку для передачи требуется сообщение от сервера для каждого блока, которое скорей всего будет потеряно во входном потоке. Необходимо рассмотреть попытку уместить PUT в одну передачу или разделить PUT на несколько дискретных запросов PUT, каждый из которых умещается в один пакет.

Опции Q-Block1 и Q-Block2 похожи на опции CoAP Block1 и Block2, но обеспечивают надёжную передачу больших блоков данных с меньшим числом обменов пакетами, используя сообщения NON, определённые в [RFC9177]. Реализации DOTS могут рассмотреть использование опций Q-Block1 и Q-Block2 [DOTS-Robust-Blocks].

4.3. Многодомные системы DOTS

Вопросы выбора многодомными клиентами DOTS серверов для контакта и префиксов IP для включения в телеметрию для данного партнёрского сервера DOTS рассмотрены в [DOTS-Multihoming]. Например, если каждая из восходящих сетей раскрывает сервер DOTS и клиент DOTS поддерживает каналы DOTS со всеми из них, по каналам DOTS будет передаваться лишь информация, относящаяся к префиксам, назначенным клиентскому домену восходящей сетью.

Соображения, связанные с тем, собирает ли (и как) клиент DOTS ту или иную телеметрию (например, детали атаки), которую он получает от первого сервера DOTS и передаёт её второму серверу, зависят от реализации и развёртывания.

4.4. Вопросы YANG

Сообщения телеметрии между агентами DOTS сериализуются с использованием краткого представления двоичных объектов (Concise Binary Object Representation или CBOR) [RFC8949]. Данные в кодировке CBOR служат для передачи относящихся к каналу сигнализации сообщений, которые содержат параметры запросов и данные откликов, такие как ошибки.

Этот документ задаёт модель YANG [RFC7950] для представления типов сообщений телеметрии DOTS (параграф 11.1). Все параметры в полях данных сигнального канала DOTS отображаются на типы CBOR, как указано в разделе 12. Напомним, что правила отображения данных из модели YANG на CBOR даны в разделе 3 [RFC9132].

Модуль телеметрии (параграф 11.1) не предназначен для использования по протоколам NETCONF (Network Configuration Protocol) и RESTCONF с целью управления серверами DOTS. Он задаёт модель данных и кодирование в соответствии с [RFC8791]. Серверные отклонения (параграф 5.6.3 в [RFC7950]) настоятельно не рекомендуются, поскольку партнерский агент DOTS не может получить список отклонений и могут возникать проблемы совместимости.

Модуль телеметрии DOTS (параграф 11.1) использует enumeration вместо identity для определения единиц, выборок и интервалов, поскольку в ином случае нужно включать идентификатор пространства имён ietf-dots-telemetry при включении атрибута телеметрии (например, при обновлении эффективности смягчения). Применение identity неоптимально с точки зрения компактности сообщений, которая очень важна для сигнального канала DOTS.

Модуль телеметрии DOTS (параграф 11.1) включает списки без оператора key. Это соответствует [RFC8791]. Причина отсутствия ключей состоит в том, что они не включаются в тело запросов DOTS, эти ключи обязательны лишь в запросах Uri-Paths (разделы 7 и 8). Иначе при каждом включении оператора key предполагается такое же определение, как в параграфе 7.8.2 [RFC7950].

Некоторые параметры (например, значения low-percentile) могут быть связаны с разными типами YANG (например, decimal64 и yang:gauge64). Чтобы проще было различать типы этих параметров и сохранить осмысленность имён, применяются суффиксы, показанные в таблице 1.

Таблица 1. Суффиксы и типы YANG.

Суффикс	***Тип YANG***	***Пример***
-g	yang:gauge64	low-percentile-g
-c	container	connection-c
-ps	per second	connection-ps

Диаграмму полного дерева телеметрии DOTS можно создать с помощью pyang [PYANG]. Дерево не включено в документ из-за слишком большого размера (параграф 3.3 а [RFC8340]) и представлено лишь частями.

Для оптимизации обмена данных по сигнальному каналу DOTS в этом документе определён второй модуль YANG (ietf-dots-mapping, параграф 11.2), дополняющий канал данных DOTS [RFC8783]. Это дополнение можно использовать в период «бездействия» для обмена сведениями о сопоставлениях атак (параграф 8.1.5). Клиенты DOTS могут применять инструменты, такие как YANG Library [RFC8525], для получения списка свойств и отклонений, поддерживаемых сервером по каналу данных.

5. Базовые вопросы

5.1. Идентификация клиентов DOTS

В соответствии с правилами параграфа 4.4.1 в [RFC9132] клиент DOTS создаёт уникальный идентификатор cuid для предотвращения конфликтов. Напомним, что параграф 4.4.1.3 в [RFC9132] запрещает возврат cuid в теле откликов.

5.2. Шлюзы DOTS

Между клиентами и серверами DOTS могут размещаться шлюзы DOTS. Необходимо следовать соображениям, изложенным в параграфе 4.4.1 [RFC9132]. В частности, атрибут cdid служит для однозначной идентификации клиентского домена DOTS. Напомним, что параграф 4.4.1.3 в [RFC9132] запрещает возврат cuid (при наличии) в теле откликов.

5.3. Параметры Uri-Path и пустые значения

Параметры Uri-Path и атрибуты с пустыми значениями **недопустимо** включать в запрос. Наличие пустого значения делает всё сообщение недействительным.

5.4. Управление данными конфигурации

Сервер DOTS руководствуется теми же соображениями, которые изложены в параграфе 4.5.3 [RFC9132], для поддержки актуальности и уведомлений конфигурации телеметрии DOTS.

Аналогично, клиент DOTS может управлять выбором конфигурационных и неконфигурационных узлов данных при отправке запроса GET с помощью опции Uri-Query с (content - содержимое), следуя процедуре, заданной в параграфе 4.4.2 [RFC9132]. В последующих параграфах эти соображения не повторяются.

5.5. Проверка сообщения

Полномочными для проверки сообщений, передаваемых по сигнальному каналу DOTS являются разделы 7 - 9 и таблица сопоставлений в разделе 12. Структура тела сообщений представлена в модуле YANG (параграф 11.1).

5.6. Замечания о примерах

Примеры представлены для иллюстрации и не представляют полный набор сообщений.

JSON-кодирование данных модели YANG служит для демонстрации операций телеметрии. Для удобочитаемости в примерах применяются имена параметров и их типы JSON, а не значение ключей CBOR и типы CBOR (см. раздел 12). Эти соглашения заимствованы из [RFC9132].

В примерах применяется значение Enterprise Number = 32473, выделенное для документации (см. [RFC5612]).

6. Пути операций телеметрии

Как отмечено в параграфе 4.2 [RFC9132], каждая операция DOTS указывается суффиксом пути, который задаёт предусмотренную операцию. Путь операции добавляется в конце префикса пути для создания URI, применяемого с запросом CoAP для выполнения нужной операции DOTS. Суффиксы путей телеметрии приведены в таблице 2.

Таблица 2. Операции телеметрии DOTS.

Операция	***Путь операции***	***Описание***
Telemetry Setup	/tm-setup	Раздел 7
Telemetry	/tm	Раздел 8

Модуль YANG ietf-dots-telemetry, заданный в параграфе 11.1, определяет структуру данных для представления новых типов сообщений DOTS - telemetry-setup и telemetry. Структура дерева показана на рисунке 1, а описания даны в разделах 7 и 8 с указанием точной структуры типов сообщений telemetry-setup и telemetry.

```
structure dots-telemetry:
  +-- (telemetry-message-type)?
     +--:(telemetry-setup)
     |  ...
     |  +-- telemetry* []
     |      ...
     |      +-- (setup-type)?
     |         +--:(telemetry-config)
     |         |  ...
     |         +--:(pipe)
     |         |  ...
     |         +--:(baseline)
     |            ...
     +--:(telemetry)
        ...
```

Рисунок 1. Новые типы сообщений DOTS (дерево YANG).

Реализации DOTS **должны** поддерживать Observe Option [RFC7641] для tm (раздел 8).

7. Конфигурация установки телеметрии DOTS

Как показано на рисунке 1, сообщение установки телеметрии DOTS должно включать лишь относящиеся к телеметрии параметры конфигурации (параграф 7.1), сведения о «емкости трубы» клиентского домена DOTS (параграф 7.2) или данные о базовом уровне трафика телеметрии (параграф 7.3). Поэтому запросы, включающие комбинацию настроек телеметрии, ёмкости трубы и данные о базовом уровне трафика, **должны** отвергаться серверами DOTS с кодом отклика 4.00 (Bad Request - недопустимый запрос).

Клиент DOTS может сбросить все установленные данные конфигурации телеметрии DOTS в соответствии с параграфом 7.4.

Сервер DOTS может обнаруживать конфликты при обработке запросов, относящихся к ёмкости трубы клиентского домена DOTS или данным о базовом трафике телеметрии, с запросами от других клиентов из того же домена (см. параграф 7.5).

Конфигурация установки телеметрии привязана к клиентскому домену DOTS. Серверам DOTS **недопустимо** ожидать от клиентов DOTS регулярной отправки запросов на обновление конфигурации телеметрии. Доступная конфигурация установки телеметрии действительна, пока сервер DOTS не прекратит обслуживание клиентского домена DOTS. Серверам DOTS **недопустимо** сбрасывать tsid из-за отказа сессии с клиентом DOTS. Клиенты DOTS обновляют свою конфигурацию установки телеметрии при смене параметров, способных влиять на смягчение атак.

Запросы и отклики конфигурации телеметрии DOTS помечаются как сообщения Confirmable (параграф 2.1 [RFC7252]).

7.1. Конфигурация телеметрии

В телеметрии DOTS применяется несколько значений процентилей для представления общего распределения картины трафика, а не моментального снимка в конкретное время. Моделирование необработанных данных о потоке трафика в форме распределения и описание такого распределения влечёт за собой выбор периода измерений, описываемого распределением, и числа интервалов выборки или «сегментов» (bucket) в этом интервале измерений. Трафик в каждом сегменте считается одним событием (усредняется), а распределение сегментов служит для описания распределения трафика в интервале измерений. Распределение можно характеризовать статистически (например, среднее значение, медиана, стандартное отклонение), а также указанием значений на разных уровнях процентилей рассматриваемого набора данных (например, квартили для 25-го, 50-го и 75-го процентиля). Значения процентилей и их расчёт подробно описаны в параграфе 11.3 [RFC2330].

В телеметрии DOTS применяется 3 значения процентилей и общий пик для описания распределений трафика. Значения для low-percentile, mid-percentile и high-percentile настраиваются. Принятые по умолчанию значения указаны в

параграфе 7.1.2. Клиент DOTS может согласовать с серверами набор используемых параметров конфигурации телеметрии, включая указанные ниже.

- Связанные с процентилями параметры измерений. В частности, measurement-interval задаёт период, в течение которого рассчитываются процентили, а measurement-sample определяет распределение по времени измерений, служащих для расчёта процентилей.
- Единицы измерения.
- Допустимые значения процентилей.
- Интервал уведомлений телеметрии.
- Допустимая телеметрия от сервера.

7.1.1. Извлечение текущей конфигурации телеметрии DOTS

Запрос GET служит для получения приемлемых и текущих параметров телеметрии от сервера DOTS. Запрос может включать Uri-Path cdid при трансляции шлюзом DOTS. Пример запроса (без шлюза) показан на рисунке 2.

```
Header: GET (Code=0.01)
Uri-Path: ".well-known"
Uri-Path: "dots"
Uri-Path: "tm-setup"
Uri-Path: "cuid=dz6pHjaADkaFTbjr0JGBpw"
```

Рисунок 2. GET для извлечение текущей и приемлемой конфигурации телеметрии DOTS.

При получении такого запроса и отсутствии ошибок при его обработке сервер DOTS возвращает отклик 2.05 (Content — содержимое) с параметрами телеметрии, которые подходят для сервера, сведениями о трубе (параграф 7.2) и текущими данными о базовом уровне (параграф 7.3), которые сервер поддерживает для этого клиента DOTS. Структура дерева тела отклика показана на рисунке 3.

Серверы DOTS, поддерживающие передачу клиентам сведений телеметрии до или в процессе смягчения атаки (параграф 9.2) устанавливают для server-originated-telemetry в max-config-values значение true (false в ином случае). Если server-originated-telemetry отсутствует в отклике, это эквивалентно отклику в server-originated-telemetry = false.

```
structure dots-telemetry:
  +-- (telemetry-message-type)?
     +--:(telemetry-setup)
     |  +-- (direction)?
     |  |  +--:(server-to-client-only)
     |  |     +-- max-config-values
     |  |     |  +-- measurement-interval?          interval
     |  |     |  +-- measurement-sample?            sample
     |  |     |  +-- low-percentile?                percentile
     |  |     |  +-- mid-percentile?                percentile
     |  |     |  +-- high-percentile?               percentile
     |  |     |  +-- server-originated-telemetry?   boolean
     |  |     |  +-- telemetry-notify-interval?     uint16
     |  |     +-- min-config-values
     |  |     |  +-- measurement-interval?        interval
     |  |     |  +-- measurement-sample?          sample
     |  |     |  +-- low-percentile?              percentile
     |  |     |  +-- mid-percentile?              percentile
     |  |     |  +-- high-percentile?             percentile
     |  |     |  +-- telemetry-notify-interval?   uint16
     |  |     +-- supported-unit-classes
     |  |     |  +-- unit-config* [unit]
     |  |     |     +-- unit           unit-class
     |  |     |     +-- unit-status    boolean
     |  |     +-- supported-query-type*  query-type
     |  +-- telemetry* []
     |     +-- (direction)?
     |     |  +--:(server-to-client-only)
     |     |     +-- tsid?                  uint32
     |     +-- (setup-type)?
     |        +--:(telemetry-config)
     |        |  +-- current-config
     |        |     +-- measurement-interval?          interval
     |        |     +-- measurement-sample?            sample
     |        |     +-- low-percentile?                percentile
     |        |     +-- mid-percentile?                percentile
     |        |     +-- high-percentile?               percentile
     |        |     +-- unit-config* [unit]
     |        |     |  +-- unit           unit-class
     |        |     |  +-- unit-status    boolean
     |        |     +-- server-originated-telemetry?   boolean
     |        |     +-- telemetry-notify-interval?     uint16
     |        +--:(pipe)
     |        |  ...
     |        +--:(baseline)
     |           ...
     +--:(telemetry)
        ...
```

Рисунок 3. Структура дерева конфигурации телеметрии.

При наличии атрибутов min-config-values и max-config-values значения в max-config-values **должны** быть больше значений в соответствующих атрибутах min-config-values.

7.1.2. Передача конфигурации телеметрии DOTS

Запрос PUT служит для передаси конфигурационных параметров данных телеметрии (например, значений процентилей). К примеру, клиент DOTS может обратиться к своему серверу DOTS для смены принятых по умолчанию значений процентилей, служащих базой для данных телеметрии. На рисунке 3 показаны атрибуты, которые клиент DOTS может установить таким запросом PUT. Пример изменения значений процентилей приведён на рисунке 4.

Примечание. Содержимое сообщения на рисунке 4 представлено в кодировке CBOR, как указано Content-Format application/dots+cbor (см. параграф 10.3 в [RFC9132]). Однако для удобочитаемости этот пример (и другие рисунки, показывающие сообщения телеметрии DOTS) следует параграфу 5.6, используя имена и типы JSON из раздела 12.

```
Header: PUT (Code=0.03)
Uri-Path: ".well-known"
Uri-Path: "dots"
Uri-Path: "tm-setup"
Uri-Path: "cuid=dz6pHjaADkaFTbjr0JGBpw"
Uri-Path: "tsid=123"
Content-Format: "application/dots+cbor"

{
  "ietf-dots-telemetry:telemetry-setup": {
    "telemetry": [
      {
        "current-config": {
          "low-percentile": "5.00",
          "mid-percentile": "65.00",
          "high-percentile": "95.00"
        }
      }
    ]
  }
}
```

Рисунок 4. PUT для доставки конфигурации телеметрии DOTS.

Параметр cuid является обязательным в Uri-Path запросов PUT.

Ниже приведено определение дополнительного параметра Uri-Path.

tsid

Идентификатор установки телеметрии (Telemetry Setup Identifier) служит для представления конфигурационных данных установки телеметрии DOTS целым числом и **должен** создаваться клиентами DOTS. Значения tsid **должны** монотонно возрастать по мере необходимости передачи клиентом новых (а не просто изменённых) параметров конфигурации.

При достижении максимального значения (rollover) параметра tsid **должна** применяться процедура, заданная в параграфе 4.4.1 [RFC9132] для параметра mid.

Этот атрибут является обязательным и **должен** размещаться в опциях Uri-Path после атрибута cuid.

Атрибуты cuid и tsid **недопустимо** включать в тело запросов PUT.

В запросе PUT **должен** присутствовать хотя бы один настраиваемый атрибут.

Запрос PUT с большим значением tsid переопределяет установленные запросом с меньшим tsid данные конфигурации телеметрии DOTS. Чтобы не поддерживать слишком длинные списки tsid для доставки данных конфигурации телеметрии от клиента DOTS, наименьшее значение tsid **должно** автоматически удаляться без доступности серверу.

Сервер DOTS указывает результат обработки запроса PUT с помощью приведённых ниже кодов отклика.

- Если в запросе отсутствуют обязательные атрибуты, не включён параметр Uri-Path cuid или tsid или имеется хотя бы один недействительный или неизвестный параметр, **должен** возвращаться отклик с кодом 4.00 (Bad Request - непригодный запрос).
- Если сервер DOTS не находит в своей конфигурации значение параметра tsid, переданное в запросе PUT, и воспринимает параметры конфигурации, в отклике **должен** указываться код 2.01 (Created - создано).
- Если сервер DOTS находит в своей конфигурации значение параметра tsid, переданное в запросе PUT, и воспринимает параметры конфигурации, в отклике **должен** указываться код 2.04 (Changed - изменено).
- Если какое-либо из включённых значений атрибутов не приемлемо для сервера DOTS (параграф 7.1.1), **должен** возвращаться отклик 4.22 (Unprocessable Entity - необрабатываемый элемент).

  Клиент DOTS может повторить и передать запрос PUT с другими значениями атрибутов, приемлемыми для сервера DOTS.

По умолчанию для представления данных телеметрии служат значения low-percentile (10-й), mid-percentile (50-й), high-percentile (90-й) и peak (100-й). Клиент DOTS может отменить некоторые типы процентилей (low, mid, high). В частности, low-percentile = 0.00 указывает, что клиенту DOTS не нужны значения low-percentile. Аналогично, установка для mid-percentile (или high-percentile) таклго же значения, как для low-percentile (или mid-percentile) указывает, что клиенту не нужны значения mid-percentile (или high-percentile). Например, клиент DOTS может отправить запрос. Показанный на рисунке 5, для информирования сервера о желании получать лишь значения high-percentile. Это предполагает, что клиент будет применять этот тип процентилей при совместном с сервером использовании данных телеметрии.

Клиенты DOTS могут также настраивать класс единиц измерения для относящихся к трафику данных телеметрии, а также другие классы единиц измерения, такие как пакет/сек, бит/сек, байт/сек. Для одного набора данных телеметрии можно применять одновременно классы бит/сек и байт/сек. Однако получение конфликтующих значений считается недействительным параметром и отвергается с кодом 4.00 (Bad Request).

```
Header: PUT (Code=0.03)
Uri-Path: ".well-known"
Uri-Path: "dots"
Uri-Path: "tm-setup"
Uri-Path: "cuid=dz6pHjaADkaFTbjr0JGBpw"
Uri-Path: "tsid=124"
Content-Format: "application/dots+cbor"

{
  "ietf-dots-telemetry:telemetry-setup": {
    "telemetry": [
      {
        "current-config": {
          "low-percentile": "0.00",
          "mid-percentile": "0.00",
          "high-percentile": "95.00"
        }
      }
    ]
  }
}
```

Рисунок 5. PUT для отключения Low и Mid процентилей.

Клиенты DOTS, заинтересованные в получении от сервера данных телеметрии до и в процессе смягчения атак (pre-or-ongoing-mitigation, см. параграф 9.2), **должны** установить для server-originated-telemetry значение true. Отсутствие server-originated-telemetry в запросе PUT эквивалентно установке для атрибута значения false. Пример запроса для включения телеметрии pre-or-ongoing-mitigation от сервера DOTS показан на рисунке 6.

```
Header: PUT (Code=0.03)
Uri-Path: ".well-known"
Uri-Path: "dots"
Uri-Path: "tm-setup"
Uri-Path: "cuid=dz6pHjaADkaFTbjr0JGBpw"
Uri-Path: "tsid=125"
Content-Format: "application/dots+cbor"

{
  "ietf-dots-telemetry:telemetry-setup": {
    "telemetry": [
      {
        "current-config": {
          "server-originated-telemetry": true
        }
      }
    ]
  }
}
```

Рисунок 6. PUT для включения телеметрии до или в процессе смягчения от сервера DOTS.

7.1.3. Извлечение установленной конфигурации телеметрии DOTS

Клиент DOTS может передать сообщение GET с параметром Uri-Path tsid для извлечения текущей конфигурации телеметрии DOTS. Пример такого запроса показан на рисунке 7.

```
Header: GET (Code=0.01)
Uri-Path: ".well-known"
Uri-Path: "dots"
Uri-Path: "tm-setup"
Uri-Path: "cuid=dz6pHjaADkaFTbjr0JGBpw"
Uri-Path: "tsid=123"
```

Рисунок 7. GET для извлечения текущей конфигурации телеметрии DOTS.

Если сервер DOTS не находит полученного значения Uri-Path tsid в своих данных конфигурации, он **должен** передать отклик с кодом 4.04 (Not Found - не найдено).

7.1.4. Удаление конфигурации телеметрии DOTS

Запрос DELETE служит для удаления установленных данных конфигурации телеметрии DOTS (Рисунок 8). Параметры Uri-Path cuid и tsid обязательны для таких запрсов DELETE.

```
Header: DELETE (Code=0.04)
Uri-Path: ".well-known"
Uri-Path: "dots"
Uri-Path: "tm-setup"
Uri-Path: "cuid=dz6pHjaADkaFTbjr0JGBpw"
Uri-Path: "tsid=123"
```

Рисунок 8. Удаление конфигурации телеметрии.

Сервер DOTS сбрасывает конфигурацию телеметрии DOTS в принятые по умолчанию значения и подтверждает запрос клиента DOTS откликом с кодом 2.02 (Deleted - удаллено). Код 2.02 (Deleted) возвращается даже в том случае, когда значение параметра tsid из запроса DELETE отсутствовало в данных конфигурации до запроса.

В параграфе 7.4 описана процедура сброса всех конфигурационных данных установки телеметрии DOTS.

7.2. Общая «ёмкость» трубы

Клиент DOTS может сообщать серверам DOTS сведения о трубе (pipe) в свой домен DOTS. Структура дерева таких сведений показана на рисунке 9.

```
structure dots-telemetry:
  +-- (telemetry-message-type)?
     +--:(telemetry-setup)
     |  ...
     |  +-- telemetry* []
     |     +-- (direction)?
     |     |  +--:(server-to-client-only)
     |     |     +-- tsid?                    uint32
     |     +-- (setup-type)?
     |        +--:(telemetry-config)
     |        |  ...
     |        +--:(pipe)
     |        |  +-- total-pipe-capacity* [link-id unit]
     |        |     +-- link-id     nt:link-id
     |        |     +-- capacity    uint64
     |        |     +-- unit        unit
     |        +--:(baseline)
     |           ...
     +--:(telemetry)
        ...
```

Рисунок 9. Структура дерева данных о «трубе».

Труба клиентского домена DOTS определена как список ограничений на (входящий) трафик (total-pipe-capacity), который может пересылаться в домен по входящим каналам, каждый из которых указывается link-id [RFC8345].

Единицы, применяемые клиентом DOTS при передаче информации о трубе, указываются в атрибуте unit. Клиент DOTS **должен** автоматически приводить значения к соответствующим единицам. Т. е. для данного класса unit клиент DOTS использует наибольшую единицу измерения, дающую значение больше 1. Таким образом, разрешён лишь 1 класс unit.

7.2.1. Передача «ёмкости трубы» клиентского домена DOTS

Применимы соображения параграфа 7.1.2 с одним исключением.

> Относительный порядок двух запросов PUT с атрибутами трубы клиентского домена DOTS от клиента DOTS определяется сравнением значений tsid. Если в двух запросах установки link-id и unit перекрываются, PUT с большим значением tsid будет иметь преимущество. Перекрывающиеся значения с меньшим tsid **должны** автоматически удаляться с утратой доступности.

Клиентам DOTS **следует** минимизировать число активных tsid, используемых для сведений о трубе. Чтобы список не стал слишком большим, клиенту DOTS **рекомендуется** включать в любой запрос на обновление сведений, относящихся к каналу, информацию о других каналах (уже переданную с меньшим tsid). Такое обновление будет переопределять прежние запросы и минимизирует число tsid в запросах от клиента DOTS.

Примечание. Это предполагает, что все сведения о каналах помещаются в одно сообщение.

Пример настройки сведений о каналах для однодомного домена DOTS показан на рисунке 10, где клиент передаёт запрос PUT (Рисунок 11) с ёмкостью канала link1, подключённого к его ISP.

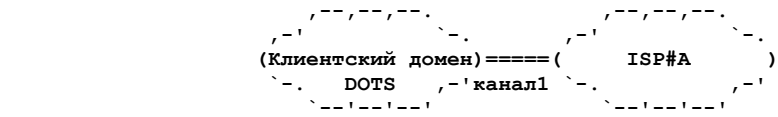

Рисунок 10. Однодомный клиентский домен DOTS.

```
Header: PUT (Code=0.03)
Uri-Path: ".well-known"
Uri-Path: "dots"
Uri-Path: "tm-setup"
Uri-Path: "cuid=dz6pHjaADkaFTbjr0JGBpw"
Uri-Path: "tsid=126"
Content-Format: "application/dots+cbor"

{
  "ietf-dots-telemetry:telemetry-setup": {
    "telemetry": [
      {
        "total-pipe-capacity": [
          {
            "link-id": "link1",
            "capacity": "500",
            "unit": "megabit-ps"
          }
        ]
      }
    ]
  }
}
```

Рисунок 11. Пример запроса PUT для передачи сведений о трубе (однодомной).

Клиентам DOTS можно задать указание совокупных сведений вместо отдельных каналов. Например, клиент с 2 каналами к восходящим ISP (Рисунок 12) может передать запрос PUT (Рисунок 13) для информирования сервера о суммарной пропускной способности каналов к ISP. Информирование об отдельных каналах определяет реализация.

```
Header: PUT (Code=0.03)
Uri-Path: ".well-known"
Uri-Path: "dots"
Uri-Path: "tm-setup"
Uri-Path: "cuid=hmcpH87lmPGsSTjkhXCbin"
Uri-Path: "tsid=896"
Content-Format: "application/dots+cbor"

{
  "ietf-dots-telemetry:telemetry-setup": {
    "telemetry": [
      {
        "total-pipe-capacity": [
          {
            "link-id": "aggregate",
            "capacity": "700",
            "unit": "megabit-ps"
          }
        ]
      }
    ]
  }
}
```

Рисунок 13. Пример запроса PUT для передачи сведений о трубе (агрегат).

Рассмотрим клиентский домен DOTS подключенный к дополнительному ISP (например, ISP#B на рисунке 14). Клиент может информировать сервер DOTS, расположенный вне ISP#A и ISP#B о таком подключении, передавая запрос PUT, показанный на рисунке 15. Этот запрос включает сведения о канале link1, даже если этот канал не обновлён. При получении запроса сервер DOTS удалит запрос с tsid=126 и обновит конфигурацию, включив в неё link1 и link2.

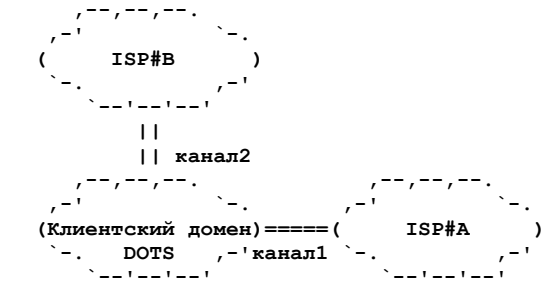

Рисунок 14. Многодомный клиентский домен DOTS.

```
Header: PUT (Code=0.03)
Uri-Path: ".well-known"
Uri-Path: "dots"
Uri-Path: "tm-setup"
Uri-Path: "cuid=dz6pHjaADkaFTbjr0JGBpw"
Uri-Path: "tsid=127"
Content-Format: "application/dots+cbor"

{
  "ietf-dots-telemetry:telemetry-setup": {
    "telemetry": [
      {
        "total-pipe-capacity": [
          {
            "link-id": "link1",
            "capacity": "500",
            "unit": "megabit-ps"
          },
          {
            "link-id": "link2",
            "capacity": "500",
            "unit": "megabit-ps"
          }
        ]
      }
    ]
  }
}
```

Рисунок 15. Пример запроса PUT для передачи сведений о трубе (многодомный).

Клиент DOTS может исключить канал, передав запрос PUT с атрибутом capacity = 0, если для того же домена DOTS другие каналы сохраняются. Например, клиентский домен DOTS может сменить ISP и тогда клиент DOTS сообщает своему серверу DOTS об этом (например, смена конфигурации с рисунка 10 на конфигурацию с рисунка 16) в запросе PUT, показанном на рисунке 17. При получении этого запроса (если при его обработке не будет ошибок) сервер удалит канал link1 из своей конфигурации для этого клиентского домена DOTS. Отметим, что при получении сервером DOTS запроса PUT с capacity 0 для всех каналов он **должен** отклонить запрос с возвратом кода 4.00 (Bad Request). Для удаления всех каналов клиент DOTS может передать запрос DELETE (параграф 7.2.3).

```
Header: PUT (Code=0.03)
Uri-Path: ".well-known"
Uri-Path: "dots"
Uri-Path: "tm-setup"
Uri-Path: "cuid=dz6pHjaADkaFTbjr0JGBpw"
Uri-Path: "tsid=128"
Content-Format: "application/dots+cbor"

{
  "ietf-dots-telemetry:telemetry-setup": {
    "telemetry": [
      {
        "total-pipe-capacity": [
          {
            "link-id": "link1",
            "capacity": "0",
            "unit": "megabit-ps"
          },
          {
            "link-id": "link2",
            "capacity": "500",
            "unit": "megabit-ps"
          }
        ]
      }
    ]
  }
}
```

Рисунок 17. Пример запроса PUT для передачи сведений о трубе (многодомный).

7.2.2. Извлечение сведений о ёмкости установленной трубы клиентского домена DOTS

Запрос GET с параметром Uri-Path tsid служит для извлечения конкретных сведений об установленной трубе клиентского домена DOTS в соответствии с процедурой, описанной в параграфе 7.1.3.

Для получения всех данных о трубе, связанных с клиентом DOTS клиент выполняет процедуру из параграфа 7.1.1.

7.2.3. Удаление установленной ёмкости трубы клиентского домена

Запрос DELETE служит для удаления конкретных сведений об установленной трубе клиентского домена DOTS по процедуре, описанной в параграфе 7.1.4.

7.3. Базовый уровень телеметрии

Клиент DOTS может сообщать серверам DOTS базовый уровень трафика и пропускную способность соединений.

Total traffic normal baseline - общий базовый уровень трафика

Данные о суммарном базовом уровне трафика обеспечивают значения процентилей, представляющие общий базовый уровень трафика. Их можно представить для цели с использованием total-traffic-normal.

Нормальный уровень трафика по протоколам (total-traffic-normal-per-protocol) представляется для цели и зависит от транспортного протокола.

Нормальный уровень трафика по номерам портов (total-traffic-normal-per-port) представляется для каждого порта, связанного с целью.

Если клиент DOTS согласует значения процентилей и единицы измерения (параграф 7.1), эти согласованные параметры применяются взамен принятых по умолчанию. Для каждого используемого класса единиц клиент DOTS **должен** обеспечивать автоматическое приведение к соответствующим единицам.

Total connection capacity - общая пропускная способность соединений

Если цель подвержена истощающим ресурсы DDoS-атакам, полезны указанные ниже атрибуты на уровне транспортного протокола для обнаружения таких DdoS-атак.

- Максимальное число разрешённых для цели одновременных соединений.
- Максимальное число разрешённых для цели одновременных соединений на клиента.
- Максимальное число разрешённых для цели «эмбриональных» соединений. Этим термином обозначают соединения, в которых ещё не завершено согласование. Такие соединения возможны лишь в ориентированных на соединения протоколах, таких как TCP или (SCTP) [RFC9260].
- Максимальное число разрешённых для цели «эмбриональных» соединений на клиента.
- Максимальное число разрешённых для цели соединений в секунду.
- Максимальное число разрешённых для цели соединений в секунду на клиента.
- Максимальное число разрешённых для цели запросов (например, HTTP/DNS/SIP) в секунду.
- Максимальное число разрешённых для цели запросов в секунду на клиента.
- Максимальное число разрешённых для цели остающихся неполных запросов. Атаки на основе неполных запросов создают соединения с жертвой, но не передают полного запроса (например, HTTP).
- Максимальное число разрешённых для цели остающихся неполных запросов на клиента.

Совокупные данные для транспортного протокола выражаются в total-connection-capacity, а возможности порта - в total-connection-capacity-per-port.

Отметим, что целевой ресурс указывается с использованием атрибутов target-prefix, target-port-range, target-protocol, target-fqdn, target-uri, alias-name, как указано в параграфе 4.4.1.1 [RFC9132].

Структура дерева для базового уровня трафика показана на рисунке 18.

```
structure dots-telemetry:
  +-- (telemetry-message-type)?
     +--:(telemetry-setup)
     |  ...
     |  +-- telemetry* []
     |     +-- (direction)?
     |     |  +--:(server-to-client-only)
     |     |     +-- tsid?                  uint32
     |     +-- (setup-type)?
     |        +--:(telemetry-config)
     |        |  ...
     |        +--:(pipe)
     |        |  ...
     |        +--:(baseline)
     |           +-- baseline* [id]
     |              +-- id                       uint32
     |              +-- target-prefix*
     |              |       inet:ip-prefix
     |              +-- target-port-range* [lower-port]
     |              |  +-- lower-port    inet:port-number
     |              |  +-- upper-port?   inet:port-number
     |              +-- target-protocol*                        uint8
     |              +-- target-fqdn*
     |              |       inet:domain-name
     |              +-- target-uri*
     |              |       inet:uri
     |              +-- alias-name*
     |              |       string
     |              +-- total-traffic-normal* [unit]
     |              |  +-- unit                  unit
     |              |  +-- low-percentile-g?     yang:gauge64
     |              |  +-- mid-percentile-g?     yang:gauge64
     |              |  +-- high-percentile-g?    yang:gauge64
     |              |  +-- peak-g?               yang:gauge64
     |              +-- total-traffic-normal-per-protocol*
     |              |       [unit protocol]
     |              |  +-- protocol              uint8
     |              |  +-- unit                  unit
     |              |  +-- low-percentile-g?     yang:gauge64
     |              |  +-- mid-percentile-g?     yang:gauge64
     |              |  +-- high-percentile-g?    yang:gauge64
     |              |  +-- peak-g?               yang:gauge64
     |              +-- total-traffic-normal-per-port* [unit port]
     |              |  +-- port                  inet:port-number
     |              |  +-- unit                  unit
     |              |  +-- low-percentile-g?     yang:gauge64
     |              |  +-- mid-percentile-g?     yang:gauge64
     |              |  +-- high-percentile-g?    yang:gauge64
     |              |  +-- peak-g?               yang:gauge64
     |              +-- total-connection-capacity* [protocol]
     |              |  +-- protocol                        uint8
     |              |  +-- connection?                     uint64
     |              |  +-- connection-client?              uint64
     |              |  +-- embryonic?                      uint64
     |              |  +-- embryonic-client?               uint64
     |              |  +-- connection-ps?                  uint64
     |              |  +-- connection-client-ps?           uint64
     |              |  +-- request-ps?                     uint64
     |              |  +-- request-client-ps?              uint64
     |              |  +-- partial-request-max?            uint64
     |              |  +-- partial-request-client-max?     uint64
     |              +-- total-connection-capacity-per-port*
     |                      [protocol port]
     |                 +-- port
     |                 |       inet:port-number
     |                 +-- protocol                       uint8
     |                 +-- connection?                    uint64
     |                 +-- connection-client?             uint64
     |                 +-- embryonic?                     uint64
     |                 +-- embryonic-client?              uint64
     |                 +-- connection-ps?                 uint64
     |                 +-- connection-client-ps?          uint64
     |                 +-- request-ps?                    uint64
     |                 +-- request-client-ps?             uint64
     |                 +-- partial-request-max?           uint64
     |                 +-- partial-request-client-max?    uint64
     +--:(telemetry)
        ...
```

Рисунок 18. Структура дерева базового уровня телеметрии.

Клиент DOTS может использовать один или несколько базовых уровней трафика (например, совокупный или по префиксам), каждый из которых указывается в клиентском домене DOTS идентификатором id, позволяющим обновить базовый уровень, удалить конкретную запись и т. п.

7.3.1. Передача сведений о базовом уровне клиентского домена DOTS

Здесь применимы соображения из параграфа 7.1.2 с одним исключением.

> Относительный порядок двух запросов PUT с атрибутами базового уровня клиентского домена DOTS от клиента DOTS определяется сравнением значений tsid. Если в двух запросах установки цели перекрываются, PUT с большим значением tsid будет иметь преимущество. Перекрывающиеся значения с меньшим tsid должны автоматически удаляться с утратой доступности.

Два запроса PUT от клиента DOTS имеют перекрывающиеся цели если у них общие адреса или префиксы IP, FQDN, URI или псевдонимы. Кроме того, цели запросов PUT от клиента DOTS имеют пересекающиеся цели с точки зрения сервера DOTS, если адреса, связанные с FQDN, URI, псевдонимами перекрываются между собой или с target-prefix.

Клиентам DOTS **следует** минимизировать число активных tsid, используемых для сведений о трубе. Чтобы список не стал слишком большим, клиенту DOTS **рекомендуется** включать в любой запрос на обновление сведений, относящихся к цели, информацию о других целях (уже переданную с меньшим tsid). Такое обновление будет переопределять прежние запросы и минимизирует число tsid в запросах от клиента DOTS.

Если в запросе нет атрибута target это указывает применение данных о базовом уровне ко всему клиентскому домену.

Пример запроса PUT для передачи сведения о базовом уровне приведён на рисунке 19.

```
Header: PUT (Code=0.03)
Uri-Path: ".well-known"
Uri-Path: "dots"
Uri-Path: "tm-setup"
Uri-Path: "cuid=dz6pHjaADkaFTbjr0JGBpw"
Uri-Path: "tsid=129"
Content-Format: "application/dots+cbor"

{
  "ietf-dots-telemetry:telemetry-setup": {
    "telemetry": [
      {
        "baseline": [
          {
            "id": 1,
            "target-prefix": [
              "2001:db8:6401::1/128",
              "2001:db8:6401::2/128"
            ],
            "total-traffic-normal": [
              {
                "unit": "megabit-ps",
                "peak-g": "60"
              }
            ]
          }
        ]
      }
    ]
  }
}
```

Рисунок 19. Запрос PUT для передачи сведений о базовом уровне DOTS.

Клиенты DOTS могут совместно использовать связанные с протоколом данные, как показано на рисунке 20.

```
Header: PUT (Code=0.03)
Uri-Path: ".well-known"
Uri-Path: "dots"
Uri-Path: "tm-setup"
Uri-Path: "cuid=dz6pHjaADkaFTbjr0JGBpw"
Uri-Path: "tsid=130"
Content-Format: "application/dots+cbor"

{
  "ietf-dots-telemetry:telemetry-setup": {
    "telemetry": [
      {
        "baseline": [
          {
            "id": 1,
            "target-prefix": [
              "2001:db8:6401::1/128",
              "2001:db8:6401::2/128"
            ],
            "total-traffic-normal-per-protocol": [
              {
                "unit": "megabit-ps",
                "protocol": 6,
                "peak-g": "50"
              },
              {
                "unit": "megabit-ps",
                "protocol": 17,
                "peak-g": "10"
              }
            ]
          }
        ]
      }
    ]
  }
}
```

Рисунок 20. Запрос PUT для передачи сведений о базовом уровне DOTS (2).

Сведения о базовом уровне трафика следует обновлять для отражения легитимных перегрузок (например, большое скопление людей), чтобы предотвратить ненужные «смягчения атак».

7.3.2. Получение сведений об установленном базовом уровне трафика

Запрос GET с параметром Uri-Path tsid служит для извлечения сведений об установленном в домене базовом уровне. Применяется такая же процедура, как описано а параграфе 7.1.3. Для получения сведений обо всех базовых уровнях, связанных с клиентом DOTS, клиент DOTS выполняет процедуру, описанную в параграфе 7.1.1.

7.3.3. Удаление установленных сведений о базовом уровне трафика

Запрос DELETE служит для удаления сведений об установленном в клиентском домене DOTS обычном уровне трафика. Процедура описана в параграфе 7.1.4.

7.4. Сброс установленной телеметрии

При загрузке (перезагрузке или ином событии, которое может изменить установки клиента DOTS) клиент **может** передать запрос DELETE для установки принятых по умолчанию значений параметров телеметрии. Такие запросы не включают параметр tsid. Пример запроса показан на рисунке 21.

```
Header: DELETE (Code=0.04)
Uri-Path: ".well-known"
Uri-Path: "dots"
Uri-Path: "tm-setup"
Uri-Path: "cuid=dz6pHjaADkaFTbjr0JGBpw"
```

Рисунок 21. Удаление конфигурации телеметрии.

7.5. Конфликт с другими клиентами DOTS из того же домена

Сервер DOTS может сталкиваться с конфликтами запросов, содержащих сведения о трубе или базовом уровне от разных клиентов одного клиентского домена DOTS. Для уведомления клиентов о конфликте применяется атрибут conflict-information, следуя рекомендациям по устранению конфликта, подобным описанным в параграфе 4.4.1 [RFC9132]. В качестве причины конфликта может быть указано одно из двух значений:

1 - перекрытие целей (параграф 4.4.1 в [RFC9132]);
5 - перекрытие области действия трубы (раздел 13).

8. Телеметрия DOTS до и во время смягчения атак

Имеется два обширных класса атак DDoS: атаки с расходом пропускной способности и ресурсов цели. В этом разделе описаны атрибуты телеметрии DOTS (параграф 8.1), охватывающие оба типа атак. Эти атрибуты предназначены для обеспечения полных сведений об атаках и различных аспектов, наиболее полно характеризующих атаки.

В модуле ietf-dots-telemetry (параграф 11.1) задана структура данных нового типа сообщений telemetry (Рисунок 22).

```
structure dots-telemetry:
  +-- (telemetry-message-type)?
     +--:(telemetry-setup)
     |  ...
     |  +-- telemetry* []
     |     +-- (direction)?
     |     |  +--:(server-to-client-only)
     |     |     +-- tsid?                    uint32
     |     +-- (setup-type)?
     |        +--:(telemetry-config)
     |        |  ...
     |        +--:(pipe)
     |        |  ...
     |        +--:(baseline)
     |           ...
     +--:(telemetry)
        +-- pre-or-ongoing-mitigation* []
           +-- (direction)?
           |  +--:(server-to-client-only)
           |     +-- tmid?                       uint32
           +-- target
           |  ...
           +-- total-traffic* [unit]
           |  ...
           +-- total-traffic-protocol* [unit protocol]
           |  ...
           +-- total-traffic-port* [unit port]
           |  ...
           +-- total-attack-traffic* [unit]
           |  ...
           +-- total-attack-traffic-protocol* [unit protocol]
           |  ...
           +-- total-attack-traffic-port* [unit port]
           |  ...
           +-- total-attack-connection-protocol* [protocol]
           |  ...
           +-- total-attack-connection-port* [protocol port]
           |  ...
           +-- attack-detail* [vendor-id attack-id]
              ...
```

Рисунок 22. Структура дерева типов сообщений.

Атрибуты телеметрии до и во время атаки указываются суффиксом пути /tm, который добавляется после префикса пути для формирования URI, применяемого с запросом CoAP для сигналов телеметрии DOTS. Атрибуты телеметрии pre-or-ongoing-mitigation, указанные в параграфе 8.1, могут передаваться между агентами DOTS и их могут передавать клиенты и серверы DOTS.

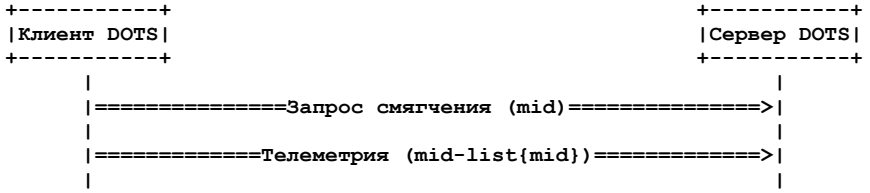

Рисунок 23. Пример запроса сопоставления с использованием mid.

Агентам DOTS **следует** привязывать атрибуты pre-or-ongoing-mitigation к запросам на смягчение, связанным с атакованными ресурсами. В частности, запрос PUT после запроса смягчения может включать ссылку на запрос смягчения (mid-list) как показано на рисунке 23. Пример сопоставления запросов через target-prefix дан на рисунке 24.

Большинство данных телеметрии pre-or-ongoing-mitigation использует единицы измерения, относящиеся к классу unit, заданному процедурой, описанной в параграфе 7.1.2. При генерации данных телеметрии агент DOTS должен автоматически приводить данные к корректным единицам измерения.

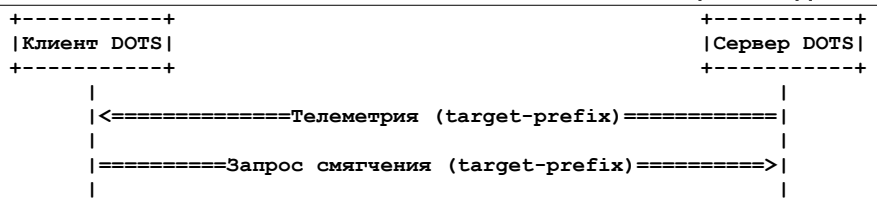

Рисунок 24. Пример запроса сопоставления с использованием target-prefix.

Агентам DOTS **недопустимо** передавать уведомления телеметрии pre-or-ongoing-mitigation одному партнёру чаще, чем 1 раз за telemetry-notify-interval (параграф 7.1). Если уведомление телеметрии передаётся в режиме подобном блочному (например,, [RFC9177]), этому правилу ограничения скорости **недопустимо** считать блоки отдельными сообщениями.

Запросы и отклики телеметрии DOTS до и во время смягчения атак **должны** помечаться как неподтверждаемые сообщения (параграф 2.1 в [RFC7252]).

8.1. Атрибуты телеметрии Pre-or-Ongoing-Mitigation

В разделе 3 описаны мотивы применения атрибутов телеметрии DOTS, описанных в последующих параграфах.

8.1.1. Цель

Ресурс target (Рисунок 25) идентифицируется с использованием атрибутов target-prefix, target-port-range, target-protocol, target-fqdn, target-uri, alias-name или указателя на запрос смягчения (mid-list).

```
+--:(telemetry)
   +-- pre-or-ongoing-mitigation* []
      +-- (direction)?
      |  +--:(server-to-client-only)
      |     +-- tmid?                      uint32
      +-- target
      |  +-- target-prefix*       inet:ip-prefix
      |  +-- target-port-range* [lower-port]
      |  |  +-- lower-port    inet:port-number
      |  |  +-- upper-port?   inet:port-number
      |  +-- target-protocol*     uint8
      |  +-- target-fqdn*         inet:domain-name
      |  +-- target-uri*          inet:uri
      |  +-- alias-name*          string
      |  +-- mid-list*            uint32
      +-- total-traffic* [unit]
      |  ...
      +-- total-traffic-protocol* [unit protocol]
      |  ...
      +-- total-traffic-port* [unit port]
      |  ...
      +-- total-attack-traffic* [unit]
      |  ...
      +-- total-attack-traffic-protocol* [unit protocol]
      |  ...
      +-- total-attack-traffic-port* [unit port]
      |  ...
      +-- total-attack-connection-protocol* [protocol]
      |  ...
      +-- total-attack-connection-port* [protocol port]
      |  ...
      +-- attack-detail* [vendor-id attack-id]
         ...
```

Рисунок 25. Структура дерева цели.

В определении цели **должен** присутствовать хотя бы один из атрибутов target-prefix, target-fqdn, target-uri, alias-name, mid-list.

Если цель восприимчива к атакам на пропускную способность, включаются атрибуты, представляющие значения процентилей трафика атаки attack-id. Для целей, восприимчивых к DDoS-атакам с потреблением ресурсов, атаки можно представлять с помощью атрибутов, определённых в параграфе 8.1.4.

В сообщении телеметрии DOTS **должен** присутствовать хотя бы атрибут target и ещё один иной атрибут pre-or-ongoing-mitigation.

8.1.2. Суммарный трафик

Атрибут total-traffic (Рисунок 26) передаёт значения процентилей (включая пиковое и наблюдаемые в данный момент) для общего наблюдаемого трафика. Более детальные сведения о суммарном трафике можно передать в атрибутах total-traffic-protocol и total-traffic-port. Атрибут total-traffic-protocol представляет суммарный трафик для цели и зависит от транспортного протокола, атрибут total-traffic-port представляет суммарный трафик для номера порта цели.

```
      +--:(telemetry)
         +-- pre-or-ongoing-mitigation* []
            +-- (direction)?
            |  +--:(server-to-client-only)
            |     +-- tmid?                        uint32
            +-- target
            |  ...
            +-- total-traffic* [unit]
            |  +-- unit                  unit
            |  +-- low-percentile-g?     yang:gauge64
            |  +-- mid-percentile-g?     yang:gauge64
            |  +-- high-percentile-g?    yang:gauge64
            |  +-- peak-g?               yang:gauge64
            |  +-- current-g?            yang:gauge64
            +-- total-traffic-protocol* [unit protocol]
            |  +-- protocol              uint8
            |  +-- unit                  unit
            |  +-- low-percentile-g?     yang:gauge64
            |  +-- mid-percentile-g?     yang:gauge64
            |  +-- high-percentile-g?    yang:gauge64
            |  +-- peak-g?               yang:gauge64
            |  +-- current-g?            yang:gauge64
            +-- total-traffic-port* [unit port]
            |  +-- port                  inet:port-number
            |  +-- unit                  unit
            |  +-- low-percentile-g?     yang:gauge64
            |  +-- mid-percentile-g?     yang:gauge64
            |  +-- high-percentile-g?    yang:gauge64
            |  +-- peak-g?               yang:gauge64
            |  +-- current-g?            yang:gauge64
            +-- total-attack-traffic* [unit]
            |  ...
            +-- total-attack-traffic-protocol* [unit protocol]
            |  ...
            +-- total-attack-traffic-port* [unit port]
            |  ...
            +-- total-attack-connection-protocol* [protocol]
            |  ...
            +-- total-attack-connection-port* [protocol port]
            |  ...
            +-- attack-detail* [vendor-id attack-id]
               ...
```

Рисунок 26. Структура дерева суммарного трафика.

8.1.3. Суммарный трафик атаки

Атрибут total-attack-traffic (Рисунок 27) указывает суммарный наблюдаемый трафик атаки. Более детальные сведения могут передавать атрибуты total-attack-traffic-protocol и total-attack-traffic-port. Атрибут total-attack-traffic-protocol представляет суммарный трафик атаки для цели и зависит от транспортного протокола, атрибут total-attack-traffic-port представляет суммарный трафик атаки для номера порта цели.

```
      +--:(telemetry)
         +-- pre-or-ongoing-mitigation* []
            +-- (direction)?
            |  +--:(server-to-client-only)
            |     +-- tmid?                        uint32
            +-- target
            |  ...
            +-- total-traffic* [unit]
            |  ...
            +-- total-traffic-protocol* [unit protocol]
            |  ...
            +-- total-traffic-port* [unit port]
            |  ...
            +-- total-attack-traffic* [unit]
            |  +-- unit                  unit
            |  +-- low-percentile-g?     yang:gauge64
            |  +-- mid-percentile-g?     yang:gauge64
            |  +-- high-percentile-g?    yang:gauge64
            |  +-- peak-g?               yang:gauge64
            |  +-- current-g?            yang:gauge64
            +-- total-attack-traffic-protocol* [unit protocol]
            |  +-- protocol              uint8
            |  +-- unit                  unit
            |  +-- low-percentile-g?     yang:gauge64
            |  +-- mid-percentile-g?     yang:gauge64
            |  +-- high-percentile-g?    yang:gauge64
            |  +-- peak-g?               yang:gauge64
            |  +-- current-g?            yang:gauge64
            +-- total-attack-traffic-port* [unit port]
            |  +-- port                  inet:port-number
            |  +-- unit                  unit
            |  +-- low-percentile-g?     yang:gauge64
            |  +-- mid-percentile-g?     yang:gauge64
            |  +-- high-percentile-g?    yang:gauge64
            |  +-- peak-g?               yang:gauge64
            |  +-- current-g?            yang:gauge64
            +-- total-attack-connection-protocol* [protocol]
            |  ...
            +-- total-attack-connection-port* [protocol port]
            |  ...
            +-- attack-detail* [vendor-id attack-id]
               ...
```

Рисунок 27. Структура дерева суммарного трафика атаки.

8.1.4. Суммарные соединения атаки

Если цель восприимчива к DDoS-атакам на поглощение ресурсов, атрибут total-attack-connection-protocol служит для передачи значений процентилей (включая пиковое и наблюдаемые текущие значения) различных атрибутов,

связанных с общим числом соединений атаки. Ниже перечислены субатрибуты для цели по транспортным протоколам, представляющие характеристики атаки.

- Число одновременных соединений с целью.
- Число одновременных эмбриональных соединений с целью.
- Число соединений с целью за секунду.
- Число запросов для цели за секунду.
- Число неполных запросов для цели.

Общее число соединений на порт представляется атрибутом total-attack-connection-port.

```
+--:(telemetry)
   +-- pre-or-ongoing-mitigation* []
      +-- (direction)?
      |  +--:(server-to-client-only)
      |     +-- tmid?                        uint32
      +-- target
      |  ...
      +-- total-traffic* [unit]
      |  ...
      +-- total-traffic-protocol* [unit protocol]
      |  ...
      +-- total-traffic-port* [unit port]
      |  ...
      +-- total-attack-traffic* [unit]
      |  ...
      +-- total-attack-traffic-protocol* [unit protocol]
      |  ...
      +-- total-attack-traffic-port* [unit port]
      |  ...
      +-- total-attack-connection-protocol* [protocol]
      |  +-- protocol               uint8
      |  +-- connection-c
      |  |  +-- low-percentile-g?    yang:gauge64
      |  |  +-- mid-percentile-g?    yang:gauge64
      |  |  +-- high-percentile-g?   yang:gauge64
      |  |  +-- peak-g?              yang:gauge64
      |  |  +-- current-g?           yang:gauge64
      |  +-- embryonic-c
      |  |  +-- low-percentile-g?    yang:gauge64
      |  |  +-- mid-percentile-g?    yang:gauge64
      |  |  +-- high-percentile-g?   yang:gauge64
      |  |  +-- peak-g?              yang:gauge64
      |  |  +-- current-g?           yang:gauge64
      |  +-- connection-ps-c
      |  |  +-- low-percentile-g?    yang:gauge64
      |  |  +-- mid-percentile-g?    yang:gauge64
      |  |  +-- high-percentile-g?   yang:gauge64
      |  |  +-- peak-g?              yang:gauge64
      |  |  +-- current-g?           yang:gauge64
      |  +-- request-ps-c
      |  |  +-- low-percentile-g?    yang:gauge64
      |  |  +-- mid-percentile-g?    yang:gauge64
      |  |  +-- high-percentile-g?   yang:gauge64
      |  |  +-- peak-g?              yang:gauge64
      |  |  +-- current-g?           yang:gauge64
      |  +-- partial-request-c
      |     +-- low-percentile-g?    yang:gauge64
      |     +-- mid-percentile-g?    yang:gauge64
      |     +-- high-percentile-g?   yang:gauge64
      |     +-- peak-g?              yang:gauge64
      |     +-- current-g?           yang:gauge64
      +-- total-attack-connection-port* [protocol port]
      |  +-- protocol               uint8
      |  +-- port                   inet:port-number
      |  +-- connection-c
      |  |  +-- low-percentile-g?    yang:gauge64
      |  |  +-- mid-percentile-g?    yang:gauge64
      |  |  +-- high-percentile-g?   yang:gauge64
      |  |  +-- peak-g?              yang:gauge64
      |  |  +-- current-g?           yang:gauge64
      |  +-- embryonic-c
      |  |  +-- low-percentile-g?    yang:gauge64
      |  |  +-- mid-percentile-g?    yang:gauge64
      |  |  +-- high-percentile-g?   yang:gauge64
      |  |  +-- peak-g?              yang:gauge64
      |  |  +-- current-g?           yang:gauge64
      |  +-- connection-ps-c
      |  |  +-- low-percentile-g?    yang:gauge64
      |  |  +-- mid-percentile-g?    yang:gauge64
      |  |  +-- high-percentile-g?   yang:gauge64
      |  |  +-- peak-g?              yang:gauge64
      |  |  +-- current-g?           yang:gauge64
      |  +-- request-ps-c
      |  |  +-- low-percentile-g?    yang:gauge64
      |  |  +-- mid-percentile-g?    yang:gauge64
      |  |  +-- high-percentile-g?   yang:gauge64
      |  |  +-- peak-g?              yang:gauge64
      |  |  +-- current-g?           yang:gauge64
      |  +-- partial-request-c
      |     +-- low-percentile-g?    yang:gauge64
      |     +-- mid-percentile-g?    yang:gauge64
      |     +-- high-percentile-g?   yang:gauge64
      |     +-- peak-g?              yang:gauge64
      |     +-- current-g?           yang:gauge64
      +-- attack-detail* [vendor-id attack-id]
         ...
```

Рисунок 28. Структура дерева соединений атаки.

8.1.5. Детали атаки

Этот атрибут (Рисунок 29) служит для передачи подробных характеристик атаки. Субатрибуты происходящих атак перечислены ниже.

vendor-id
Идентификатор производитель (систем защиты) в форме номера предприятия из реестра IANA Private Enterprise Numbers [Private-Enterprise-Numbers].

attack-id
Уникальный идентификатор, присвоенный атаке производителем. Этот параметр **должен** присутствовать независимо от наличия attack-description.

description-lang
Указывает тег языка, используемого в тексте атрибута attack-description. Кодирование атрибута определяется правилами параграфа 2.1 в [RFC5646]. По умолчанию применяется кодировка en-US.

attack-description
Текстовое описание атаки, относящееся скорее к классу атак, нежели к конкретному экземпляру. Методы обработки естественных языков (например, встраивание слов) могут оказаться полезными для сопоставления описания атаки с её типом. Текстовое представление атаки решает две задачи, избавляя от необходимости (a) создавать вручную таблицы сопоставления между производителями и (b) стандартизировать типы атак, которые со временем меняются.

attack-severity
Уровень серьёзности атаки (значения определены в параграфе 3.12.2 [RFC7970]).

start-time
Время начала атаки в секундах с 1970-01-01T00:00Z (параграф 3.4.2 в [RFC8949]). Кодирование CBOR изменено и ведущий тег 1 (дата и время на основе эпохи) **должен** быть опущен.

end-time
Время завершения атаки в секундах с 1970-01-01T00:00Z (параграф 3.4.2 в [RFC8949]). Кодирование CBOR изменено и ведущий тег 1 (дата и время на основе эпохи) **должен** быть опущен.

source-count
Цисло источников, вовлечённых в нацеленную на жертву атаку.

top-talker
Список источников, вовлечённых в атаку и генерирующих важную часть трафика атаки. Источники представляются с помощью source-prefix.
Атрибут spoofed-status указывает, использует ли источник фиктивный адрес IP (например, в атаке с отражением). Если узел spoofed-status не включён, это означает, что статус подмены адреса не известен.
Если цель подвергается атаке на поглощение пропускной способности, включается статистический профиль атаки для каждого из основных участников (total-attack-traffic, см. параграф 8.1.3).
Если цель подвергается DDoS-атаке на поглощение ресурсов, применимы заданные в параграфе 8.1.4 атрибуты для характеризации по участникам атаки.

```
      +--:(telemetry)
         +-- pre-or-ongoing-mitigation* []
            +-- (direction)?
            |  +--:(server-to-client-only)
            |     +-- tmid?                      uint32
            +-- target
            |  ...
            +-- total-traffic* [unit]
            |  ...
            +-- total-traffic-protocol* [unit protocol]
            |  ...
            +-- total-traffic-port* [unit port]
            |  ...
            +-- total-attack-traffic* [unit]
            |  ...
            +-- total-attack-traffic-protocol* [unit protocol]
            |  ...
            +-- total-attack-traffic-port* [unit port]
            |  ...
            +-- total-attack-connection-protocol* [protocol]
            |  ...
            +-- total-attack-connection-port* [protocol port]
            |  ...
            +-- attack-detail* [vendor-id attack-id]
               +-- vendor-id             uint32
               +-- attack-id             uint32
               +-- description-lang?     string
               +-- attack-description?   string
               +-- attack-severity?      attack-severity
               +-- start-time?           uint64
               +-- end-time?             uint64
               +-- source-count
               |  +-- low-percentile-g?    yang:gauge64
               |  +-- mid-percentile-g?    yang:gauge64
               |  +-- high-percentile-g?   yang:gauge64
               |  +-- peak-g?              yang:gauge64
               |  +-- current-g?           yang:gauge64
               +-- top-talker
                  +-- talker* [source-prefix]
                     +-- spoofed-status?            boolean
                     +-- source-prefix              inet:ip-prefix
                     +-- source-port-range* [lower-port]
                     |  +-- lower-port    inet:port-number
                     |  +-- upper-port?   inet:port-number
                     +-- source-icmp-type-range* [lower-type]
                     |  +-- lower-type    uint8
                     |  +-- upper-type?   uint8
                     +-- total-attack-traffic* [unit]
                     |  +-- unit                 unit
                     |  +-- low-percentile-g?    yang:gauge64
                     |  +-- mid-percentile-g?    yang:gauge64
```

```
|  +-- high-percentile-g?   yang:gauge64
|  +-- peak-g?              yang:gauge64
|  +-- current-g?           yang:gauge64
+-- total-attack-connection-protocol*
       [protocol]
   +-- protocol             uint8
   +-- connection-c
   |  +-- low-percentile-g?    yang:gauge64
   |  +-- mid-percentile-g?    yang:gauge64
   |  +-- high-percentile-g?   yang:gauge64
   |  +-- peak-g?              yang:gauge64
   |  +-- current-g?           yang:gauge64
   +-- embryonic-c
   |  +-- low-percentile-g?    yang:gauge64
   |  +-- mid-percentile-g?    yang:gauge64
   |  +-- high-percentile-g?   yang:gauge64
   |  +-- peak-g?              yang:gauge64
   |  +-- current-g?           yang:gauge64
   +-- connection-ps-c
   |  +-- low-percentile-g?    yang:gauge64
   |  +-- mid-percentile-g?    yang:gauge64
   |  +-- high-percentile-g?   yang:gauge64
   |  +-- peak-g?              yang:gauge64
   |  +-- current-g?           yang:gauge64
   +-- request-ps-c
   |  +-- low-percentile-g?    yang:gauge64
   |  +-- mid-percentile-g?    yang:gauge64
   |  +-- high-percentile-g?   yang:gauge64
   |  +-- peak-g?              yang:gauge64
   |  +-- current-g?           yang:gauge64
   +-- partial-request-c
      +-- low-percentile-g?    yang:gauge64
      +-- mid-percentile-g?    yang:gauge64
      +-- high-percentile-g?   yang:gauge64
      +-- peak-g?              yang:gauge64
      +-- current-g?           yang:gauge64
```

Рисунок 29. Структура дерева деталей атаки.

Для оптимизации размера сообщений телеметрии в сигнальном канале DOTS агенты DOTS могут применять канал данных DOTS [RFC8783] при обмене зависящими от производителей деталями атак (т. е. {vendor identifier, attack identifier} ==> текстовое описание атаки). Таким образом, агенты DOTS не передают систематически описаний атаки в сообщениях телеметрии по сигнальному каналу DOTS.

8.1.6. Отображения атак

Можно использовать несколько отображений для разных идентификаторов производителей - агент DOTS, передающий данные телеметрии, может выбрать для использования одно или несколько отображений даже в одном сообщении.

> Примечание. Сервер DOTS может использовать несколько средств смягчения атак от разных производителей. Способы обмена телеметрическими данными и отображениями производителей между серверами и средствами смягчения DOTS выходят за рамки этого документа.

У клиентов и серверов DOTS могут быть отображения от разных производителей и свои наборы отображений атак. Агент DOTS **должен** воспринимать данные телеметрии с идентификаторами производителя, отличными от тех, которые он принимает для передачи данных телеметрии. Кроме того, клиент и сервер DOTS могут иметь средства от одного производителя, но с разными выпусками таблиц отображения. Клиенту DOTS **следует** передавать данные телеметрии с использованием любых отображений производителя, которые он предоставил серверу (например, в запросе POST, показанном на рисунке 30), а серверу DOTS **следует** использовать предоставленные клиенту DOTS отображения производителя при передаче данных телеметрии партнёрскому агенту DOTS.

```
POST /restconf/data/ietf-dots-data-channel:dots-data\
     /dots-client=dz6pHjaADkaFTbjr0JGBpw HTTP/1.1
Host: example.com
Content-Type: application/yang-data+json

{
  "ietf-dots-mapping:vendor-mapping": {
    "vendor": [
      {
        "vendor-id": 345,
        "vendor-name": "mitigator-c",
        "last-updated": "1629898958",
        "attack-mapping": [
          {
            "attack-id": 1,
            "attack-description":
               "Описание атаки"
          },
          {
            "attack-id": 2,
            "attack-description":
               "Описание атаки"
          }
        ]
      }
    ]
  }
}
```

Рисунок 30. POST для установки деталей Vendor Attack Mapping.

Модуль YANG ietf-dots-mapping, определённый в параграфе 11.2, дополняет модуль ietf-dots-data-channel [RFC8783]. Структура ietf-dots-mapping показана на рисунке 31.

```
module: ietf-dots-mapping
  augment /data-channel:dots-data/data-channel:dots-client:
    +--rw vendor-mapping {dots-telemetry}?
       +--rw vendor* [vendor-id]
          +--rw vendor-id         uint32
          +--rw vendor-name?      string
          +--rw description-lang?   string
          +--rw last-updated      uint64
          +--rw attack-mapping* [attack-id]
             +--rw attack-id             uint32
             +--rw attack-description    string
  augment /data-channel:dots-data/data-channel:capabilities:
    +--ro vendor-mapping-enabled?   boolean {dots-telemetry}?
  augment /data-channel:dots-data:
    +--ro vendor-mapping {dots-telemetry}?
       +--ro vendor* [vendor-id]
          +--ro vendor-id         uint32
          +--ro vendor-name?      string
          +--ro description-lang?   string
          +--ro last-updated      uint64
          +--ro attack-mapping* [attack-id]
             +--ro attack-id             uint32
             +--ro attack-description    string
```

Рисунок 31. Структура дерева Vendor Attack Mapping.

Клиент DOTS передаёт запрос GET по каналу данных DOTS для получения списка поддерживаемых сервером DOTS возможностей, как указано в параграфе 7.1 [RFC8783]. Этот запрос позволяет проверить, поддерживает ли сервер совместное использование деталей отображения атак от производителя (проверка vendor-mapping-enabled).

Если vendor-mapping-enabled имеет значение true, клиент DOTS **может** передавать запрос GET для получения от сервера деталей отображения атак. Пример такого запроса GET представлен на рисунке 32.

```
GET /restconf/data/ietf-dots-data-channel:dots-data\
    /ietf-dots-mapping:vendor-mapping HTTP/1.1
Host: example.com
Accept: application/yang-data+json
```

Рисунок 32. GET для получения Vendor Attack Mapping от сервера DOTS.

Клиент DOTS может извлечь лишь список производителей, поддерживаемых сервером DOTS, устанавливая для параметра depth (параграф 4.8.2 в [RFC8040]) значение 3 в запросе GET, как показано на рисунке 33. Пример тела отклика сервера DOTS на такой запрос приведён на рисунке 34.

```
GET /restconf/data/ietf-dots-data-channel:dots-data\
    /ietf-dots-mapping:vendor-mapping?depth=3 HTTP/1.1
Host: example.com
Accept: application/yang-data+json
```

Рисунок 33. GET для извлечения Vendors List, используемого сервером DOTS.

```
{
   "ietf-dots-mapping:vendor-mapping": {
     "vendor": [
       {
         "vendor-id": 32473,
         "vendor-name": "mitigator-s",
         "last-updated": "1629898758",
         "attack-mapping": []
       }
     ]
   }
 }
```

Рисунок 34. Тело отклика на запрос GET для Vendors List.

Клиент DOTS регулярно (например, каждую неделю) повторяет процедуру обновления отображений атак производителями с сервера DOTS.

Если клиент DOTS видит, что у сервера DOTS нет ссылок на детали конкретного отображения атак, он использует запрос POST для установки своего отображения атак от производителя. Пример такого запроса показан на рисунке 30.

Сервер DOTS указывает результат обработки запроса POST в строке состояния. Если сервер воспринимает предложенное в запросе отображение, он **должен** возвращать строку «201 Created». Если в запросе нет обязательного атрибута или имеется недействительный или неизвестный параметр, сервер **должен** возвращать в отклике строку «400 Bad Request», устанавливая тег ошибки missing-attribute, invalid-value или unknown-element.

Если запрос получен через шлюз серверного домена DOTS, а сервер DOTS не поддерживает указанное значение cdid тогда как cuid ожидается, сервер **должен** вернуть строку «403 Forbidden» с тегом ошибки access-denied. При получении такого отклика клиент DOTS **должен** зарегистрироваться (параграф 5.1 в [RFC8783]).

Клиент DOTS использует запрос PUT для обновления деталей отображения атак от производителя на сервере DOTS (например, для добавления нового или обновления имеющегося отображения). Клиент DOTS использует запрос GET для получения дателей отображения атак от производителей на сревере (Рисунок 35).

При передаче деталей атак в сообщениях телеметрии DOTS (параграфы 8.2, 8.3 и раздел 9) агентам DOTS **недопустимо** включать атрибут attack-description, если соответствующие детали отображения атак ранее не были обобщены с партнерским агентом DOTS.

```
GET /restconf/data/ietf-dots-data-channel:dots-data\
    /dots-client=dz6pHjaADkaFTbjr0JGBpw\
    /ietf-dots-mapping:vendor-mapping?\
    content=all HTTP/1.1
Host: example.com
Accept: application/yang-data+json
```

Рисунок 35. GET для извлечения Installed Vendor Attack Mapping Details.

8.2. От клиентов к серверам DOTS

Клиенты DOTS применяют запрос PUT (Рисунок 36) для передачи телеметрии pre-or-ongoing-mitigation серверам DOTS.

```
Header: PUT (Code=0.03)
Uri-Path: ".well-known"
Uri-Path: "dots"
Uri-Path: "tm"
Uri-Path: "cuid=dz6pHjaADkaFTbjr0JGBpw"
Uri-Path: "tmid=123"
Content-Format: "application/dots+cbor"

{
  "ietf-dots-telemetry:telemetry": {
    "pre-or-ongoing-mitigation": [
      {
        "target": {
          "target-prefix": [
            "2001:db8::1/128"
          ]
        },
        "total-attack-traffic-protocol": [
          {
            "protocol": 17,
            "unit": "megabit-ps",
            "mid-percentile-g": "900"
          }
        ],
        "attack-detail": [
          {
            "vendor-id": 32473,
            "attack-id": 77,
            "start-time": "1608336568",
            "attack-severity": "high"
          }
        ]
      }
    ]
  }
}
```

Рисунок 36. PUT для отправки телеметрии до и во время смягчения атак.

Параметр cuid является обязательным в Uri-Path запросов PUT.

Ниже приведено определение дополнительного параметра Uri-Path.

tmid

Идентификатор телеметрии (Telemetry Identifier) служит для представления данных телеметрии DOTS до и во время смягчения атак целым числом и **должен** создаваться клиентами DOTS. Значения tmid **должны** монотонно возрастать по мере необходимости передачи клиентом новых данных телеметрии pre-or-ongoing-mitigation.

При достижении максимального значения (rollover) параметра tmid **должна** применяться процедура, заданная в параграфе 4.4.1 [RFC9132] для параметра mid.

Этот атрибут является обязательным и **должен** размещаться в опциях Uri-Path после атрибута cuid.

Атрибуты cuid и tmid **недопустимо** включать в тело запросов PUT.

В запросе PUT **должен** присутствовать хотя бы один атрибут target и иной атрибут pre-or-ongoing-mitigation (параграф 8.1). При наличии лишь атрибута target, запрос обрабатывается в соответствии с параграфом 8.3.

Относительный порядов пары запросов PUT с телеметрией pre-or-ongoing-mitigation от клиента DOTS определяется сравнением значений tmid. Если в двух запросах перекрываются атрибуты target, преимущество имеет запрос с большим tmid, а запрос с меньшим tmid **должен** автоматически удаляться, теряя доступность.

Сервер DOTS указывает результат обработки запроса PUT кодом отклика CoAP. В частности, возвращается код 2.04 (Changed), если сервер воспринял телеметрию pre-or-ongoing-mitigation. При возникновении ошибки на сервере возвращается код 5.03 (Service Unavailable) с использованием опции Max-Age для указания числа секунд, по истечении которых можно повторить запрос.

Продолжительность поддержки сервером DOTS активности tmid или регистрации вложенных данных телеметрии зависит от реализации. Если идентификатор tmid остаётся активным, данные журнала обновляются сервером DOTS в зависимости от полученных от клиента-партнера сведений.

```
Header: DELETE (Code=0.04)
Uri-Path: ".well-known"
Uri-Path: "dots"
Uri-Path: "tm"
Uri-Path: "cuid=dz6pHjaADkaFTbjr0JGBpw"
Uri-Path: "tmid=123"
```

Рисунок 37. Удаление конкретных данных телеметрии Pre-or-Ongoing-Mitigation.

Клиент DOTS, который потерял статус своих активных tmid или должен сбросить tmid в 0 (например, при аварии или перезапуске) должен передать запрос GET серверам DOTS для получения списка активных tmid. Затем клиент DOTS может удалить tmid, которым больше не нужно быть активными (Рисунок 37). Передача запроса DELETE без tmid указывает, что нужно деактивировать все tmid (Рисунок 38).

```
Header: DELETE (Code=0.04)
Uri-Path: ".well-known"
Uri-Path: "dots"
Uri-Path: "tm"
Uri-Path: "cuid=dz6pHjaADkaFTbjr0JGBpw"
```

Рисунок 38. Удаление всех данных телеметрии Pre-or-Ongoing-Mitigation.

8.3. От серверов DOTS к клиентам

Данные pre-or-ongoing-mitigation (в частности, детали атаки) могут передаваться от серверов DOTS клиентам. Например, сервер DOTS, размещённый вместе с детектором DDoS, может вести мониторинг целевых сетей и обнаруживать DDoS-атаки путём статистического анализа или глубокого изучения, сообщая детали атак клиентам DOTS. Клиент может использовать эти сведения для решения вопроса о потребности в смягчении атак. Кроме того, персонал служб безопасности клиентских доменов DOTS может применять детали атак для определения стратегии защиты и выбора подходящего сервера DOTS для ослабления атаки.

Для получения уведомлений телеметрии до и во время смягчения от сервера DOTS клиент DOTS **должен** передать запрос PUT (затем GET) с фильтром цели. Пример такого запроса PUT показан на рисунке 39. Чтобы не поддерживать длинный список таких запросов, клиентам DOTS **рекомендуется** включать в один запрос все цели (в предположении, что эти сведения помещаются в одну дейтаграмму). Серверам DOTS можно задать ограничение числа запросов pre-or-ongoing-mitigation на клиентский домен DOTS. Запросы pre-or-ongoing-mitigation **должны** поддерживаться в активном состоянии сервером DOTS до получения от того же клиента DOTS запроса DELETE, отменяющего эту телеметрию pre-or-ongoing-mitigation, или пока клиент DOTS не будет сочтён неактивным (см., например, параграф 3.5 в [RFC8783]).

Относительный порядов пары запросов PUT для телеметрии pre-or-ongoing-mitigation от клиента DOTS определяется сравнением значений tmid. Если в двух запросах перекрываются атрибуты target, преимущество имеет запрос с большим tmid, а запрос с меньшим tmid **должен** автоматически удаляться, теряя доступность.

```
Header: PUT (Code=0.03)
Uri-Path: ".well-known"
Uri-Path: "dots"
Uri-Path: "tm"
Uri-Path: "cuid=dz6pHjaADkaFTbjr0JGBpw"
Uri-Path: "tmid=567"
Content-Format: "application/dots+cbor"

{
  "ietf-dots-telemetry:telemetry": {
    "pre-or-ongoing-mitigation": [
      {
        "target": {
          "target-prefix": [
            "2001:db8::/32"
          ]
        }
      }
    ]
  }
}
```

Рисунок 39. PUT для запроса телеметрии Pre-or-Ongoing-Mitigation.

Клиенты DOTS из одного домена могут запросить получение телеметрии pre-or-ongoing-mitigation, привязанной к одной цели, без возникновения «перекрытий» и конфликтов.

```
Header: GET (Code=0.01)
Uri-Path: ".well-known"
Uri-Path: "dots"
Uri-Path: "tm"
Uri-Path: "cuid=dz6pHjaADkaFTbjr0JGBpw"
Uri-Path: "tmid=567"
Observe: 0
```

Рисунок 40. GET для подписки на асинхронные уведомления телеметрии для tmid.

После успешного выполнения запроса PUT для создания статуса запроса на сервере клиент DOTS передаёт запрос GET для получения обновлений телеметрии. Клиент использует опцию Observe = 0 (регистрация) в запросе GET для получения асинхронных уведомлений с данными телеметрии до и во время смягчения атак от сервера DOTS. Запрос GET может указывать конкретное значение tmid (Рисунок 40) или не включать tmid (Рисунок 41) для получения обновления по всем запросам от этого клиента.

```
Header: GET (Code=0.01)
Uri-Path: ".well-known"
Uri-Path: "dots"
Uri-Path: "tm"
Uri-Path: "cuid=dz6pHjaADkaFTbjr0JGBpw"
Observe: 0
```

Рисунок 41. GET для подписки на асинхронные уведомления телеметрии для всех tmid.

Клиент DOTS может задать фильтр для запроса части асинхронных уведомлений от сервера DOTS, включив одну или несколько опций Uri-Query в свой запрос GET. Опция Uri-Query может включать параметры target-prefix, target-port, target-protocol, target-fqdn, target-uri, alias-name, mid и c (content - содержимое) (параграф 5.4).

- При наличии в запросе нескольких опций Uri-Query они интерпретируются так же, как при включении нескольких атрибутов target в тело сообщения (параграф 4.4.1 в [RFC9132]).

- Если в запрос включается несколько значений параметра, эти значения должны размещаться в одной опции Uri-Query через запятую без пробелов.
- Диапазоны значений (непрерывный блок с включением границ) можно применять в параметрах target-port, target-protocol и mid, указывая границы через дефис (-).
- Шаблоны имён (левая часть имени заменена символом *) можно включать в параметры target-fqdn и target-uri. Клиентам DOTS **недопустимо** указывать шаблоны, где символ * не является первым. Примером корректного шаблона служит *.example.com и такие шаблоны можно включать в параметр target-fqdn опции Uri-Query.

Клиенты DOTS могут также филтровать асинхронные уведомления от сервера DOTS, указывая информацию о конкретном источнике атаки с помощью атрибутов source-prefix, source-port, source-icmp-type в опции Uri-Query. Здесь применяется такой же подход (диапазоны, множество значений) как для атрибутов цели. При использовании этих фильтров **следует** соблюдать особую осторожность, поскольку фильтры могут скрывать некоторые атаки от запрашивающего клиента DOTS (например, при смене информации об источнике атаки).

Запросы с недействительными (например, не поддерживаемыми или некорректно сформированными) типами сервер DOTS **должен** отвергать с кодом 4.00 (Bad Request).

Пример запроса подписки на асинхронные уведомления, относящиеся к трафику UDP, показан на рисунке 42. Этот фильтр применяется для всех tmid.

```
Header: GET (Code=0.01)
Uri-Path: ".well-known"
Uri-Path: "dots"
Uri-Path: "tm"
Uri-Path: "cuid=dz6pHjaADkaFTbjr0JGBpw"
Uri-Query: "target-protocol=17"
Observe: 0
```

Рисунок 42. GET для получения асинхронных уведомлений с фильтром Uri-Query.

Сервер DOTS будет передавать асинхронные уведомления клиентам DOTS при обнаружении связанных с атакой событий, следуя процедурам, подобным описанным в параграфе 4.4.2.1 [RFC9132]. Пример асинхронного уведомления телеметрии pre-or-ongoing-mitigation показан на рисунке 43.

```
{
  "ietf-dots-telemetry:telemetry": {
    "pre-or-ongoing-mitigation": [
      {
        "tmid": 567,
        "target": {
          "target-prefix": [
            "2001:db8::1/128"
          ]
        },
        "target-protocol": [
          17
        ],
        "total-attack-traffic": [
          {
            "unit": "megabit-ps",
            "mid-percentile-g": "900"
          }
        ],
        "attack-detail": [
          {
            "vendor-id": 32473,
            "attack-id": 77,
            "start-time": "1618339785",
            "attack-severity": "high"
          }
        ]
      }
    ]
  }
}
```

Рисунок 43. Тело уведомления телеметрии Pre-or-Ongoing-Mitigation от сервера DOTS.

Сервер DOTS передаёт совокупные данные для цели, используя атрибут total-attack-traffic. Агрегирование предполагает применение для цели фильтров Uri-Query. Сервер DOTS при необходимости может включить более подробные сведения (т. е. total-attack-traffic-protocol и total-attack-traffic-port). Если в запрос включён фильтр по протоколу (или порту), применяется total-attack-traffic-protocol (или total-attack-traffic-port).

Сервер DOTS может агрегировать данные pre-or-ongoing-mitigation (например, top-talker) для всех целей домена или (когда это задано) передавать конкретные сведения (например, top-talker) для указанной цели.

Клиент DOTS может регистрировать данные телеметрии pre-or-ongoing-mitigation с передачей сигналов администратору или сетевому контроллеру. Клиент DOTS может передавать запросы на смягчение, если атаку не удаётся обработать локально. Клиент DOTS, не заинтересованный в телеметрии pre-or-ongoing-mitigation, передаёт запрос DELETE, подобно показанному на рисунке 37 запросу DELETE.

9. Обновление статуса телеметрии смягчения DOTS

9.1. От клиентов к серверам DOTS - телеметрия эффективности смягчения

Атрибуты телеметрии эффективности смягчения атак могут передаваться серверам от клиентов DOTS как часть периодических обновления эффективности смягчения, отправляемых серверу (параграф 4.4.3 в [RFC9132]).

Total attack traffic - суммарный трафик атаки

Суммарный трафик с точки зрения клиента DOTS в процессе смягчения атаки (см. Рисунок 27).

Attack details - детали атаки

Детали атаки с точки зрения клиента DOTS в процессе смягчения атаки (см. параграф 8.1.5).

Модуль YANG ietf-dots-telemetry (параграф 11.1) дополняет тип mitigation-scope из модуля ietf-dots-signal-channel [RFC9132], чтобы клиент DOTS мог передавать эти атрибуты в обновлениях эффективности смягчения (Рисунок 44).

```
augment-structure /dots-signal:dots-signal/dots-signal:message-type
                  /dots-signal:mitigation-scope/dots-signal:scope:
  +-- total-attack-traffic* [unit]
  |  +-- unit                 unit
  |  +-- low-percentile-g?    yang:gauge64
  |  +-- mid-percentile-g?    yang:gauge64
  |  +-- high-percentile-g?   yang:gauge64
  |  +-- peak-g?              yang:gauge64
  |  +-- current-g?           yang:gauge64
  +-- attack-detail* [vendor-id attack-id]
     +-- vendor-id             uint32
     +-- attack-id             uint32
     +-- attack-description?   string
     +-- attack-severity?      attack-severity
     +-- start-time?           uint64
     +-- end-time?             uint64
     +-- source-count
     |  +-- low-percentile-g?    yang:gauge64
     |  +-- mid-percentile-g?    yang:gauge64
     |  +-- high-percentile-g?   yang:gauge64
     |  +-- peak-g?              yang:gauge64
     |  +-- current-g?           yang:gauge64
     +-- top-talker
        +-- talker* [source-prefix]
           +-- spoofed-status?            boolean
           +-- source-prefix              inet:ip-prefix
           +-- source-port-range* [lower-port]
           |  +-- lower-port    inet:port-number
           |  +-- upper-port?   inet:port-number
           +-- source-icmp-type-range* [lower-type]
           |  +-- lower-type    uint8
           |  +-- upper-type?   uint8
           +-- total-attack-traffic* [unit]
           |  +-- unit                 unit
           |  +-- low-percentile-g?    yang:gauge64
           |  +-- mid-percentile-g?    yang:gauge64
           |  +-- high-percentile-g?   yang:gauge64
           |  +-- peak-g?              yang:gauge64
           |  +-- current-g?           yang:gauge64
           +-- total-attack-connection
              +-- connection-c
              |  +-- low-percentile-g?    yang:gauge64
              |  +-- mid-percentile-g?    yang:gauge64
              |  +-- high-percentile-g?   yang:gauge64
              |  +-- peak-g?              yang:gauge64
              |  +-- current-g?           yang:gauge64
              +-- embryonic-c
              |  ...
              +-- connection-ps-c
              |  ...
              +-- request-ps-c
              |  ...
              +-- partial-request-c
                 ...
```

Рисунок 44. Структура дерева обновления эффективности телеметрии.

Для передачи данных телеметрии в обновлениях эффективности смягчения атаки **рекомендуется**, чтобы клиент DOTS уже организовал сессию установки телеметрии DOTS с сервером в период «бездействия». Эта сессия предназначена в первую очередь для оценки поддержки партнёром DOTS расширений телеметрии и, таким образом, предотвращения отказов при обработке сообщений (параграф 3.1 в [RFC9132]).

```
Header: PUT (Code=0.03)
Uri-Path: ".well-known"
Uri-Path: "dots"
Uri-Path: "mitigate"
Uri-Path: "cuid=dz6pHjaADkaFTbjr0JGBpw"
Uri-Path: "mid=123"
If-Match:
Content-Format: "application/dots+cbor"

{
  "ietf-dots-signal-channel:mitigation-scope": {
    "scope": [
      {
        "alias-name": [
          "https1",
          "https2"
        ],
        "attack-status": "under-attack",
        "ietf-dots-telemetry:total-attack-traffic": [
          {
            "unit": "megabit-ps",
            "mid-percentile-g": "900"
          }
        ]
      }
    ]
  }
}
```

Рисунок 45. Пример обновления эффективности телеметрии с атрибутами телеметрии.

Пример обновления эффективности смягчения с атрибутами телеметрии показан на рисунке 45.

9.2. От серверов к клиентам - атрибуты телеметрии статуса смягчения DOTS

Атрибуты телеметрии статуса смягчения атак могут передаваться от сервера DOTS клиентам как часть периодического обновления статуса смягчения (параграф 4.4.2 в [RFC9132]). В частности, клиенты DOTS могут получать асинхронные уведомления о деталях атаки от сервера с использованием опции Observe, определённой в [RFC7641]. Для использования этой возможности клиенты DOTS **должны** организовать сеанс телеметрии с сервером в состоянии «бездействия» и **должны** установить для атрибута server-originated-telemetry значение true. Серверам DOTS **недопустимо** включать атрибуты в обновления статуса смягчения для клиентов DOTS в сессиях телеметрии, где для server-originated-telemetry задано значение false.

Как задано в [RFC8612], фактические действия по смягчению атак могут включать несколько мер противодействия. Сервер DOTS передаёт текущий статус соответствующих мер и **может** включать список обнаруженных атак. Атрибуты, заданные в параграфе 8.1.5, применимы к описанию атак, обнаруженных и смягчаемых в домене сервера DOTS.

Модуль YANG ietf-dots-telemetry (параграф 11.1) дополняет тип сообщений mitigation-scope из модуля ietf-dots-signal-channel [RFC9132] данными телеметрии, как показано на рисунке 46.

```
augment-structure /dots-signal:dots-signal/dots-signal:message-type
                  /dots-signal:mitigation-scope/dots-signal:scope:
  +-- (direction)?
  |  +--:(server-to-client-only)
  |     +-- total-traffic* [unit]
  |     |  +-- unit                 unit
  |     |  +-- low-percentile-g?    yang:gauge64
  |     |  +-- mid-percentile-g?    yang:gauge64
  |     |  +-- high-percentile-g?   yang:gauge64
  |     |  +-- peak-g?              yang:gauge64
  |     |  +-- current-g?           yang:gauge64
  |     +-- total-attack-connection
  |        +-- connection-c
  |        |  +-- low-percentile-g?    yang:gauge64
  |        |  +-- mid-percentile-g?    yang:gauge64
  |        |  +-- high-percentile-g?   yang:gauge64
  |        |  +-- peak-g?              yang:gauge64
  |        |  +-- current-g?           yang:gauge64
  |        +-- embryonic-c
  |        |  ...
  |        +-- connection-ps-c
  |        |  ...
  |        +-- request-ps-c
  |        |  ...
  |        +-- partial-request-c
  |           ...
  +-- total-attack-traffic* [unit]
  |  +-- unit                 unit
  |  +-- low-percentile-g?    yang:gauge64
  |  +-- mid-percentile-g?    yang:gauge64
  |  +-- high-percentile-g?   yang:gauge64
  |  +-- peak-g?              yang:gauge64
  |  +-- current-g?           yang:gauge64
  +-- attack-detail* [vendor-id attack-id]
     +-- vendor-id             uint32
     +-- attack-id             uint32
     +-- attack-description?   string
     +-- attack-severity?      attack-severity
     +-- start-time?           uint64
     +-- end-time?             uint64
     +-- source-count
     |  +-- low-percentile-g?    yang:gauge64
     |  +-- mid-percentile-g?    yang:gauge64
     |  +-- high-percentile-g?   yang:gauge64
     |  +-- peak-g?              yang:gauge64
     |  +-- current-g?           yang:gauge64
     +-- top-talker
        +-- talker* [source-prefix]
           +-- spoofed-status?            boolean
           +-- source-prefix              inet:ip-prefix
           +-- source-port-range* [lower-port]
           |  +-- lower-port    inet:port-number
           |  +-- upper-port?   inet:port-number
           +-- source-icmp-type-range* [lower-type]
           |  +-- lower-type    uint8
           |  +-- upper-type?   uint8
           +-- total-attack-traffic* [unit]
           |  +-- unit                 unit
           |  +-- low-percentile-g?    yang:gauge64
           |  +-- mid-percentile-g?    yang:gauge64
           |  +-- high-percentile-g?   yang:gauge64
           |  +-- peak-g?              yang:gauge64
           |  +-- current-g?           yang:gauge64
           +-- total-attack-connection
              +-- connection-c
              |  +-- low-percentile-g?    yang:gauge64
              |  +-- mid-percentile-g?    yang:gauge64
              |  +-- high-percentile-g?   yang:gauge64
              |  +-- peak-g?              yang:gauge64
              |  +-- current-g?           yang:gauge64
              +-- embryonic-c
              |  ...
              +-- connection-ps-c
              |  ...
              +-- request-ps-c
              |  ...
              +-- partial-request-c
                 ...
```

Рисунок 46. Структура дерева телеметрии статуса смягчения от сервера к клиенту.

На рисунке 47 приведён пример асинхронного уведомления о статусе смягчения атак от сервера DOTS. Это уведомления содержит значение mid-percentile трафика обрабатываемой атаки и пиковые значения уникальных участников атаки.

```
{
  "ietf-dots-signal-channel:mitigation-scope": {
    "scope": [
      {
        "mid": 12332,
        "mitigation-start": "1507818434",
        "alias-name": [
          "https1",
          "https2"
        ],
        "lifetime": 1600,
        "status": "attack-successfully-mitigated",
        "bytes-dropped": "134334555",
        "bps-dropped": "43344",
        "pkts-dropped": "333334444",
        "pps-dropped": "432432",
        "ietf-dots-telemetry:total-attack-traffic": [
          {
            "unit": "megabit-ps",
            "mid-percentile-g": "752"
          }
        ],
        "ietf-dots-telemetry:attack-detail": [
          {
            "vendor-id": 32473,
            "attack-id": 77,
            "source-count": {
              "peak-g": "12683"
            }
          }
        ]
      }
    ]
  }
}
```

Рисунок 47. Тело отклика со статусом смягчения и атрибутами телеметрии.

Клиенты DOTS могут фильтровать асинхронные уведомления от сервера DOTS, указывая опцию Uri-Query в запросе GET. Опция Uri-Query может включать параметры target-prefix, target-port, target-protocol, target-fqdn, target-uri, alias-name, с (content) (параграф 5.4). **Должны** применяться приведённые в параграфе 8.3 соображения в части включения нескольких значений, диапазонов (target-port, target-protocol) и шаблонов (target-fqdn, target-uri). Пример запроса подписки на асинхронные уведомления для псевдонима https1 приведён на рисунке 48.

```
Header: GET (Code=0.01)
Uri-Path: ".well-known"
Uri-Path: "dots"
Uri-Path: "mitigate"
Uri-Path: "cuid=dz6pHjaADkaFTbjr0JGBpw"
Uri-Path: "mid=12332"
Uri-Query: "target-alias=https1"
Observe: 0
```

Рисунок 48. GET для получения асинхронных уведомлений с фильтром Uri-Query.

Если запрос не соответствует цели встроенного mid на сервере DOTS, сервер **должен** возвращать отклик 4.04 (Not Found). Серверу DOTS **недопустимо** добавлять новую запись Observe, если запрос перекрывается с имеющейся записью Observe. В таком случае сервер должен возвращать отклик 4.09 (Conflict).

10. Обработка ошибок

Список ошибок CoAP, поддерживаемых серверами DOTS, представлен в разделе 9 [RFC9132]. Ниже указаны ошибки, относящиеся к расширению для телеметрии.

- 4.00 (Bad Request) сервер DOTS возвращает при получении от клиента запросов, нарушающих расширение телеметрии DOTS.
- 4.04 (Not Found) сервер DOTS возвращает при получении от клиента запросов с недействительным tsid или tmid.
- 4.00 (Bad Request) сервер DOTS возвращает при получении от клиента недействительных запросов (например, не поддерживаемых или некорректно сформированных).
- 4.04 (Not Found) сервер DOTS возвращает при получении от клиента запросов с целью, не соответствующей вложенному mid на сервере DOTS.

Как указано в разделе 9 [RFC9132], в теле отклика возвращается дополнительный текст (параграф 5.5.2 в [RFC7252]), помогающий в поиске неполадок.

11. Модули YANG

11.1. Модуль телеметрии сигнального канала DOTS

Этот модуль использует типы, определённые в [RFC6991] и [RFC8345], а также группировки из [RFC8783].

```
<CODE BEGINS> file "ietf-dots-telemetry@2022-06-20.yang"
module ietf-dots-telemetry {
  yang-version 1.1;
  namespace "urn:ietf:params:xml:ns:yang:ietf-dots-telemetry";
  prefix dots-telemetry;

  import ietf-dots-signal-channel {
    prefix dots-signal;
    reference
      "RFC 9132: Distributed Denial-of-Service Open Threat
                 Signaling (DOTS) Signal Channel Specification";
  }
  import ietf-dots-data-channel {
    prefix data-channel;
    reference
      "RFC 8783: Distributed Denial-of-Service Open Threat
                 Signaling (DOTS) Data Channel Specification";
  }
  import ietf-yang-types {
    prefix yang;
    reference
      "RFC 6991: Common YANG Data Types, Section 3";
  }
  import ietf-inet-types {
    prefix inet;
    reference
      "RFC 6991: Common YANG Data Types, Section 4";
  }
  import ietf-network-topology {
    prefix nt;
    reference
      "RFC 8345: A YANG Data Model for Network Topologies,
                 Section 6.2";
  }
  import ietf-yang-structure-ext {
    prefix sx;
    reference
      "RFC 8791: YANG Data Structure Extensions";
  }

  organization
    "IETF DDoS Open Threat Signaling (DOTS) Working Group";
  contact
    "WG Web:   <https://datatracker.ietf.org/wg/dots/>
     WG List:  <mailto:dots@ietf.org>

     Author:   Mohamed Boucadair
               <mailto:mohamed.boucadair@orange.com>

     Author:   Konda, Tirumaleswar Reddy.K
               <mailto:kondtir@gmail.com>";
  description
    "Этот модуль содержит определения YANG для сигналов телеметрии
     DOTS, передаваемых между клиентами и серверами DOTS по
     сигнальному каналу DOTS.

     Авторские права (Copyright (c) 2022) принадлежат IETF Trust и
     лицам, указанным как авторы. Все права защищены.

     Распространение и применение модуля в исходной или двоичной
     форме с изменениями или без таковых разрешено в соответствии с
     лицензией Simplified BSD License, изложенной в параграфе 4.c
     IETF Trust's Legal Provisions Relating to IETF Documents
     (https://trustee.ietf.org/license-info).

     Эта версия модуля YANG является частью RFC 9244, где правовые
     аспекты приведены более полно.";

  revision 2022-06-20 {
    description
      "Исходный выпуск.";
    reference
      "RFC 9244: Distributed Denial-of-Service Open Threat
                 Signaling (DOTS) Telemetry";
  }

  typedef attack-severity {
    type enumeration {
      enum none {
        value 1;
        description
          "Без влияния на клиентский домен DOTS.";
      }
      enum low {
        value 2;
        description
```

```
            "Минимальное влияние на клиентский домен DOTS.";
        }
        enum medium {
          value 3;
          description
            "Часть ресурсов клиентского домена DOTS не обслуживается.";
        }
        enum high {
          value 4;
          description
            "Клиентский домен DOTS находится в крайне тяжёлых условиях.";
        }
        enum unknown {
          value 5;
          description
            "Влияние атаки не известно.";
        }
      }
      description
        "Перечисляемые значения для уровня атак.";
      reference
        "RFC 7970: The Incident Object Description Exchange
                   Format Version 2, Section 3.12.2";
    }

    typedef unit-class {
      type enumeration {
        enum packet-ps {
          value 1;
          description
            "Пакетов в секунду (pps).";
        }
        enum bit-ps {
          value 2;
          description
            "Бит в секунду (bps).";
        }
        enum byte-ps {
          value 3;
          description
            "Байт в секунду (Bps).";
        }
      }
      description
        "Перечисляемые значения для классов единиц измерения.
         Поддерживаются классы pps, bps, Bps.";
    }

    typedef unit {
      type enumeration {
        enum packet-ps {
          value 1;
          description
            "Пакетов в секунду (pps).";
        }
        enum bit-ps {
          value 2;
          description
            "Бит в секунду (bps).";
        }
        enum byte-ps {
          value 3;
          description
            "Байт в секунду (Bps).";
        }
        enum kilopacket-ps {
          value 4;
          description
            "Килопакетов в секунду (kpps).";
        }
        enum kilobit-ps {
          value 5;
          description
            "Килобит в секунду (kbps).";
        }
        enum kilobyte-ps {
          value 6;
          description
            "Килобайт в секунду (kBps).";
        }
        enum megapacket-ps {
          value 7;
          description
            "Мегапакетов в секунду (Mpps).";
        }
        enum megabit-ps {
```

```
          value 8;
          description
            "Мегабит в секунду (Mbps).";
        }
        enum megabyte-ps {
          value 9;
          description
            "Мегабайт в секунду (MBps).";
        }
        enum gigapacket-ps {
          value 10;
          description
            "Гигапакетов в секунду (Gpps).";
        }
        enum gigabit-ps {
          value 11;
          description
            "Гигабит в секунду (Gbps).";
        }
        enum gigabyte-ps {
          value 12;
          description
            "Гигабайт в секунду (GBps).";
        }
        enum terapacket-ps {
          value 13;
          description
            "Терапакетов в секунду (Tpps).";
        }
        enum terabit-ps {
          value 14;
          description
            "Терабит в секунду (Tbps).";
        }
        enum terabyte-ps {
          value 15;
          description
            "Терабайт в секунду (TBps).";
        }
        enum petapacket-ps {
          value 16;
          description
            "Петапакетов в секунду (Ppps).";
        }
        enum petabit-ps {
          value 17;
          description
            "Петабит в секунду (Pbps).";
        }
        enum petabyte-ps {
          value 18;
          description
            "Петабайт в секунду (PBps).";
        }
        enum exapacket-ps {
          value 19;
          description
            "Экзапакетов в секунду (Epps).";
        }
        enum exabit-ps {
          value 20;
          description
            "Экзабит в секунду (Ebps).";
        }
        enum exabyte-ps {
          value 21;
          description
            "Экзабайт в секунду (EBps).";
        }
        enum zettapacket-ps {
          value 22;
          description
            "Зеттапакетов в секунду (Zpps).";
        }
        enum zettabit-ps {
          value 23;
          description
            "зеттабит в секунду (Zbps).";
        }
        enum zettabyte-ps {
          value 24;
          description
            "Зеттабайт в секунду (ZBps).";
        }
      }
      description
```

```
        "Перечисляемые значения для единиц измерения. В связи с
         автоматическим масштабированием применяется лишь одна
         единица на класс.";
    }

    typedef interval {
      type enumeration {
        enum 5-minutes {
          value 1;
          description
            "5 минут.";
        }
        enum 10-minutes {
          value 2;
          description
            "10 минут.";
        }
        enum 30-minutes {
          value 3;
          description
            "30 минут.";
        }
        enum hour {
          value 4;
          description
            "Час.";
        }
        enum day {
          value 5;
          description
            "День.";
        }
        enum week {
          value 6;
          description
            "Неделя.";
        }
        enum month {
          value 7;
          description
            "Месяц.";
        }
      }
      description
        "Перечисляемые значения для продолжительности измерения.";
    }

    typedef sample {
      type enumeration {
        enum second {
          value 1;
          description
            "1-секундный интервал измерения.";
        }
        enum 5-seconds {
          value 2;
          description
            "5-секундный интервал измерения.";
        }
        enum 30-seconds {
          value 3;
          description
            "30-секундный интервал измерения.";
        }
        enum minute {
          value 4;
          description
            "1-минутный интервал измерения.";
        }
        enum 5-minutes {
          value 5;
          description
            "5-минутный интервал измерения.";
        }
        enum 10-minutes {
          value 6;
          description
            "10-минутный интервал измерения.";
        }
        enum 30-minutes {
          value 7;
          description
            "30-минутный интервал измерения.";
        }
        enum hour {
          value 8;
```

```
        description
          "Часовой интервал измерения.";
      }
    }
    description
      "Перечисляемые значения для интервала выборки.";
  }

  typedef percentile {
    type decimal64 {
      fraction-digits 2;
    }
    description
      "n-й процентиль из набора данных имеет значение при котором
       n процентов данных меньше его.";
  }

  typedef query-type {
    type enumeration {
      enum target-prefix {
        value 1;
        description
          "Запрос на основе префикса цели.";
      }
      enum target-port {
        value 2;
        description
          "Запрос на основе номера порта цели.";
      }
      enum target-protocol {
        value 3;
        description
          "Запрос на основе протокола цели.";
      }
      enum target-fqdn {
        value 4;
        description
          "Запрос на основе FQDN цели.";
      }
      enum target-uri {
        value 5;
        description
          "Запрос на основе URI цели.";
      }
      enum target-alias {
        value 6;
        description
          "Запрос на основе псевдонима цели.";
      }
      enum mid {
        value 7;
        description
          "Запрос на основе идентификатора смягчения (mid).";
      }
      enum source-prefix {
        value 8;
        description
          "Запрос на основе префикса источника.";
      }
      enum source-port {
        value 9;
        description
          "Запрос на основе номера порта.";
      }
      enum source-icmp-type {
        value 10;
        description
          "Запрос на основе типа ICMP.";
      }
      enum content {
        value 11;
        description
          "Запрос на основе опции Uri-Query с (content), применяемый для
           выбора конфигурационных и неконфигурационных узлов данных.";
        reference
          "RFC 9132: Distributed Denial-of-Service Open Threat
                     Signaling (DOTS) Signal Channel
                     Specification, Section 4.4.2";
      }
    }
    description
      "Перечисляемые значения для типов запроса, которые могут применться
       в запросах GET для фильтрации данных. Запрос с недействительным
       типом (например, не поддерживаемым или с неверным форматом) сервер
       DOTS отвергает с возвратом кода 4.00 (Bad Request).";
  }
```

```
  grouping telemetry-parameters {
    description
      "Группировка с набором параметров подготовки отчётов телеметрии.

       Группировка указывает интервалы измерения и выборки, а также
       значение low-percentile/mid-percentile/high-percentile.";
    leaf measurement-interval {
      type interval;
      description
        "Задаёт период, в течение которого рассчитываются процентили.";
    }
    leaf measurement-sample {
      type sample;
      description
        "Задаёт распределение времени измерения значений, применяемых
         для расчёта процентилей.

         Интервал выборки должен быть меньше длительности измерений.";
    }
    leaf low-percentile {
      type percentile;
      default "10.00";
      description
        "Low-percentile. Значение 0 отменяет low-percentile.";
    }
    leaf mid-percentile {
      type percentile;
      must '. >= ../low-percentile' {
        error-message
          "Значение mid-percentile должно быть не меньше low-percentile.";
      }
      default "50.00";
      description
        "Mid-percentile. Совпадение с low-percentile означает отказ от
         значений mid-percentile.";
    }
    leaf high-percentile {
      type percentile;
      must '. >= ../mid-percentile' {
        error-message
          "Должно быть не меньше mid-percentile.";
      }
      default "90.00";
      description
        "High-percentile. Совпадение с mid-percentile означает отказ от
         значений high-percentile.";
    }
  }

  grouping percentile-and-peak {
    description
      "Базовая группировка для значений процентилей и пиков.";
    leaf low-percentile-g {
      type yang:gauge64;
      description
        "Значение Low-percentile.";
    }
    leaf mid-percentile-g {
      type yang:gauge64;
      description
        "Значение Mid-percentile.";
    }
    leaf high-percentile-g {
      type yang:gauge64;
      description
        "Значение High-percentile.";
    }
    leaf peak-g {
      type yang:gauge64;
      description
        "Значение пика.";
    }
  }

  grouping percentile-peak-and-current {
    description
      "Базовая группировка для значений процентилей и пиков.";
    uses percentile-and-peak;
    leaf current-g {
      type yang:gauge64;
      description
        "Текущее значение.";
    }
  }
```

```
  grouping unit-config {
    description
      "Базовая группировка для настройки единиц измерения.";
    list unit-config {
      key "unit";
      description
        "Управляет калссами единиц, разрешенными для обмена данными.";
      leaf unit {
        type unit-class;
        description
          "Можно применять packet-ps, bit-ps или byte-ps.";
      }
      leaf unit-status {
        type boolean;
        mandatory true;
        description
          "Разрешает или запрещает класс единиц измерения.";
      }
    }
  }

  grouping traffic-unit {
    description
      "Группировка трафика как функции единицы измерения.";
    leaf unit {
      type unit;
      description
        "Трафик можно измерять с использованием классов packet-ps,
         bit-ps или byte-ps. Агенты DOTS автоматически приводят к
         соответствующим единицам (например, megabit-ps, kilobit-ps).";
    }
    uses percentile-and-peak;
  }

  grouping traffic-unit-all {
    description
      "Группировка трафика как функции единицы измерения,
       включая текущие значения.";
    uses traffic-unit;
    leaf current-g {
      type yang:gauge64;
      description
        "Текущее наблюдаемое значение.";
    }
  }

  grouping traffic-unit-protocol {
    description
      "Группировка трафика данного транспортного протокола
       как функции единицы измерения.";
    leaf protocol {
      type uint8;
      description
        "Транспортный протокол из реестра IANA Protocol Numbers
         <https://www.iana.org/assignments/protocol-numbers/>.

         Например, этот параметр содержит значение 6 для TCP,
         17 для UDP, 33 для DCCP, 132 для SCTP.";
    }
    uses traffic-unit;
  }

  grouping traffic-unit-protocol-all {
    description
      "Группировка трафика данного транспортного протокола как
       функции единицы измерения, включая текущие значения.";
    uses traffic-unit-protocol;
    leaf current-g {
      type yang:gauge64;
      description
        "Текущее наблюдаемое значение.";
    }
  }

  grouping traffic-unit-port {
    description
      "Группировка трафика данного номера порта как функции
       единицы измерения.";
    leaf port {
      type inet:port-number;
      description
        "Номер порта, используемый транспортным протоколом.";
    }
    uses traffic-unit;
  }
```

```
    grouping traffic-unit-port-all {
      description
        "Группировка трафика данного номера порта как функции
         единицы измерения, включая текущие значения.";
      uses traffic-unit-port;
      leaf current-g {
        type yang:gauge64;
        description
          "Текущее наблюдаемое значение.";
      }
    }

    grouping total-connection-capacity {
      description
        "Общая ёмкость различных типов соединений, а также суммарная
         пропускная способность. Эти узлы данных полезны для обнаружения
         нацеленных на поглощение ресурсов DDoS-атак.";
      leaf connection {
        type uint64;
        description
          "Максимальное число разрешённых одновременных соединений с
           целевым сервером.";
      }
      leaf connection-client {
        type uint64;
        description
          "Максимальное число разрешённых одновременных соединений с
           целевым сервером на клиента.";
      }
      leaf embryonic {
        type uint64;
        description
          "Максимальное число разрешённых одновременных эмбриональных
           соединений с сервером. Эмбриональным считается соединение, где
           согласование не завершено. Такие соединения возможны лишь для
           ориентированных на соединения протоколов, таких как TCP и SCTP.";
      }
      leaf embryonic-client {
        type uint64;
        description
          " Максимальное число разрешённых одновременных эмбриональных
           соединений с сервером на клиента.";
      }
      leaf connection-ps {
        type uint64;
        description
          "Максимальное число новых соединений с целевым сервером в
           секунду.";
      }
      leaf connection-client-ps {
        type uint64;
        description
          "Максимальное число новых соединений с целевым сервером в
           секунду на клиента.";
      }
      leaf request-ps {
        type uint64;
        description
          "Максимальное число запросов в секунду для целевого сервера.";
      }
      leaf request-client-ps {
        type uint64;
        description
          "Максимальное число запросов в секунду для целевого сервера
           на клиента.";
      }
      leaf partial-request-max {
        type uint64;
        description
          "Максимальное число остающихся частичных запросов для целевого
           сервера.";
      }
      leaf partial-request-client-max {
        type uint64;
        description
          "Максимальное число остающихся частичных запросов для целевого
           сервера на клиента.";
      }
    }

    grouping total-connection-capacity-protocol {
      description
        "Общая ёмкость соединения на протокол. Эти узлы данных полезны для
         обнаружения нацеленных на поглощение ресурсов DDoS-атак.";
      leaf protocol {
        type uint8;
```

```
      description
        "Транспортный протокол из реестра IANA Protocol Numbers
         <https://www.iana.org/assignments/protocol-numbers/>.";
    }
    uses total-connection-capacity;
  }

  grouping connection-percentile-and-peak {
    description
      "Узлы данных для представления характеристик атаки.";
    container connection-c {
      uses percentile-and-peak;
      description
        "Число одновременных соединений с сервером для атаки.";
    }
    container embryonic-c {
      uses percentile-and-peak;
      description
        "Число одновременных эмбриональных соединений с сервером
         для атаки.";
    }
    container connection-ps-c {
      uses percentile-and-peak;
      description
        "Число соединений с сервером в секунду для атаки.";
    }
    container request-ps-c {
      uses percentile-and-peak;
      description
        "Число запросов к серверу в секунду для атаки.";
    }
    container partial-request-c {
      uses percentile-and-peak;
      description
        "Число частичных запросов к серверу в секунду для атаки.";
    }
  }

  grouping connection-all {
    description
      "Всего соединений в атаке с учётом текущих значений.";
    container connection-c {
      uses percentile-peak-and-current;
      description
        "Число одновременных соединений с сервером для атаки.";
    }
    container embryonic-c {
      uses percentile-peak-and-current;
      description
        "Число одновременных эмбриональных соединений с сервером
         для атаки.";
    }
    container connection-ps-c {
      uses percentile-peak-and-current;
      description
        "Число соединений с сервером в секунду для атаки.";
    }
    container request-ps-c {
      uses percentile-peak-and-current;
      description
        "Число запросов к серверу в секунду для атаки.";
    }
    container partial-request-c {
      uses percentile-peak-and-current;
      description
        "Число частичных запросов к серверу в секунду для атаки.";
    }
  }

  grouping connection-protocol {
    description
      "Общее число соединений для атаки.";
    leaf protocol {
      type uint8;
      description
        "Транспортый протокол из реестра IANA Protocol Numbers
         <https://www.iana.org/assignments/protocol-numbers/>.";
    }
    uses connection-percentile-and-peak;
  }

  grouping connection-port {
    description
      "Общее число соединений для атаки с данным портом.";
    leaf protocol {
      type uint8;
```

```
        description
          "Транспортый протокол из реестра IANA Protocol Numbers
           <https://www.iana.org/assignments/protocol-numbers/>.";
        }
        leaf port {
          type inet:port-number;
          description
            "Номер порта.";
        }
        uses connection-percentile-and-peak;
      }

      grouping connection-protocol-all {
        description
          "Общее число соединений для атаки, включая текущие значения.";
        leaf protocol {
          type uint8;
          description
            "Транспортый протокол из реестра IANA Protocol Numbers
             <https://www.iana.org/assignments/protocol-numbers/>.";
        }
        uses connection-all;
      }

      grouping connection-protocol-port-all {
        description
          "Общее число соединений для атаки на порт, включая
           текущие значения.";
        leaf protocol {
          type uint8;
          description
            "Транспортый протокол из реестра IANA Protocol Numbers
             <https://www.iana.org/assignments/protocol-numbers/>.";
        }
        leaf port {
          type inet:port-number;
          description
            "Номер порта.";
        }
        uses connection-all;
      }

      grouping attack-detail {
        description
          "Детали, описывающие происходящие атаки, которые нужно смягчать
           с помощью сервера DOTS. Детали должны охватывать общие и
           хорошо известные атаки (такие, как SYN flood), а также новые
           и связанные с определёнными производителями атаки.";
        leaf vendor-id {
          type uint32;
          description
            "Vendor ID - идентификатор производителя из реестра IANA
             Private Enterprise Number.";
          reference
            "IANA: Private Enterprise Numbers
             (https://www.iana.org/assignments/enterprise-numbers/)";
        }
        leaf attack-id {
          type uint32;
          description
            "Уникальный идентификатор, заданный для атаки производителем.";
        }
        leaf description-lang {
          type string {
            pattern '((([A-Za-z]{2,3}(-[A-Za-z]{3}(-[A-Za-z]{3})'
                  + '{0,2})?)|[A-Za-z]{4}|[A-Za-z]{5,8})(-[A-Za-z]{4})'
                  + '?(-([A-Za-z]{2}|[0-9]{3}))?(-([A-Za-z0-9]{5,8}'
                  + '|([0-9][A-Za-z0-9]{3})))*(-[0-9A-WYZa-wyz]'
                  + '(-([A-Za-z0-9]{2,8}))+)*(-[Xx](-([A-Za-z0-9]'
                  + '{1,8}))+)?|[Xx](-([A-Za-z0-9]{1,8}))+|'
                  + '(([Ee][Nn]-[Gg][Bb]-[Oo][Ee][Dd]|[Ii]-'
                  + '[Aa][Mm][Ii]|[Ii]-[Bb][Nn][Nn]|[Ii]-'
                  + '[Dd][Ee][Ff][Aa][Uu][Ll][Tt]|[Ii]-'
                  + '[Ee][Nn][Oo][Cc][Hh][Ii][Aa][Nn]'
                  + '|[Ii]-[Hh][Aa][Kk]|'
                  + '[Ii]-[Kk][Ll][Ii][Nn][Gg][Oo][Nn]|'
                  + '[Ii]-[Ll][Uu][Xx]|[Ii]-[Mm][Ii][Nn][Gg][Oo]|'
                  + '[Ii]-[Nn][Aa][Vv][Aa][Jj][Oo]|[Ii]-[Pp][Ww][Nn]|'
                  + '[Ii]-[Tt][Aa][Oo]|[Ii]-[Tt][Aa][Yy]|'
                  + '[Ii]-[Tt][Ss][Uu]|[Ss][Gg][Nn]-[Bb][Ee]-[Ff][Rr]|'
                  + '[Ss][Gg][Nn]-[Bb][Ee]-[Nn][Ll]|[Ss][Gg][Nn]-'
                  + '[Cc][Hh]-[Dd][Ee])|([Aa][Rr][Tt]-'
                  + '[Ll][Oo][Jj][Bb][Aa][Nn]|[Cc][Ee][Ll]-'
                  + '[Gg][Aa][Uu][Ll][Ii][Ss][Hh]|'
                  + '[Nn][Oo]-[Bb][Oo][Kk]|[Nn][Oo]-'
                  + '[Nn][Yy][Nn]|[Zz][Hh]-[Gg][Uu][Oo][Yy][Uu]|'
```

```
                + '[Zz][Hh]-[Hh][Aa][Kk][Kk][Aa]|[Zz][Hh]-'
                + '[Mm][Ii][Nn]|[Zz][Hh]-[Mm][Ii][Nn]-'
                + '[Nn][Aa][Nn]|[Zz][Hh]-[Xx][Ii][Aa][Nn][Gg]))))';
      }
      default "en-US";
      description
        "Тег языка для attack-description.";
      reference
        "RFC 5646: Tags for Identifying Languages, Section 2.1";
    }
    leaf attack-description {
      type string;
      description
        "Текстовое описание атаки. Методы обработки естественных
         языков (например, встраивание слов) могут быть полезны
         в сопоставлении описания атаки с её типом.";
    }
    leaf attack-severity {
      type attack-severity;
      description
        "Уровень серьёзности атаки, определяемый реализацией.";
    }
    leaf start-time {
      type uint64;
      description
        "Время начала атаки в секундах с 1970-01-01T00:00:00Z.";
    }
    leaf end-time {
      type uint64;
      description
        "Время завершения атаки в секундах с 1970-01-01T00:00:00Z.";
    }
    container source-count {
      description
        "Число уникальных источников, вовлечённых в атаку.";
      uses percentile-and-peak;
      leaf current-g {
        type yang:gauge64;
        description
          "Текущее наблюдаемое значение.";
      }
    }
  }

  grouping talker {
    description
      "Определяет базовые данные о наиболее активных участниках.";
    leaf spoofed-status {
      type boolean;
      description
        "Значение true указывает, что адрес подменен.";
    }
    leaf source-prefix {
      type inet:ip-prefix;
      description
        "Префикс IPv4 или IPv6, указывающий атакующих.";
    }
    list source-port-range {
      key "lower-port";
      description
        "Диапазон портов. Наличие лишь lower-port указывает 1 порт.";
      leaf lower-port {
        type inet:port-number;
        description
          "Меньший номер диапазона портов.";
      }
      leaf upper-port {
        type inet:port-number;
        must '. >= ../lower-port' {
          error-message
            "Старший порт должен иметь номер больше, чем младший.";
        }
        description
          "Больший номер диапазона портов.";
      }
    }
    list source-icmp-type-range {
      key "lower-type";
      description
        "Диапазон типов ICMP. Указание лишь lower-type представляет
         один тип ICMP.";
      leaf lower-type {
        type uint8;
        description
          "Меньшее значение диапазона типов ICMP.";
      }
```

```
        leaf upper-type {
          type uint8;
          must '. >= ../lower-type' {
            error-message
              "Верхнее значение типа ICMP должно быть больше нижнего.";
          }
          description
            "Большее значение диапазона типов ICMP.";
        }
      }
      list total-attack-traffic {
        key "unit";
        description
          "Суммарный трафик атаки от данного источника.";
        uses traffic-unit-all;
      }
    }

    grouping top-talker-aggregate {
      description
        "Агрегат основных источников атаки. Обычно служит для
         включения в запрос на смягчение.";
      list talker {
        key "source-prefix";
        description
          "Основные участники атаки, указанные префиксом IPv4 или IPv6.";
        uses talker;
        container total-attack-connection {
          description
            "Общее число соединений атаки от данного источника.";
          uses connection-all;
        }
      }
    }

    grouping top-talker {
      description
        "Основные источники атаки с деталями по протоколам.";
      list talker {
        key "source-prefix";
        description
          "Основные участники атаки, указанные префиксом IPv4 или IPv6.";
        uses talker;
        list total-attack-connection-protocol {
          key "protocol";
          description
            "Общее число соединений атаки от данного источника.";
          uses connection-protocol-all;
        }
      }
    }

    grouping baseline {
      description
        "Группировка для базового уровня телеметрии.";
      uses data-channel:target;
      leaf-list alias-name {
        type string;
        description
          "Псевдоним ресурса IP (маршрутизатор, хост, объект (IoT),
           сервер и т. п.";
      }
      list total-traffic-normal {
        key "unit";
        description
          "Общий уровень нормального трафика.";
        uses traffic-unit;
      }
      list total-traffic-normal-per-protocol {
        key "unit protocol";
        description
          "Общий уровень нормального трафика на протокол.";
        uses traffic-unit-protocol;
      }
      list total-traffic-normal-per-port {
        key "unit port";
        description
          "Общий уровень нормального трафика на номер порта.";
        uses traffic-unit-port;
      }
      list total-connection-capacity {
        key "protocol";
        description
          "Общая пропускная способность соединений.";
        uses total-connection-capacity-protocol;
      }
```

```
      list total-connection-capacity-per-port {
        key "protocol port";
        description
          "Общая пропускная способность соединений с портом.";
        leaf port {
          type inet:port-number;
          description
            "Номер целевого порта.";
        }
        uses total-connection-capacity-protocol;
      }
    }

    grouping pre-or-ongoing-mitigation {
      description
        "Группировка для данных телеметрии.";
      list total-traffic {
        key "unit";
        description
          "Общий трафик.";
        uses traffic-unit-all;
      }
      list total-traffic-protocol {
        key "unit protocol";
        description
          "Общий трафик для протокола.";
        uses traffic-unit-protocol-all;
      }
      list total-traffic-port {
        key "unit port";
        description
          "Общий трафик для номера порта.";
        uses traffic-unit-port-all;
      }
      list total-attack-traffic {
        key "unit";
        description
          "Общий трафик атаки.";
        uses traffic-unit-all;
      }
      list total-attack-traffic-protocol {
        key "unit protocol";
        description
          "Общий трафик атаки для протокола.";
        uses traffic-unit-protocol-all;
      }
      list total-attack-traffic-port {
        key "unit port";
        description
          "Общий трафик атаки для номера порта.";
        uses traffic-unit-port-all;
      }
      list total-attack-connection-protocol {
        key "protocol";
        description
          "Общее число соединений атаки.";
        uses connection-protocol-all;
      }
      list total-attack-connection-port {
        key "protocol port";
        description
          "Общее число соединений атаки для целевого порта.";
        uses connection-protocol-port-all;
      }
      list attack-detail {
        key "vendor-id attack-id";
        description
          "Детали атаки.";
        uses attack-detail;
        container top-talker {
          description
            "Список источников атаки.";
          uses top-talker;
        }
      }
    }

    sx:augment-structure "/dots-signal:dots-signal"
                         + "/dots-signal:message-type"
                         + "/dots-signal:mitigation-scope"
                         + "/dots-signal:scope" {
      description
        "Расширяет смягчение атаки данными обновления телеметрии.";
      choice direction {
        description
          "Направление связи, в котором могут включаться узлы данных.";
```

```
        case server-to-client-only {
          description
            "Эти узлы данных появляются лишь в сообщениях смягчения,
             передаваемых от сервера клиенту.";
          list total-traffic {
            key "unit";
            description
              "Общий трафик.";
            uses traffic-unit-all;
          }
          container total-attack-connection {
            description
              "Общее число соединений атаки.";
            uses connection-all;
          }
        }
      }
      list total-attack-traffic {
        key "unit";
        description
              "Общий трафик атаки.";
        uses traffic-unit-all;
      }
      list attack-detail {
        key "vendor-id attack-id";
        description
          "Детали атаки.";
        uses attack-detail;
        container top-talker {
          description
            "Основные источники атаки.";
          uses top-talker-aggregate;
        }
      }
    }
    sx:structure dots-telemetry {
      description
        "Основная структура для сообщений телеметрии DOTS.";
      choice telemetry-message-type {
        description
          "Может быть telemetry-setup или данными телеметрии.";
        case telemetry-setup {
          description
            "Указывает сообщение об установке телеметрии.";
          choice direction {
            description
              "Направление связи, в котором могут включаться узлы данных.";
            case server-to-client-only {
              description
               "Эти узлы данных появляются лишь в сообщениях телеметрии,
                передаваемых от сервера клиенту.";
              container max-config-values {
                description
                  "Максимально допустимые значения конфигурации.";
                uses telemetry-parameters;
                leaf server-originated-telemetry {
                  type boolean;
                  default "false";
                  description
                    "Указывает, можно ли задать серверу DOTS передачу
                     телеметрии pre-or-ongoing-mitigation. Значение false
                     или отсутствие узла данных указывает, что сервер не
                     поддерживает такую возможность.";
                }
                leaf telemetry-notify-interval {
                  type uint16 {
                    range "1 .. 3600";
                  }
                  units "seconds";
                  must '. >= ../../min-config-values'
                     + '/telemetry-notify-interval' {
                    error-message
                      "Значение должно быть не меньше telemetry-notify-
                       interval в атрибуте min-config-values.";
                  }
                  description
                    "Минимальное число секунд между последовательными
                     уведомлениями телеметрии.";
                }
              }
              container min-config-values {
                description
                  "Минимально допустимые значения конфигурации .";
                uses telemetry-parameters;
                leaf telemetry-notify-interval {
                  type uint16 {
```

```
            range "1 .. 3600";
          }
          units "seconds";
          description
            "Минимальное число секунд между последовательными
             уведомлениями телеметрии.";
        }
      }
      container supported-unit-classes {
        description
          "Поддерживаемые классы единиц измерения и принятый
           по умолчанию статус активации.";
        uses unit-config;
      }
      leaf-list supported-query-type {
        type query-type;
        description
          "Типы поддерживаемых сервером запросов. Если сервер не
           анонсирует поддерживаемые типы, клиент не сможет
           использовать какие-либо значения query-type для снижения
           объёма данных от сервера.";
      }
    }
  }
  list telemetry {
    description
      "Данные телеметрии для клиента DOTS. Ключами этого списка
       служат cuid и tsid, но они не представлены здесь, поскольку
       обязательны в запросах Uri-Paths. Отсутствие ключей
       согласуется с RFC 8791.";
    reference
      "RFC 8791: YANG Data Structure Extensions";
    choice direction {
      description
        "Направление связи, в котором можно включать узлы данных.";
      case server-to-client-only {
        description
          "Эти узлы данных появляются лишь в сообщениях телеметрии,
           передаваемых от сервера клиенту.";
        leaf tsid {
          type uint32;
          description
            "Заданный клиентом идентификатор для данных установки
             телеметрии DOTS.";
        }
      }
    }
    choice setup-type {
      description
        "Может указывать конфигурацию смягчения, ёмкость трубы,
         или базовый уровень.";
      case telemetry-config {
        description
          "Служит для установки параметров телеметрии, таких как
           значения low-, mid-, high-percentile.";
        container current-config {
          description
            "Текущие значения конфигурации телеметрии.";
          uses telemetry-parameters;
          uses unit-config;
          leaf server-originated-telemetry {
            type boolean;
            description
              "Применяется клиентом DOTS для управления
               возможностью запросов телеметрии
               pre-or-ongoing-mitigation у сервера.";
          }
          leaf telemetry-notify-interval {
            type uint16 {
              range "1 .. 3600";
            }
            units "seconds";
            description
              "Минимальное число секунд между последовательными
               уведомлениями телеметрии.";
          }
        }
      }
      case pipe {
        description
          "Общая емкость трубы клиентского домена DOTS.";
        list total-pipe-capacity {
          key "link-id unit";
          description
            "Общая емкость трубы клиентского домена DOTS.";
          leaf link-id {
```

```
                type nt:link-id;
                description
                  "Идентификатор внешнего канала клиентского домена
                   DOTS.";
              }
              leaf capacity {
                type uint64;
                mandatory true;
                description
                  "Ёмкость трубы. Обязательный атрибут при наличии
                   total-pipe-capacity в сообщении.";
              }
              leaf unit {
                type unit;
                description
                  "Трафик можно измерять в пакетах (pps), битах (bps)
                    и/или байтах (Bps) за секунду.

                   Для данного класса единиц агенты DOTS автоматически
                   приводят значения к соответствующим единицам
                   (например, megabit-ps, kilobit-ps).";
              }
            }
          }
          case baseline {
            description
              "Данные о базовом уровне трафика клиентского домена
               DOTS.";
            list baseline {
              key "id";
              description
                "Данные о базовом уровне трафика клиентского домена
                 DOTS.";
              leaf id {
                type uint32;
                must '. >= 1';
                description
                  "Идентификатор, однозначно указывающий запись
                   базового уровня, переданную клиентом DOTS.";
              }
              uses baseline;
            }
          }
        }
      }
    }
    case telemetry {
      description
        "Данные телеметрии.";
      list pre-or-ongoing-mitigation {
        description
          "Телеметрия Pre-or-ongoing-mitigation по клиенту DOTS.
           Ключами этого списка служат cuid и tsid, но они не
           представлены здесь, поскольку обязательны в запросах
           Uri-Paths. Отсутствие ключей согласуется с RFC 8791.";
        reference
          "RFC 8791: YANG Data Structure Extensions";
        choice direction {
          description
            "Направление, в котором могут включаться узлы данных.";
          case server-to-client-only {
            description
              "Эти узлы данных появляются лишь в сообщениях телеметрии,
               передаваемых от сервера клиенту.";
            leaf tmid {
              type uint32;
              description
                "Заданный клиентом идентификатор данных телеметрии
                 DOTS.";
            }
          }
        }
        container target {
          description
            "Указывает цель. В определении цели должен присутствовать
             хотя бы один из атрибутов target-prefix, target-fqdn,
             target-uri, alias-name, mid-list.";
          uses data-channel:target;
          leaf-list alias-name {
            type string;
            description
              "Псевдоним ресурса.";
          }
          leaf-list mid-list {
            type uint32;
            description
```

```
                "Ссылка на список связанных запросов смягчения.";
              reference
                "RFC 9132: Distributed Denial-of-Service Open
                           Threat Signaling (DOTS) Signal Channel
                           Specification, Section 4.4.1";
            }
          }
          uses pre-or-ongoing-mitigation;
        }
      }
    }
  }
}
<CODE ENDS>
```

11.2. Модуль YANG для деталей отображения атак от производителя

```
<CODE BEGINS> file "ietf-dots-mapping@2022-06-20.yang"
module ietf-dots-mapping {
  yang-version 1.1;
  namespace "urn:ietf:params:xml:ns:yang:ietf-dots-mapping";
  prefix dots-mapping;

  import ietf-dots-data-channel {
    prefix data-channel;
    reference
      "RFC 8783: Distributed Denial-of-Service Open Threat
                 Signaling (DOTS) Data Channel Specification";
  }

  organization
    "IETF DDoS Open Threat Signaling (DOTS) Working Group";
  contact
    "WG Web:   <https://datatracker.ietf.org/wg/dots/>
     WG List:  <mailto:dots@ietf.org>

     Author:   Mohamed Boucadair
               <mailto:mohamed.boucadair@orange.com>

     Author:   Jon Shallow
               <mailto:supjps-ietf@jpshallow.com>";
  description
    "Этот модуль содержит определения YANG для совместного использования
     деталей отображения атак DDoS клиентами и серверами DOTS,
     передаваемых по сигнальному каналу DOTS.

     Авторские права (Copyright (c) 2022) принадлежат IETF Trust и
     лицам, указанным как авторы. Все права защищены.

     Распространение и применение модуля в исходной или двоичной
     форме с изменениями или без таковых разрешено в соответствии с
     лицензией Simplified BSD License, изложенной в параграфе 4.c
     IETF Trust's Legal Provisions Relating to IETF Documents
     (https://trustee.ietf.org/license-info).

     Эта версия модуля YANG является частью RFC 9244, где правовые
     аспекты приведены более полно.";

  revision 2022-06-20 {
    description
      "Исходный выпуск.";
    reference
      "RFC 9244: Distributed Denial-of-Service Open Threat
                 Signaling (DOTS) Telemetry";
  }

  feature dots-telemetry {
    description
      "Показывает возможность совместного использования
       телеметрии DOTS клиентами и серверами.";
  }

  grouping attack-mapping {
    description
      "Набор сведений, используемых для совместного с партнёром
       использования данных отображения атак от производителя.";
    list vendor {
      key "vendor-id";
      description
        "Сведения отображения атак от производителя, относящиеся
         к клиенту/серверу.";
      leaf vendor-id {
        type uint32;
        description
          "Идентификатор производителя средств защиты из реестра
           IANA Private Enterprise Numbers.";
```

```
        reference
          "IANA: Private Enterprise Numbers
           (https://www.iana.org/assignments/enterprise-numbers/)";
      }
      leaf vendor-name {
        type string;
        description
          "Имя производителя (например, компания A).";
      }
      leaf description-lang {
        type string {
          pattern '((([A-Za-z]{2,3}(-[A-Za-z]{3}(-[A-Za-z]{3})'
                + '{0,2})?)|[A-Za-z]{4}|[A-Za-z]{5,8})(-[A-Za-z]{4})'
                + '?(-([A-Za-z]{2}|[0-9]{3}))?(-([A-Za-z0-9]{5,8}'
                + '|([0-9][A-Za-z0-9]{3})))*(-[0-9A-WYZa-wyz]'
                + '(-([A-Za-z0-9]{2,8}))+)*(-[Xx](-([A-Za-z0-9]'
                + '{1,8}))+)?|[Xx](-([A-Za-z0-9]{1,8}))+|'
                + '(([Ee][Nn]-[Gg][Bb]-[Oo][Ee][Dd]|[Ii]-'
                + '[Aa][Mm][Ii]|[Ii]-[Bb][Nn][Nn]|[Ii]-'
                + '[Dd][Ee][Ff][Aa][Uu][Ll][Tt]|[Ii]-'
                + '[Ee][Nn][Oo][Cc][Hh][Ii][Aa][Nn]'
                + '|[Ii]-[Hh][Aa][Kk]|'
                + '[Ii]-[Kk][Ll][Ii][Nn][Gg][Oo][Nn]|'
                + '[Ii]-[Ll][Uu][Xx]|[Ii]-[Mm][Ii][Nn][Gg][Oo]|'
                + '[Ii]-[Nn][Aa][Vv][Aa][Jj][Oo]|[Ii]-[Pp][Ww][Nn]|'
                + '[Ii]-[Tt][Aa][Oo]|[Ii]-[Tt][Aa][Yy]|'
                + '[Ii]-[Tt][Ss][Uu]|[Ss][Gg][Nn]-[Bb][Ee]-[Ff][Rr]|'
                + '[Ss][Gg][Nn]-[Bb][Ee]-[Nn][Ll]|[Ss][Gg][Nn]-'
                + '[Cc][Hh]-[Dd][Ee])|([Aa][Rr][Tt]-'
                + '[Ll][Oo][Jj][Bb][Aa][Nn]|[Cc][Ee][Ll]-'
                + '[Gg][Aa][Uu][Ll][Ii][Ss][Hh]|'
                + '[Nn][Oo]-[Bb][Oo][Kk]|[Nn][Oo]-'
                + '[Nn][Yy][Nn]|[Zz][Hh]-[Gg][Uu][Oo][Yy][Uu]|'
                + '[Zz][Hh]-[Hh][Aa][Kk][Kk][Aa]|[Zz][Hh]-'
                + '[Mm][Ii][Nn]|[Zz][Hh]-[Mm][Ii][Nn]-'
                + '[Nn][Aa][Nn]|[Zz][Hh]-[Xx][Ii][Aa][Nn][Gg]))))';
        }
        default "en-US";
        description
          "Тег языка, применяемого в attack-description.";
        reference
          "RFC 5646: Tags for Identifying Languages, Section 2.1";
      }
      leaf last-updated {
        type uint64;
        mandatory true;
        description
          "Время обновления таблицы отображений в секундах с
           1970-01-01T00:00:00Z.";
      }
      list attack-mapping {
        key "attack-id";
        description
          "Attack mapping details.";
        leaf attack-id {
          type uint32;
          description
            "Unique identifier assigned by the vendor for the
             attack.";
        }
        leaf attack-description {
          type string;
          mandatory true;
          description
            "Текстовое описание атаки. Методы обработки естественных
             языков (например, встраивание слов) могут быть полезны
             для сопоставляния описаний атак с их типами.";
        }
      }
    }
  }

  augment "/data-channel:dots-data/data-channel:dots-client" {
    if-feature "dots-telemetry";
    description
      "Дополняет канал данных клиентской таблицей отображений атак от
       производителя.";
    container vendor-mapping {
      description
        "Применяется клиентом DOTS для использования его сведений
         об отображении атак от производителя совместно с сервером.";
      uses attack-mapping;
    }
  }

  augment "/data-channel:dots-data/data-channel:capabilities" {
```

```
      if-feature "dots-telemetry";
      description
        "Дополняет возможности сервера DOTS параметром для индикации
         возможности совместного использования деталей отображения атак.";
      leaf vendor-mapping-enabled {
        type boolean;
        config false;
        description
          "Указывает, что сервер DOTS поддерживает использование деталей
           отображения атак от производителя совместно с клиентом.";
      }
    }

    augment "/data-channel:dots-data" {
      if-feature "dots-telemetry";
      description
        "Дополняет канал данных таблицей отображений атак от производителя
         от сервера DOTS.";
      container vendor-mapping {
        config false;
        description
          "Список деталей отображения атак от производителя, которые будут
           использоваться совместно с клиентами DOTS по их запросам.";
        uses attack-mapping;
      }
    }
  }
  <CODE ENDS>
```

12. Сопоставление параметров YANG/JSON и CBOR

Все параметры телеметрии DOTS в полях данных сигнального канала DOTS **должны** отображаться на типы CBOR, как показано в таблице 3.

Примечание. Разработчики должны убедиться, что выход отображений, обеспечиваемых схемами преобразования YANG в CBOR, соответствует таблице 3.

Таблица 3. Сопоставление параметров JSON/YANG и CBOR.

Имя параметра	Тип YANG	Ключ CBOR	Старший тип и данные CBOR	Тип JSON
tsid	uint32	128	0 unsigned	Number
telemetry	list	129	4 array	Array
low-percentile	decimal64	130	6 tag 4 [-2, integer]	String
mid-percentile	decimal64	131	6 tag 4 [-2, integer]	String
high-percentile	decimal64	132	6 tag 4 [-2, integer]	String
unit-config	list	133	4 array	Array
unit	enumeration	134	0 unsigned	String
unit-status	boolean	135	7 bits 20	False
			7 bits 21	True
total-pipe-capacity	list	136	4 array	Array
link-id	string	137	3 text string	String
pre-or-ongoing-mitigation	list	138	4 array	Array
total-traffic-normal	list	139	4 array	Array
low-percentile-g	yang:gauge64	140	0 unsigned	String
mid-percentile-g	yang:gauge64	141	0 unsigned	String
high-percentile-g	yang:gauge64	142	0 unsigned	String
peak-g	yang:gauge64	143	0 unsigned	String
total-attack-traffic	list	144	4 array	Array
total-traffic	list	145	4 array	Array
total-connection-capacity	list	146	4 array	Array
connection	uint64	147	0 unsigned	String
connection-client	uint64	148	0 unsigned	String
embryonic	uint64	149	0 unsigned	String
embryonic-client	uint64	150	0 unsigned	String
connection-ps	uint64	151	0 unsigned	String
connection-client-ps	uint64	152	0 unsigned	String
request-ps	uint64	153	0 unsigned	String
request-client-ps	uint64	154	0 unsigned	String
partial-request-max	uint64	155	0 unsigned	String
partial-request-client-max	uint64	156	0 unsigned	String
total-attack-connection	container	157	5 map	Object
connection-c	container	158	5 map	Object
embryonic-c	container	159	5 map	Object
connection-ps-c	container	160	5 map	Object
request-ps-c	container	161	5 map	Object
attack-detail	list	162	4 array	Array
id	uint32	163	0 unsigned	Number
attack-id	uint32	164	0 unsigned	Number
attack-description	string	165	3 text string	String
attack-severity	enumeration	166	0 unsigned	String
start-time	uint64	167	0 unsigned	String
end-time	uint64	168	0 unsigned	String

source-count	container	169	5 map	Object
top-talker	container	170	5 map	Object
spoofed-status	boolean	171	7 bits 20	False
			7 bits 21	True
partial-request-c	container	172	5 map	Object
total-attack-connection-protocol	list	173	4 array	Array
baseline	list	174	4 array	Array
current-config	container	175	5 map	Object
max-config-values	container	176	5 map	Object
min-config-values	container	177	5 map	Object
supported-unit-classes	container	178	5 map	Object
server-originated-telemetry	boolean	179	7 bits 20	False
			7 bits 21	True
telemetry-notify-interval	uint16	180	0 unsigned	Number
tmid	uint32	181	0 unsigned	Number
measurement-interval	enumeration	182	0 unsigned	String
measurement-sample	enumeration	183	0 unsigned	String
talker	list	184	4 array	Array
source-prefix	inet:ip-prefix	185	3 text string	String
mid-list	leaf-list	186	4 array	Array
	uint32		0 unsigned	Number
source-port-range	list	187	4 array	Array
source-icmp-type-range	list	188	4 array	Array
target	container	189	5 map	Object
capacity	uint64	190	0 unsigned	String
protocol	uint8	191	0 unsigned	Number
total-traffic-normal-per-protocol	list	192	4 array	Array
total-traffic-normal-per-port	list	193	4 array	Array
total-connection-capacity-per-port	list	194	4 array	Array
total-traffic-protocol	list	195	4 array	Array
total-traffic-port	list	196	4 array	Array
total-attack-traffic-protocol	list	197	4 array	Array
total-attack-traffic-port	list	198	4 array	Array
total-attack-connection-port	list	199	4 array	Array
port	inet:port-number	200	0 unsigned	Number
supported-query-type	leaf-list	201	4 array	Array
			0 unsigned	String
vendor-id	uint32	202	0 unsigned	Number
ietf-dots-telemetry:telemetry-setup	container	203	5 map	Object
ietf-dots-telemetry:total-traffic	list	204	4 array	Array
ietf-dots-telemetry:total-attack-traffic	list	205	4 array	Array
ietf-dots-telemetry:total-attack-connection	container	206	5 map	Object
ietf-dots-telemetry:attack-detail	list	207	4 array	Array
ietf-dots-telemetry:telemetry	container	208	5 map	Object
current-g	yang:gauge64	209	0 unsigned	String
description-lang	string	210	3 text string	String
lower-type	uint8	32771	0 unsigned	Number
upper-type	uint8	32772	0 unsigned	Number

13. Взаимодействие с IANA

13.1. Значения ключей CBOR

Эта спецификация регистрирует приведенные в таблице 4 необязательные для понимания параметры в реестре IANA «DOTS Signal Channel CBOR Key Values» [Key-Map].

Таблица 4. Зарегистрированные ключи CBOR для сигнального канала DOTS.

Имя параметра	***Значение ключа CBOR***	***Старший тип CBOR***	***Контролёр изменений***	***Документ***
tsid	128	0	IESG	RFC 9244
telemetry	129	4	IESG	RFC 9244
low-percentile	130	6tag4	IESG	RFC 9244
mid-percentile	131	6tag4	IESG	RFC 9244

high-percentile	132	6tag4	IESG	RFC 9244
unit-config	133	4	IESG	RFC 9244
unit	134	0	IESG	RFC 9244
unit-status	135	7	IESG	RFC 9244
total-pipe-capacity	136	4	IESG	RFC 9244
link-id	137	3	IESG	RFC 9244
pre-or-ongoing-mitigation	138	4	IESG	RFC 9244
total-traffic-normal	139	4	IESG	RFC 9244
low-percentile-g	140	0	IESG	RFC 9244
mid-percentile-g	141	0	IESG	RFC 9244
high-percentile-g	142	0	IESG	RFC 9244
peak-g	143	0	IESG	RFC 9244
total-attack-traffic	144	4	IESG	RFC 9244
total-traffic	145	4	IESG	RFC 9244
total-connection-capacity	146	4	IESG	RFC 9244
connection	147	0	IESG	RFC 9244
connection-client	148	0	IESG	RFC 9244
embryonic	149	0	IESG	RFC 9244
embryonic-client	150	0	IESG	RFC 9244

connection-ps	151	0	IESG	RFC 9244
connection-client-ps	152	0	IESG	RFC 9244
request-ps	153	0	IESG	RFC 9244
request-client-ps	154	0	IESG	RFC 9244
partial-request-max	155	0	IESG	RFC 9244
partial-request-client-max	156	0	IESG	RFC 9244
total-attack-connection	157	5	IESG	RFC 9244
connection-c	158	5	IESG	RFC 9244
embryonic-c	159	5	IESG	RFC 9244
connection-ps-c	160	5	IESG	RFC 9244
request-ps-c	161	5	IESG	RFC 9244
attack-detail	162	4	IESG	RFC 9244
id	163	0	IESG	RFC 9244
attack-id	164	0	IESG	RFC 9244
attack-description	165	3	IESG	RFC 9244
attack-severity	166	0	IESG	RFC 9244
start-time	167	0	IESG	RFC 9244
end-time	168	0	IESG	RFC 9244
source-count	169	5	IESG	RFC 9244

top-talker	170	5	IESG	RFC 9244
spoofed-status	171	7	IESG	RFC 9244
partial-request-c	172	5	IESG	RFC 9244
total-attack-connection-protocol	173	4	IESG	RFC 9244
baseline	174	4	IESG	RFC 9244
current-config	175	5	IESG	RFC 9244
max-config-values	176	5	IESG	RFC 9244
min-config-values	177	5	IESG	RFC 9244
supported-unit-classes	178	5	IESG	RFC 9244
server-originated-telemetry	179	7	IESG	RFC 9244
telemetry-notify-interval	180	0	IESG	RFC 9244
tmid	181	0	IESG	RFC 9244
measurement-interval	182	0	IESG	RFC 9244
measurement-sample	183	0	IESG	RFC 9244
talker	184	4	IESG	RFC 9244
source-prefix	185	3	IESG	RFC 9244
mid-list	186	4	IESG	RFC 9244
source-port-range	187	4	IESG	RFC 9244
source-icmp-type-range	188	4	IESG	RFC 9244

target	189	5	IESG	RFC 9244
capacity	190	0	IESG	RFC 9244
protocol	191	0	IESG	RFC 9244
total-traffic-normal-per-protocol	192	4	IESG	RFC 9244
total-traffic-normal-per-port	193	4	IESG	RFC 9244
Total-connection- capacity-per-port	194	4	IESG	RFC 9244
total-traffic-protocol	195	4	IESG	RFC 9244
total-traffic-port	196	4	IESG	RFC 9244
total-attack-traffic-protocol	197	4	IESG	RFC 9244
total-attack-traffic-port	198	4	IESG	RFC 9244
total-attack-connection-port	199	4	IESG	RFC 9244
port	200	0	IESG	RFC 9244
supported-query-type	201	4	IESG	RFC 9244
vendor-id	202	0	IESG	RFC 9244
ietf-dots-telemetry:telemetry-setup	203	5	IESG	RFC 9244
ietf-dots-telemetry:total-traffic	204	4	IESG	RFC 9244
ietf-dots-telemetry:total-attack-traffic	205	4	IESG	RFC 9244
ietf-dots-telemetry:total-attack-connection	206	5	IESG	RFC 9244
ietf-dots-telemetry:attack-detail	207	4	IESG	RFC 9244

ietf-dots-telemetry:telemetry	208	5	IESG	RFC 9244
current-g	209	0	IESG	RFC 9244
description-lang	210	3	IESG	RFC 9244

13.2. Код причины конфликтов в сигнальном канале DOTS

В соответствии с этим документом агетство IANA выделило новый код в реестре DOTS Signal Channel Conflict Cause Codes [Cause].

Таблица 5. Зарегистрированные коды причин конфликтов в сигнальном канале DOTS.

Код	***Метка***	***Описание***	***Документ***
5	overlapping-pipes	Перекрытие области действия трубы	RFC 9244

13.3. Регистрация DOTS Telemetry URI и модулей YANG

В соответствии с этим документом агетство IANA зарегистрировало приведённые ниже UR в субреестре ns реестра IETF XML Registry [RFC3688]:

```
URI:  urn:ietf:params:xml:ns:yang:ietf-dots-telemetry
Registrant Contact:  The IESG.
XML:  N/A; the requested URI is an XML namespace.

URI:  urn:ietf:params:xml:ns:yang:ietf-dots-mapping
Registrant Contact:  The IESG.
XML:  N/A; the requested URI is an XML namespace.

Per this document, IANA has registered the following YANG modules in
the "YANG Module Names" subregistry [RFC6020] within the "YANG
Parameters" registry.

Name:  ietf-dots-telemetry
Namespace:  urn:ietf:params:xml:ns:yang:ietf-dots-telemetry
Maintained by IANA:  N
Prefix:  dots-telemetry
Reference:  RFC 9244

Name:  ietf-dots-mapping
Namespace:  urn:ietf:params:xml:ns:yang:ietf-dots-mapping
Maintained by IANA:  N
Prefix:  dots-mapping
Reference:  RFC 9244
```

14. Вопросы безопасности

14.1. Телеметрия в сигнальном канале DOTS

Вопросы безопасности для сигнального канала DOTS рассмотрены в разделе 11 [RFC9132], а ниже обсуждаются вопросы, связанные с безопасностью определённых в документе расширений сигнального канала DOTS.

Данные телеметрии DOTS включают топологию клиентской сети DOTS, пропускную способность «трубы» клиентского домена DOTS, базовый уровень нормального трафика, а также сведения об угрозах и их смягчении. Такая информация является конфиденциальной и **должна** быть защищена серверным доменом DOTS для предотвращения утечки данных. Отметим, что совместное использование этих конфиденциальных сведений с доверенным сервером DOTS не создаёт новых существенных проблем безопасности, кроме необходимости упомянутой защиты. Серверу DOTS уже предоставлен доступ к информации, поскольку он может наблюдать и смягчать атаки.

Клиенты DOTS обычно считаются доверенными устройствами в клиентском домене DOTS. Клиенты могут размещаться вместе с устройствами защиты сети (например, межсетевыми экранами) и взломанная служба защиты может нанести сети гораздо больший вред, незели просто клиентские компоненты DOTS. Это допущение отличается от распространенного мнения о том, что устройства не являются доверенными (это часто называют моделью нулевого доверия - zero-trust model). Взломанный клиент DOTS может передавать фальшивые данные телеметрии серверу DOTS, чтобы ввести тот в заблуждение. Такие атаки можно предотвратить, отслеживая и проверяя клиентов DOTS для обнаружения недопустимого поведения и его предотвращения и разрешая передачу данных телеметрии для конкретных ресурсов лишь уполномоченным клиентам DOTS (например, серверу приложений разрешается обмен телеметрией DOTS для его адреса IP, а системе смягчения DDoS-атак разрешено обмениваться телеметрией DOTS для любого целевого ресурса в сети). Напомним, что имеется вариант работы со взломанными клиентами DOTS, описанный в разделе 11 [RFC9132].

Серверы DOTS должны быть способны защитить себя от DoS-атак со стороны скомпрометированных клиентов DOTS. Ниже указаны некоторые методы смягчения, которые сервер DOTS может применять против таких клиентов DOTS.

- Скорость зондирования (probing rate, как определено в параграфе 4.5 [RFC9132]) может служить ограничением скорости передачи данных серверу DOTS.
- Телеметрия DOTS с ограничением скорости, включая пакеты с новыми значениями tmid от одного клиента DOTS, защищает от DoS-атак, которые могут приводит к изменению tmid для истощения ресурсов сервера

DOTS. Сервер DOTS может также установить квоту и ограничение по времени для числа активных элементов телеметрии до и во время смягчения атак (определяется по tmid) для клиента DOTS.

Отметим также, что можно применять интервал между уведомлениями телеметрии для ограничения скорости уведомлений pre-or-ongoing-mitigation, получаемых клиентским доменом DOTS.

14.2. Отображение атак от производителя

Вопросы безопасности для протокола канала данных DOTS рассмотрены в разделе 10 [RFC8783], а ниже приведены соображения, связанные с определённым в документе расширением канала данных DOTS.

Все узлы данных в модуле YANG, заданном в параграфе 11.2, которые можно создавать, изменять, удалять (т. е. узлы с config true, что принято по умолчанию) считаются чувствительными к угрозам. Операции записи в такие узлы без подобающей защиты могут оказывать негативное влияние на работу сети. Рекомендуется применять подходящие меры защиты, предотвращающие вызов примитивов канала данных DOTS недоверенными пользователями, как описано в [RFC8783]. Тем не менее, злоумышленник с доступом к клиенту DOTS технически способен организовать разные атаки, атке как передача серверу недействительных сведений о деталях отображения атак (/data-channel:dots-data/data-channel:dots-client/dots-telemetry:vendor-mapping), которые могут ввести сервер в заблуждение.

Некоторые доступные для чтения узлы данных модуля YANG, заданного в параграфе 11.2, могут содержать конфиденциальные сведения, поэтому важно контролировать считывание узлов, указанных ниже.

- /data-channel:dots-data/data-channel:dots-client/dots-telemetry:vendor-mapping могут использоваться для выяснения технологии защиты от атак DDoS, реализованной в клиентском домене DOTS.
- /data-channel:dots-data/dots-telemetry:vendor-mapping могут использоваться взломанными клиентами DOTS для утечки сведений о возможностях обнаружения атак с сервера DOTS. Это является вариантом атак со стороны скомпрометрированных клиентов DOTS, упомянутых в параграфе 14.1.

15. Литература

15.1. Нормативные документы

[Private-Enterprise-Numbers] IANA, "Private Enterprise Numbers", <https://www.iana.org/assignments/enterprise-numbers/>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC3688] Mealling, M., "The IETF XML Registry", BCP 81, RFC 3688, DOI 10.17487/RFC3688, January 2004, <https://www.rfc-editor.org/info/rfc3688>.

[RFC5646] Phillips, A., Ed. and M. Davis, Ed., "Tags for Identifying Languages", BCP 47, RFC 5646, DOI 10.17487/RFC5646, September 2009, <https://www.rfc-editor.org/info/rfc5646>.

[RFC6020] Bjorklund, M., Ed., "YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)", RFC 6020, DOI 10.17487/RFC6020, October 2010, <https://www.rfc-editor.org/info/rfc6020>.

[RFC6991] Schoenwaelder, J., Ed., "Common YANG Data Types", RFC 6991, DOI 10.17487/RFC6991, July 2013, <https://www.rfc-editor.org/info/rfc6991>.

[RFC7252] Shelby, Z., Hartke, K., and C. Bormann, "The Constrained Application Protocol (CoAP)", RFC 7252, DOI 10.17487/RFC7252, June 2014, <https://www.rfc-editor.org/info/rfc7252>.

[RFC7641] Hartke, K., "Observing Resources in the Constrained Application Protocol (CoAP)", RFC 7641, DOI 10.17487/RFC7641, September 2015, <https://www.rfc-editor.org/info/rfc7641>.

[RFC7950] Bjorklund, M., Ed., "The YANG 1.1 Data Modeling Language", RFC 7950, DOI 10.17487/RFC7950, August 2016, <https://www.rfc-editor.org/info/rfc7950>.

[RFC7959] Bormann, C. and Z. Shelby, Ed., "Block-Wise Transfers in the Constrained Application Protocol (CoAP)", RFC 7959, DOI 10.17487/RFC7959, August 2016, <https://www.rfc-editor.org/info/rfc7959>.

[RFC7970] Danyliw, R., "The Incident Object Description Exchange Format Version 2", RFC 7970, DOI 10.17487/RFC7970, November 2016, <https://www.rfc-editor.org/info/rfc7970>.

[RFC8040] Bierman, A., Bjorklund, M., and K. Watsen, "RESTCONF Protocol", RFC 8040, DOI 10.17487/RFC8040, January 2017, <https://www.rfc-editor.org/info/rfc8040>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC8345] Clemm, A., Medved, J., Varga, R., Bahadur, N., Ananthakrishnan, H., and X. Liu, "A YANG Data Model for Network Topologies", RFC 8345, DOI 10.17487/RFC8345, March 2018, <https://www.rfc-editor.org/info/rfc8345>.

[RFC8783] Boucadair, M., Ed. and T. Reddy.K, Ed., "Distributed Denial-of-Service Open Threat Signaling (DOTS) Data Channel Specification", RFC 8783, DOI 10.17487/RFC8783, May 2020, <https://www.rfc-editor.org/info/rfc8783>.

[RFC8791] Bierman, A., Björklund, M., and K. Watsen, "YANG Data Structure Extensions", RFC 8791, DOI 10.17487/RFC8791, June 2020, <https://www.rfc-editor.org/info/rfc8791>.

[RFC8949] Bormann, C. and P. Hoffman, "Concise Binary Object Representation (CBOR)", STD 94, RFC 8949, DOI 10.17487/RFC8949, December 2020, <https://www.rfc-editor.org/info/rfc8949>.

[RFC9132] Boucadair, M., Ed., Shallow, J., and T. Reddy.K, "Distributed Denial-of-Service Open Threat Signaling (DOTS) Signal Channel Specification", RFC 9132, DOI 10.17487/RFC9132, September 2021, <https://www.rfc-editor.org/info/rfc9132>.

15.2. Дополнительная литература

[Cause] IANA, "DOTS Signal Channel Conflict Cause Codes", <https://www.iana.org/assignments/dots/>.

[DOTS-Multihoming] Boucadair, M., Reddy.K, T., and W. Pan, "Multi-homing Deployment Considerations for Distributed-Denial-of-Service Open Threat Signaling (DOTS)", Work in Progress, Internet-Draft, draft-ietf-dots-multihoming-13, 26 April 2022, <https://datatracker.ietf.org/doc/html/draft-ietf-dots-multihoming-13>.

[DOTS-Robust-Blocks] Boucadair, M. and J. Shallow, "Distributed Denial-of-Service Open Threat Signaling (DOTS) Signal Channel Configuration Attributes for Robust Block Transmission", Work in Progress, Internet-Draft, draft-ietf-dots-robust-blocks-03, 11 February 2022, <https://datatracker.ietf.org/doc/html/draft-ietf-dots-robust-blocks-03>.

[DOTS-Telemetry-Specs] Doron, E., Reddy, T., Andreasen, F., Xia, L., and K. Nishizuka, "Distributed Denial-of-Service Open Threat Signaling (DOTS) Telemetry Specifications", Work in Progress, Internet-Draft, draft-doron-dots-telemetry-00, 30 October 2016, <https://datatracker.ietf.org/doc/html/draft-doron-dots-telemetry-00>.

[Key-Map] IANA, "DOTS Signal Channel CBOR Key Values", <https://www.iana.org/assignments/dots/>.

[PYANG] "pyang", commit dad5c68, April 2022, <https://github.com/mbj4668/pyang>.

[RFC2330] Paxson, V., Almes, G., Mahdavi, J., and M. Mathis, "Framework for IP Performance Metrics", RFC 2330, DOI 10.17487/RFC2330, May 1998, <https://www.rfc-editor.org/info/rfc2330>.

[RFC4732] Handley, M., Ed., Rescorla, E., Ed., and IAB, "Internet Denial-of-Service Considerations", RFC 4732, DOI 10.17487/RFC4732, December 2006, <https://www.rfc-editor.org/info/rfc4732>.

[RFC5612] Eronen, P. and D. Harrington, "Enterprise Number for Documentation Use", RFC 5612, DOI 10.17487/RFC5612, August 2009, <https://www.rfc-editor.org/info/rfc5612>.

[RFC8340] Bjorklund, M. and L. Berger, Ed., "YANG Tree Diagrams", BCP 215, RFC 8340, DOI 10.17487/RFC8340, March 2018, <https://www.rfc-editor.org/info/rfc8340>.

[RFC8525] Bierman, A., Bjorklund, M., Schoenwaelder, J., Watsen, K., and R. Wilton, "YANG Library", RFC 8525, DOI 10.17487/RFC8525, March 2019, <https://www.rfc-editor.org/info/rfc8525>.

[RFC8612] Mortensen, A., Reddy, T., and R. Moskowitz, "DDoS Open Threat Signaling (DOTS) Requirements", RFC 8612, DOI 10.17487/RFC8612, May 2019, <https://www.rfc-editor.org/info/rfc8612>.

[RFC8811] Mortensen, A., Ed., Reddy.K, T., Ed., Andreasen, F., Teague, N., and R. Compton, "DDoS Open Threat Signaling (DOTS) Architecture", RFC 8811, DOI 10.17487/RFC8811, August 2020, <https://www.rfc-editor.org/info/rfc8811>.

[RFC8903] Dobbins, R., Migault, D., Moskowitz, R., Teague, N., Xia, L., and K. Nishizuka, "Use Cases for DDoS Open Threat Signaling", RFC 8903, DOI 10.17487/RFC8903, May 2021, <https://www.rfc-editor.org/info/rfc8903>.

[RFC9133] Nishizuka, K., Boucadair, M., Reddy.K, T., and T. Nagata, "Controlling Filtering Rules Using Distributed Denial-of-Service Open Threat Signaling (DOTS) Signal Channel", RFC 9133, DOI 10.17487/RFC9133, September 2021, <https://www.rfc-editor.org/info/rfc9133>.

[RFC9177] Boucadair, M. and J. Shallow, "Constrained Application Protocol (CoAP) Block-Wise Transfer Options Supporting Robust Transmission", RFC 9177, DOI 10.17487/RFC9177, March 2022, <https://www.rfc-editor.org/info/rfc9177>.

[RFC9260] Stewart, R., Tüxen, M., and K. Nielsen, "Stream Control Transmission Protocol", RFC 9260, DOI 10.17487/RFC9260, June 2022, <https://www.rfc-editor.org/info/rfc9260>.

Благодарности

Авторы хотели бы поблагодарить Flemming Andreasen, Liang Xia и Kaname Nishizuka, соавторов [DOTS-Telemetry-Specs], а также всех, кто внёс вклад в этот документ.

Спасибо Kaname Nishizuka, Yuhei Hayashi и Tom Petch за их замечания и рецензии.

Особая благодарность Jon Shallow и Kaname Nishizuka за из работу по реализации и взаимодуйствию.

Большое спасибо Jan Lindblad за обзор yangdoctors, Nagendra Nainar за обзор opsdir, James Gruessing за обзор artart, Michael Scharf за обзор tsv-art, Ted Lemon за обзор int-dir и Robert Sparks за обзор gen-art.

Спасибо Benjamin Kaduk за подробную рецензию AD.

Спасибо Roman Danyliw, Éric Vyncke, Francesca Palombini, Warren Kumari, Erik Kline, Lars Eggert, Robert Wilton за рецензии IESG.

Участники работы

Li Su
CMCC
Email: suli@chinamobile.com

Pan Wei

Huawei
Email: william.panwei@huawei.com

Адреса авторов

Mohamed Boucadair (editor)
Orange
35000 Rennes
France
Email: mohamed.boucadair@orange.com

Tirumaleswar Reddy.K (editor)
Akamai
Embassy Golf Link Business Park
Bangalore 560071
Karnataka
India
Email: kondtir@gmail.com

Ehud Doron
Radware Ltd.
Raoul Wallenberg Street
Tel-Aviv 69710
Israel
Email: ehudd@radware.com

Meiling Chen
CMCC
32 Xuanwumen West Street
Beijing
100053
China
Email: chenmeiling@chinamobile.com

Jon Shallow
United Kingdom
Email: supjps-ietf@jpshallow.com

Перевод на русский язык

Николай Малых

nmalykh@protokols.ru