Internet Engineering Task Force (IETF)
Request for Comments: 8520
Category: Standards Track
ISSN: 2070-1721

E. Lear
Cisco Systems
R. Droms
Google
D. Romascanu

March 2019

Manufacturer Usage Description Specification

Спецификация описаний применения от изготовителя

Аннотация

Этот документ задаёт компонентную архитектуру описаний применения от изготовителя (Manufacturer Usage Description или MUD). Цель MUD заключается в предоставлении конечным устройствам возможности сообщать сети о том, какой тип доступа и сетевые функции нужны им для нормальной работы. Исходно основное внимание уделено контроль доступа, а в последующих работах могут быть рассмотрены другие аспекты.

Документ задаёт два модуля YANG, опции IPv4 и IPv6 DHCP, LLDP[1] TLV, URL и расширение сертификатов X.509, а также средства для подписания и проверки описаний.

Статус документа

Документ относится к категории Internet Standards Track.

Документ является результатом работы IETF[2] и представляет согласованный взгляд сообщества IETF. Документ прошёл открытое обсуждение и был одобрен для публикации IESG[3]. 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/rfc8520.

Оглавление

1. Введение....2
1.1. Что MUD не делает....3
1.2. Простой пример....3
1.3. Терминология....3
1.4. Задание предполагаемого использования....3
1.5. Нахождение правил - MUD URL....3
1.6. Обработка MUD URL....4
1.7. Типы политики....4
1.8. Архитектура MUD....5
1.9. Порядок операций....6
2. Модель MUD и семантическое значение....6
2.1. Модуль YANG IETF-MUD....6
3. Определения модели MUD для корневого контейнера mud....7
3.1. mud-version....7
3.2. MUD URL....7
3.3. Контейнеры to-device-policy и from-device-policy....7
3.4. last-update....7
3.5. cache-validity....7
3.6. is-supported....7
3.7. systeminfo....7
3.8. mfg-name, software-rev, model-name, firmware-rev....7
3.9. Расширения....7
4. Дополнение модели ACL....8
4.1. manufacturer....8
4.2. same-manufacturer....8
4.3. documentation....8
4.4. model....8

[1] Link Layer Discovery Protocol - протокол обнаружения на канальном уровне.
[2] Internet Engineering Task Force - комиссия по решению инженерных задач Internet.
[3] Internet Engineering Steering Group - комиссия по инженерным разработкам Internet.

4.5. local-networks....8
4.6. controller....8
4.7. my-controller....8
4.8. direction-initiated....8
5. Обработка файла MUD....8
6. Как выглядит MUD URL?....9
7. Модель YANG MUD....9
8. Расширение для доменных имён в модели ACL....12
8.1. src-dnsname....12
8.2. dst-dnsname....12
8.3. Модель ietf-acldns....13
9. Пример файла MUD....13
10. Опции DHCP MUD URL....15
10.1. Поведение клиента....15
10.2. Поведение сервера....15
10.3. Требования к ретранслятору....15
11. Расширение MUD URL X.509....15
12. Расширение MUD LLDP....17
13. Создание и обработка подписанных файлов MUD....17
13.1. Создание подписи файла MUD....17
13.2. Проверка подписи файла MUD....17
14. Расширяемость....18
15. Вопросы внедрения....18
16. Вопросы безопасности....18
17. Взаимодействия с IANA....19
17.1. Регистрация модулей YANG....19
17.2. Регистрация URI....19
17.3. Опции DHCPv4 и DHCPv6....19
17.4. Расширения PKIX....19
17.5. Регистрация типов носителей для файлов MUD....20
17.6. Реестр IANA для субтипов LLDP TLV....20
17.7. Общеизвестное имя MUD URN....20
17.8. Реестр расширений....20
18. Литература....21
18.1. Нормативные документы....21
18.2. Дополнительная литература....22
Приложение A. Принятые по умолчанию узлы MUD....22
Приложение B. Пример расширения DETNET-indicator....25
Благодарности....27
Адреса авторов....27

1. Введение

Сеть Internet создавалась в основном для компьютеров общего назначения, устройств, применяемых в соответствии с целями их владельцев. В [RFC1984] предполагалось, что конечные устройства способны защищать себя самостоятельно. Это имело смысл, когда типичными устройствами были рабочие станции и мэйнфреймы, и продолжает быть осмысленным для современных вычислительных устройств общего назначения, включая ноутбуки, смартфоны и планшеты.

В [RFC7452] рассмотрены шаблоны устройства интеллектуальных объектов и связанные с этим вопросы. Затем появились объекты, не предназначенные специально для вычислительных задач общего назначения. Эти устройства, называемые в документе Вещами (Things), имеют конкретное назначение, поэтому иные варианты их применения не рассматриваются. Если из небольшого числа вариантов применения следует небольшое число шаблонов взаимодействия (коммуникаций), эти два утверждения можно переформулировать как описание использования от изготовителя (MUD), которое может применяться в разных точках сети. MUD относится в первую очередь к угрозам для устройства, а не исходящим от него угрозам. Однако в некоторых обстоятельствах MUD может обеспечивать некоторую защиту в последнем случае в зависимости от способа передачи MUD URL и аутентификации устройств и взаимодействий между ними.

В этом контексте термин «изготовитель» (manufacturer) используется достаточно вольно для обозначения субъекта или организации, определяющих варианты использования устройства. Например, для лампочек это может быть их производитель, для интеллектуальных устройств со стеком Linux - интегратор устройств. Важно, что предполагается ограниченное применение самого устройства и наличие в цепочке поставок организации, отвечающей за информировании сети о назначении устройства.

Назначение MUD указано ниже.

- Существенное сокращение фронта угроз на устройстве до уровня предусмотренного изготовителем.
- Предоставление средств расширения политики с учётом роста числа типов устройств в сети.
- Предоставление способа устранять хотя бы часть уязвимостей быстрее, чем происходит обновление системы. Это особенно важно для систем, поддержка которых прекращена.
- Минимизация расходов на внедрение системы.
- Предоставление изготовителям средств расширения для представления возможностей и требований к устройства.

MUD состоит за 3 архитектурных блоков:

- дескриптор URL, по которому можно найти описание;

- само описание, включая его интерпретацию;
- способы извлечения описания локальными системами управления сетью.

Наибольшая эффективность MUD обеспечивается, когда сеть способна тем или иным способом идентифицировать удалённые точки, с которыми будут взаимодействовать Вещи (Things).

В этой спецификации описан каждый из указанных выше архитектурных блоков и способы их предполагаемого совместного использования. Однако блоки могут применяться и по отдельности, независимо данной спецификации, в соответствии с целями локальной системы.

1.1. Что MUD не делает

Описание MUD не предназначено для сетевого контроля полномочий компьютеров общего назначения, поскольку их изготовители не могут предусмотреть конкретную картину взаимодействий для описания. Кроме того, даже устройства с одним или небольшим числом вариантов применения могут иметь существенно различающиеся картины взаимодействий и MUD для них не подходит.

Хотя MUD может предоставлять сетевым администраторам некоторую дополнительную защиту при наличии в устройстве уязвимостей, это не заменяет для производителей необходимости устранять уязвимости.

Что бы ни указывал производитель в файле MUD, это будут рекомендации, а не директивы. Их применение на местах будет зависеть от многих факторов и в конечном итоге администратор сети должен решить, что будет уместно в конкретных обстоятельствах.

1.2. Простой пример

Лампочка предназначены для освещения помещения и может управляться дистанционно через сеть, а также использовать службу rendezvous (доступ к которой может обеспечиваться через приложение на смартфоне). Можно сказать что для этой лампочки нежелательны все иные варианты доступа к ней через сеть. Она не будет взаимодействовать со службой новостей, разговаривать с холодильником и ей не нужен принтер или иные устройства. У лампочки нет друзей в социальных сетях. Поэтому применение списка доступа, в котором указано, что лампочка может подключаться только к службе rendezvous, не помешает выполнению основных функций и в то же время позволит сети обеспечивать лампочке и другим устройствам дополнительный уровень защиты.

1.3. Терминология

MUD
: Описание применения от изготовителя

MUD file - файл MUD
: Файл, содержащий JSON-представление на основе YANG, который описывает Вещь (Thing) и связанное с ней предполагаемое поведение сети.

MUD file server - сервер файлов MUD
: Сервер web, где хранится файл MUD.

MUD manager - диспетчер MUD
: Система, которая запрашивает и получает файл MUD от сервера MUD. После обработки файла MUD эта система может направлять изменения в соответствующие элементы сети.

MUD controller - контроллер MUD
: Синоним для диспетчера MUD.

MUD URL
: Дескриптор URL, который может использоваться диспетчером MUD для получения файла MUD.

Thing - Вещь
: Устройство, выдающее MUD URL.

Manufacturer - изготовитель
: Организация, которая настраивает Вещь (Thing) для выдачи MUD URL и задаёт рекомендации в файле MUD. Изготовителем не обязательно является организация, создавшая Вещь (Thing). Это может быть, например, системный интегратор и даже поставщик компонентов.

Ключевые слова **необходимо** (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.

1.4. Задание предполагаемого использования

Понятие целевого использования само по себе не ново. Администраторы сетей постоянно применяют списки доступа, чтобы разрешать лишь предусмотренное поведение. Понятие «белого списка» хорошо описано Chapman и Zwicky в [FW95]. Системы профилирования, применяющие эвристику для идентификации типов систем тоже существуют много лет.

Вещь (Thing) может легко сообщить сети требуемый тип доступа, не вдаваясь в детали своего типа. По сути, это было бы оборотной стороной [RFC7488]. Однако в поисках общего решения предполагается, что устройство реализует функциональность, требуемую для его ограниченных целей. Это базовое экономическое ограничение. Пока сеть не отвергает доступ к такому устройству, его разработчикам не нужно предоставлять сети какие-либо сведения. На сегодняшний это утверждение остаётся верным.

1.5. Нахождение правил - MUD URL

Работа начинается с выдачи устройством универсального локатора ресурса (URL) [RFC3986], который служит для классификации типа устройства и определения местоположения файла правил (политики). В MUD URL должна применяться схема https [RFC7230].

Этот документ определяет 3 способа выдачи MUD URL, перечисленных ниже.

- Опция DHCP [RFC2131] [RFC8415], которую клиент DHCP использует для информирования сервера DHCP. Сервер может предпринять дальнейшие действия, например, выступив в роли диспетчера MUD или передав MUD URL диспетчеру MUD.
- Ограничения X.509. В IEEE разработан стандарт IEEE 802.1AR [IEEE8021AR] для выбора подхода к передаче характеристик устройства на основе сертификата. Этот стандарт основан на [RFC5280]. Расширение MUD URL является некритическим, как того требует IEEE 802.1AR. Для передачи сертификата могут применяться различные средства, включая расширяемый туннельный протокол аутентификации (Tunnel Extensible Authentication Protocol или TEAP) [RFC7170].
- Кадр протокола обнаружения канального уровня (Link Layer Discovery Protocol или LLDP), определённого в [IEEE8021AB].

Возможны и другие способы получения MUD URL сетью. Некоторые устройства могут уже быть введены в эксплуатацию или иметь очень ограниченные возможности передачи MUD URL, но при этом могут быть идентифицированы тем или иным способом, например, по серийному номеру или открытому ключу. В таких случаях изготовители могут сопоставлять эти идентификаторы с конкретными MUD URL (или даже с файлами). В ситуациях с отсутствием или ограниченностью доступа в Internet могут быть доступны альтернативные механизмы распознавания, которые не описываются в этом документе. Разработчикам рекомендуется обеспечивать гибкость при выборе способа получения MUD URL.

1.6. Обработка MUD URL

Диспетчерам MUD, имеющим такую возможность, **следует** извлекать MUD URL и файлы подписи в соответствии с [RFC7230], используя метод GET [RFC7231]. Они **должны** проверять сертификаты по правилам параграфа 3.1 из [RFC2818].

В запросы для MUD URL **следует** включать поля заголовков Accept ([RFC7231], параграф 5.3.2) с полем application/mud+json, Accept-Language ([RFC7231], параграф 5.3.5) и User-Agent ([RFC7231], параграф 5.5.3).

Диспетчерам MUD **следует** автоматически обрабатывать отклики с кодами состояния 3xx.

Если диспетчер MUD не способен извлечь MUD URL, **можно** применять другие средства импорта файлов MUD и связанных с ними файлов подписей. Файлы, подписи которых могут быть подтверждены, допускается использовать. В таких средах контроллерам **следует** предупреждать администраторов о приближении конца срока действия cache-validity, чтобы они могли проверить наличие новых файлов.

У диспетчера MUD может не быть возможности извлечь файл MUD в данный момент. В таких случаях диспетчеру **следует** считать безопасным использование имеющегося файла в течение какого-то времени. По истечении некоторого времени **следует** занести запись о невозможности извлечения файла в системный журнал. Для таких отказов могут быть очень веские причины, включая возможность отключения (offline) диспетчера MUD или отказ локального соединения с Internet. Возможно также вмешательство злоумышленника в процесс развёртывания устройства. Поведение в таких ситуациях определяется местными условиями.

1.7. Типы политики

После распознания MUD URL диспетчер MUD извлекает файл, описывающий коммуникационные взаимодействия, предусмотренные для устройства. Изготовитель может задать конкретные хосты для облачных служб или определённые классы доступа в работающей сети. Примером такого класса могут быть устройства определённого типа, который задан изготовителем и указан компонентом полномочий (например, именем домена) в MUD URL. Другим примером может служить разрешение или запрет локального доступа. Как и для других правил, здесь возможны комбинации, например,

- разрешается доступ к устройствам того же изготовителя;
- разрешается доступ к контроллерам и от них по протоколу приложений с ограничениями (Constrained Application Protocol или COAP) [RFC7252];
- разрешается доступ к локальному DNS/NTP;
- запрещаются все прочие варианты доступа.

Принтер, например, может иметь описание, в котором:

- разрешён доступ к порту IPP или LPD;
- разрешён локальный доступ к порту HTTP;
- запрещены все иные варианты доступа.

В этом случае каждый может печатать на принтере, но для работы с интерфейсом управления нужен локальный доступ.

Извлекаемые файлы должны быть хорошо согласованы с сетевой инфраструктурой, чтобы их было легко развернуть. Язык YANG [RFC7950] выбран потому, что он обеспечивает точные и адекватные модели для использования сетевыми устройствами. Для сериализации применяется формат JSON [RFC8259], являющийся более компактным и удобочитаемым, нежели XML. В последующих версиях MUD могут быть выбраны другие форматы.

Хотя приведённые здесь примеры правил ориентированы на контроль доступа, это не является единственным направлением. Структурируя описанную в документе модель и чёткими точками расширения, можно включить и другие описания. Одним из таких описаний является качество обслуживания.

Заданные здесь модули YANG являются расширениями [RFC8519]. Расширения этой модели позволяют изготовителю выразить классы систем, которые он считает требуемыми для корректной работы устройства. В документе заданы два

модуля. Первый указывает средства использования доменных имён в списках контроля доступа (Access Control List или ACL), чтобы для устройств, имеющих контроллеры в облаке, можно было должным образом проверить полномочия по доменным именам в случаях, когда сопоставление этих имён с адресами может быстро меняться.

Во втором модуле адреса IP абстрагируются в некие классы, которые связываются с реальными IP-адресами в процессе локальной обработки. С помощью этих классов изготовители могут задать для устройств способы коммуникаций, чтобы элементы сети можно было настраивать локальным системам, знающим местную топологию. Таким образом, при внедрении заполняются классы, заданные изготовителем. Перечисленные ниже абстракции могут сопоставляться с хостами.

Manufacturer - изготовитель
Устройство конкретного изготовителя, как указано полномочным компонентом MUD URL.

Same-manufacturer - тот же изготовитель
Устройства с тем же компонентом authority в MUD URL.

controller - контроллер
Устройства, отнесённые локальным администратором сети к определённому классу.

my-controller - мой контроллер
Устройства, предназначенные быть контроллерами для MUD URL, выданного Вещью (Thing).

local - локальный
Класс адресов IP, находящихся внутри некой административной границы (по умолчанию в локальной подсети).

Классы manufacturer могут быть легко заданы изготовителем, а классы controller по замыслу задаются администратором.

Поскольку изготовители не знают, кто будет использовать их устройства, важно, чтобы функциональность, указываемая в описании применения, была достаточно естественной и однозначной. Поэтому конфигурация на основе YANG в файле MUD ограничивается модулями, заданными или упоминаемыми в этом документе или в документированных расширениях.

1.8. Архитектура MUD

Эти компоненты являются основой для архитектуры, как показано на рисунке 1.

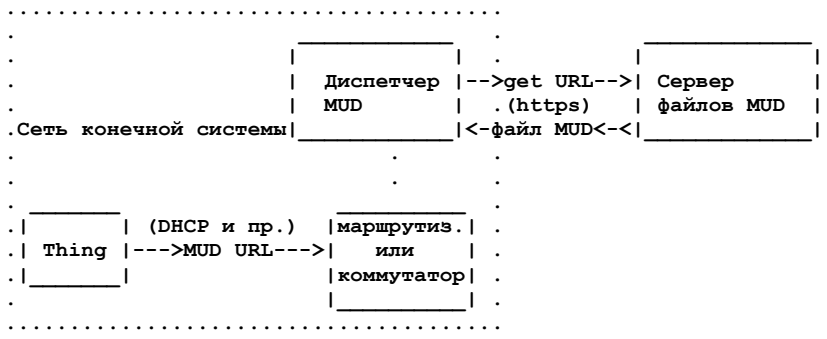

Рисунок 1. Архитектура MUD.

На приведённой выше схеме коммутатор или маршрутизатор собирает MUD URL и пересылает их диспетчеру MUD (системе управления сетью) для обработки. Это выполняется по-разному в зависимости от способа передачи URL. Например, при использовании DHCP сервер DHCP может получать и обрабатывать URL. В случае IEEE 802.1X [IEEE8021X] коммутатор может передавать URL в сертификате серверу проверки подлинности по расширяемому протоколу аутентификации (Extensible Authentication Protocol или EAP) через Radius [RFC3748] для последующей обработки. Одним из методов является протокол TEAP, описанный в [RFC7170]. Расширение сертификата описано ниже.

Информация, возвращаемая сервером файлов MUD, действительна, пока Вещь (Thing) остаётся подключённой. Завершение срока действия не предусмотрено, однако при обнаружении менеджером MUD изменения файла MUD для Thing ему **следует** оперативно обновить правила, полностью учитывая поток одобрения, требуемый при внедрении. Это позволяет своевременно обрабатывать новые рекомендации от изготовителя.

Возвращаемая сервером файлов MUD (web-сервер) действительна в течение всего срока соединения с Thing или задаётся описанием. Таким образом при отключении Thing все связанные с Вещью настройки можно удалить из коммутатора. Описание может также время от времени меняться при появлении новых возможностей или шаблонов взаимодействия, а также при обнаружении новых уязвимостей.

Сервер web обычно работает у изготовителя или от его имени в ином месте. Его доменное имя соответствует органу (authority), указанному в MUD URL. Для унаследованных вариантов, где Things не может выпускать URL, коммутатор может служить прокси, если он может определить подходящий локатор URL. В тривиальном случае коммутатор может жёстко задать (hardcode) MUD URL на своём порту или организовать сопоставление MUD URL с неким доступным идентификатором, таким как адрес L2 или хэш сертификата.

Диспетчер MUD в такой среде выполняет указанные ниже действия:

- приём MUD URL;
- извлечение файлов MUD;
- трансляция абстракций из файлов MUD в конкретную конфигурацию элемента сети;
- поддержка и обновление всех требуемых сопоставления с абстракциями;
- обновление конфигурации элементов сети.

Диспетчер MUD может быть компонентом системы проверки подлинности и полномочий, а также учёта (Authentication, Authorization, and Accounting или AAA) или системы управления сетью. Коммуникации внутри таких систем и между ними и элементами сети выходят за рамки этого документа.

1.9. Порядок операций

Как отмечено выше, MUD состоит из архитектурных блоков и порядок операций может меняться. Ниже приведён пример возможного порядка операций.

1. Вещь выдаёт URL.
2. URL пересылается диспетчеру MUD ближайшим коммутатором (это зависит от способа выдачи MUD URL).
3. Диспетчер MUD извлекает файл MUD и подпись с файлового сервера MUD, если у него их ещё нет. После проверки подписи диспетчер может проверить URL на сервере web или в службе репутации домена, а также проверить все указанные в файле хосты в службе репутации, если сочтёт нужным.
4. Диспетчер MUD может запросить у администратора разрешение добавить Вещь и связанные с ней правила. Если Вещь или её тип известны, этот шаг можно пропустить.
5. Диспетчер MUD создаёт локальную конфигурацию на основе заданных в этом документе абстракций.
6. Диспетчер MUD настраивает ближаший к Вещи коммутатор и может настроить другие системы.
7. При отсоединении Вещи правила (политика) удаляются.

2. Модель MUD и семантическое значение

Файл MUD содержит экземпляр модель YANG в представлении JSON [RFC7951]. Для целей MUD узлами, которые могут быть изменены, являются списки доступа, добавленные этой моделью. В файл MUD можно помещать представление лишь нескольких схем YANG:

- ietf-access-control-list [RFC8519];
- ietf-mud (RFC 8520);
- ietf-acldns (RFC 8520).

Для добавления схем могут применяться расширения, как описано ниже.

Для обеспечения возможности широчайшего внедрения издателям файлов MUD **следует** использовать абстракции из этого документа и избегать указания адресов IP. Диспетчеру MUD **не следует** автоматически реализовать файлы MUD с адресами IP, особенно теми, которые могут иметь локальную значимость. Адресация стороны из списка доступа является неявной в зависимости от применения как to-device-policy или from-device-policy.

Издателям файлов MUD **следует** ограничиваться применением модели ACL для узлов-листьев, указанных в этой спецификации, за исключением имени (name) ACL, типа (type), имени (name) записи контроля доступа (Access Control Entry или ACE) и сведений о портах TCP и UDP у получателя и отправителя. При отсутствии расширений предполагается, что в файлах MUD реализованы лишь свойства модели ACL match-on-ipv4, match-on-ipv6, match-on-tcp, match-on-udp, match-on-icmp.

Кроме того, **следует** включать только действия accept или drop. Диспетчер MUD **может** интерпретировать reject как drop, все прочие действия ему **следует** игнорировать. Это обусловлено тем, что изготовитель не знает точно контекста локального внедрения, чтобы понять, уместно ли действие reject. Решение следует предоставить администратору сети.

С учётом того, что MUD не работает с интерфейсами, поддержка модуля ietf-interfaces [RFC8343] не требуется. В частности, не требуется поддержка связанных с интерфейсами функций и ветвей (например, interface-attachment и interface-stats) модуля YANG ACL.

Фактически, диспетчеры MUD **могут** игнорировать любой компонент описания и описание целиком и им **следует** тщательно проверять все описания MUD. Издателям файлов MUD **недопустимо** включать узлы, не описанные в параграфе 3.9.

2.1. Модуль YANG IETF-MUD

Модуль состоит из трёх частей:

- контейнер mud со сведениями, относящимися к извлечению и проверке самого файла MUD, а также правил, нацеленных на Вещь и от неё;
- второй компонент дополняет контейнер сопоставления модели ACL для добавления нескольких узлов, относящихся к MUD URL или абстрагированных для использования в локальном окружении;
- третий компонент дополняет контейнер tcp-acl модели ACL для добавления сопоставления по направлению организации соединения TCP.

Действительный файл MUD содержит два корневых объекта - контейнеры mud и acls. При необходимости можно добавить корневые объекты с помощью расширений. Напомним, что при анализе acls элементы из блока match объединяются логической операцией И (AND). В общем случае следует применять одну абстракцию в операторе match. Например, не имеет смысла сопоставлять с аргументом my-controller и controller, поскольку вероятность совпадения их значений очень мала.

В документе применяется простое графическое представление моделей данных в соответствии с [RFC8340].

```
module: ietf-mud
  +--rw mud!
     +--rw mud-version           uint8
     +--rw mud-url               inet:uri
     +--rw last-update           yang:date-and-time
     +--rw mud-signature?        inet:uri
     +--rw cache-validity?       uint8
     +--rw is-supported          boolean
```

```
      +--rw systeminfo?           string
      +--rw mfg-name?             string
      +--rw model-name?           string
      +--rw firmware-rev?         string
      +--rw software-rev?         string
      +--rw documentation?        inet:uri
      +--rw extensions*           string
      +--rw from-device-policy
      |  +--rw acls
      |     +--rw access-list* [name]
      |        +--rw name    -> /acl:acls/acl/name
      +--rw to-device-policy
         +--rw acls
            +--rw access-list* [name]
               +--rw name    -> /acl:acls/acl/name

augment /acl:acls/acl:acl/acl:aces/acl:ace/acl:matches:
  +--rw mud
     +--rw manufacturer?        inet:host
     +--rw same-manufacturer?   empty
     +--rw model?               inet:uri
     +--rw local-networks?      empty
     +--rw controller?          inet:uri
     +--rw my-controller?       empty
augment
  /acl:acls/acl:acl/acl:aces/acl:ace/acl:matches
     /acl:l4/acl:tcp/acl:tcp:
  +--rw direction-initiated?   direction
```

3. Определения модели MUD для корневого контейнера mud

3.1. mud-version

Этот узел задаёт целочисленный номер версии спецификации MUD. Данный документ соответствует версии 1.

3.2. MUD URL

URL для указания файла MUD. Это полезно в случаях, когда файл и связанная с ним подпись загружаются вручную, например, в автономном режиме (offline).

3.3. Контейнеры to-device-policy и from-device-policy

Списки доступа описаны в [RFC8519]. В случае MUD файл MUD должен явно описывать взаимодействие с Вещью, включая указание того, что должно быть разрешено или запрещено в каждом направлении взаимодействия. Следовательно, каждый из этих контейнеров указывает соответствующее направление потока применительно к конкретной Вещи. Контейнеры содержат ссылки на конкретные списки доступа.

3.4. last-update

Значение date-and-time для момента создания файла MUD. Оно похоже на номер версии и имеет форму из [RFC6991].

3.5. cache-validity

Это значение uint8 указывает интервал в времени (в часах), который станция сетевого управления **должна** выждать между последним извлечением данных и проверкой наличия обновлений. Для любой поддерживаемой Вещи **рекомендуется** устанавливать значение не менее 24 и **недопустимы** значения больше 168. **Следует** задавать интервал не короче интервалов, определяемых директивами кэширования HTTP (например, cache-control или Expires). Важно подчеркнуть, что по истечении этого времени диспетчер MUD не обязан отбрасывать файл MUD или прерывать доступ к Вещи. Более подробные сведения представлены в разделе 16.

3.6. is-supported

Это логическое значение от изготовителя указывает администратору сети, поддерживается ли данная вещь. В этом контексте Вещь считается не поддерживаемой, если изготовитель предполагает не выпускать для неё обновлений микрокода или программ и не обновлять файл MUD. Диспетчер MUD всё равно **может** проверять обновления.

3.7. systeminfo

Текстовое (UTF-8) описание подключаемой Вещи, предназначенное для информирования администраторов. **Не следует** использовать более 60 символов.

3.8. mfg-name, software-rev, model-name, firmware-rev

Необязательные поля в соответствии с [RFC8348]. Поля firmware-rev и software-rev **недопустимо** указывать в файле MUD, если устройство может обновляться, но MUD URL не может (например, MUD URL с сертификатами 802.1AR).

3.9. Расширения

Необязательный лист-список с именами расширений MUD, применяемых в файле MUD. Отметим, что расширения MUD **недопустимо** применять в файле MUD без их объявления. Реализации **должны** игнорировать в файле любые непонятные узлы. Расширения могут применяться, как описано выше, или ссылаться на другие работы. Пример расширения приведён в Приложении B.

4. Дополнение модели ACL

Термин «сопоставление» (match) в этом разделе относится к узлу matches модели ACL.

4.1. manufacturer

Этот узел содержит имя хоста, которое будет сопоставляться с компонентом authority в MUD URL другой Вещи. В простейшей форме узлы manufacturer и same-manufacturer могут быть реализованы как списки доступа, в более сложных могут применяться дополнительные возможности сети. Например, если в manufacturer указано flobbity.example.com[1], все Вещи, зарегистрированные с MUD URL, где содержится flobbity.example.com в разделе authority, будут соответствовать.

4.2. same-manufacturer

Этот пустой (null-valued) узел является эквивалентом применения элемента manufacturer для индикации того, что authority в MUD URL другой Вещи совпадает с authority в MUD URL данной Вещи. Например, если MUD URL Вещи имеет значение https://b1.example.com/ThingV1, все устройства, имеющие MUD URL с разделом authority b1.example.com, будут соответствовать.

4.3. documentation

Этот идентификатор URI содержит URL для документации к устройству и файлу MUD. Это может быть особенно полезно при использовании класса controller, чтобы объяснить его применение.

4.4. model

Эта строка сопоставляется с полным MUD URL, охватывая модель, уникальную в контексте данного authority. Строка может включать не только сведения о модели, но и данные о версии и другие сведения, которые изготовитель сочтёт нужными. Строка предназначена для того, чтобы разрешить или запретить взаимодействия между устройствами с точно совпадающим классом.

4.5. local-networks

Этот пустой (null-valued) узел преобразуется для включения локальных сетей. Приятое по умолчанию преобразование служит для того, чтобы для пакетов не применялся маршрут по умолчанию, полученный от маршрутизатора. Однако администраторы могут преобразовать это выражение в соответствии с потребностями своё сети.

4.6. controller

Этот идентификатор URI задаёт значение, которое контроллер будет регистрировать в диспетчере MUD. Затем этот узел преобразуется в набор зарегистрированных таким способом устройств. Узел может быть также URN, описывающим в этом случае стандартизованную общеизвестную службу, например DNS или NTP (см. параграф 17.7).

При использовании my-controller администратору может быть предложено заполнять этот класс для каждой модели. Использование controller с именованным классом позволяет пользователю заполнять этот класс однократно для множества разных моделей, которые может выпускать изготовитель.

URI контроллера **могут** иметь форму URL (например, http[s]://), однако менеджерам MUD **недопустимо** преобразовывать (resolve) ссылки и извлекать такие файлы, а **рекомендуется**, чтобы в данный момент таких файлов не было, поскольку их форма и назначение могут быть определены в будущем. В настоящее время URL следует просто играть роль класса имён, которые могут заполняться локальным администратором.

Диспетчерам MUD следует соблюдать особую осторожность при вызове класса controller в форме URL. Во-первых, для этого требуется понимание администратора в плане применимости. Приветствуется предварительная регистрация таких классов на сервере MUD. Механизм регистрации выходит за рамки этого документа.

4.7. my-controller

Этот пустой (null-valued) узел сигнализирует диспетчеру MUD об использовании сопоставления, которое у него есть для данного MUD URL, с конкретной группой хостов. Для этого может потребоваться запрос администратору о членах класса. В последующих работах следует стремиться к автоматизации управления принадлежностью к классу.

4.8. direction-initiated

Этот узел **должен** применяться только для TCP и сопоставляется с направлением, в котором инициируется соединение TCP. Когда соединение инициирует устройство (from-device), пакеты, передаваемые в направлении Вещи, **должны** отбрасываться, если только Вещь не инициировала первой соединение TCP. Например, этот узел может быть реализован в простейшей форме путём просмотра явных битов SYN, а также иными способами с использованием механизмов с большей поддержкой состояний.

При использовании узла пакеты соответствовать будут пакеты потока, инициированного в соответствующем направлении. В [RFC6092] приведены рекомендации для IPv6. Хотя документ разработан специально для IPv6, его содержимое применимо и к IPv4.

5. Обработка файла MUD

Для сохранения простоты в дополнение к имеющимся определениям добавлены два варианта принятого по умолчанию поведения:

- запрещено все, что не разрешено явно;
- взаимодействие с локальным DNS и NTP по умолчанию разрешено в обоих направлениях относительно Вещи.

[1] В оригинале ошибочно указано flobbidy.example.com, см. https://www.rfc-editor.org/errata/eid6295. *Прим. перев.*

Явное описание принятых по умолчанию правил приведено в Приложении A. Эти правила применяются после всех правил, заданных явно. Таким образом, принятое по умолчанию поведение можно изменить действием drop.

6. Как выглядит MUD URL?

В MUD URL требуется применять схему https для отождествления серверов файлов MUD и обеспечения целостности MUD-файлов. В качестве MUD URL может применяться любой URL https://, например,

```
https://things.example.org/product_abc123/v5
https://www.example.net/mudfiles/temperature_sensor/
https://example.com/lightbulbs/colour/v1
```

Изготовитель может создавать MUD URL любым способом при условии использования схемы https.

7. Модель YANG MUD

```
<CODE BEGINS>file "ietf-mud@2019-01-28.yang"
module ietf-mud {
  yang-version 1.1;
  namespace "urn:ietf:params:xml:ns:yang:ietf-mud";
  prefix ietf-mud;

  import ietf-access-control-list {
    prefix acl;
  }
  import ietf-yang-types {
    prefix yang;
  }
  import ietf-inet-types {
    prefix inet;
  }

  organization
    "IETF OPSAWG (Operations and Management Area Working Group)";
  contact
    "WG Web: <https://datatracker.ietf.org/wg/opsawg/>
     WG List: opsawg@ietf.org

     Author: Eliot Lear
             lear@cisco.com

     Author: Ralph Droms
             rdroms@gmail.com

     Author: Dan Romascanu
             dromasca@gmail.com
    ";
  description
    "Этот модуль YANG задаёт дополнение к описанию IETF для списков
     доступа. Модуль сосредоточен на дополнительных фильтрах,
     включающих local, model и same-manufacturer.

     Модуль предназначен для представления в формате JSON и сохранения
     в файле, как описано в RFC 8520.

     Ключевые слова ДОЛЖНО, НЕДОПУСТИМО, ТРЕБУЕТСЯ, НУЖНО, НЕ НУЖНО,
     СЛЕДУЕТ, НЕ СЛЕДУЕТ, РЕКОМЕНДУЕТСЯ, НЕ РЕКОМЕНДУЕТСЯ, МОЖНО,
     НЕОБЯЗАТЕЛЬНО в этом документе трактуются в соответствии с
     BCP 14 (RFC 2119) (RFC 8174) тогда и только тогда, когда они
     указаны заглавными буквами, как показано здесь.
     Авторские права (Copyright (c) 2019) принадлежат IETF Trust
     и лицам, указанным в качестве авторов кода. Все права защищены.

     Распространение и использование в исходной или двоичной форме с
     изменениями или без таковых разрешено в соответствии с лицензией
     Simplified BSD, изложенной в разделе 4  IETF Trust's Legal
     Provisions применительно к документам IETF
     (http://trustee.ietf.org/license-info).

     Эта версия данного модуля YANG является частью RFC 8520,
     где правовые вопросы рассмотрены более полно.";

  revision 2019-01-28 {
    description
      "Исходный предложенный стандарт.";
    reference
      "RFC 8520: Manufacturer Usage Description Specification";
  }

  typedef direction {
    type enumeration {
      enum to-device {
        description
          "Пакеты или потоки для целевой Вещи.";
      }
      enum from-device {
```

```
        description
          "Пакеты или потоки от целевой Вещи.";
      }
    }
    description
      "О каком направлении идёт речь?";
  }

  container mud {
    presence "Разрешено для этого конкретного MUD URL";
    description
      "Связанные с MUD сведения в соответствии с RFC 8520.";
    uses mud-grouping;
  }

  grouping mud-grouping {
    description
      "Сведения о времени завершения и обновления поддержки.";
    leaf mud-version {
      type uint8;
      mandatory true;
      description
        "Версия спецификации MUD (этот документ задаёт версию 1)";
    }
    leaf mud-url {
      type inet:uri;
      mandatory true;
      description
        "MUD URL, связанный с записью в файле MUD.";
    }
    leaf last-update {
      type yang:date-and-time;
      mandatory true;
      description
        "Предполагается, что это время создания текущего файла MUD.
         Диспетчерам MUD НЕ СЛЕДУЕТ проверять наличие обновлений в
         интервале между этим моментом и сроком действия кэша.";
    }
    leaf mud-signature {
      type inet:uri;
      description
        "URI, преобразующийся в подпись с соответствии с этой
         спецификацией.";
    }
    leaf cache-validity {
      type uint8 {
        range "1..168";
      }
      units "hours";
      default "48";
      description
        "Сведения, получаемые от сервера MUD, действительны в течение
         указанного числа часов, после чего их следует обновить.
         Важно отметить, что реализации диспетчера MUD не обязаны
         отбрасывать файлы MUD по истечении этого времени.";
    }
    leaf is-supported {
      type boolean;
      mandatory true;
      description
        "Указывает, поддерживается ли сейчас эта Вещь изготовителем.";
    }
    leaf systeminfo {
      type string;
      description
        "Описание (UTF-8) данной Вещи. Следует давать краткие
         сведения, которые могут выводиться пользователю для
         определения возможности установки Вещи в сеть.";
    }
    leaf mfg-name {
      type string;
      description
        "Заданное изготовителем имя в соответствии с модулем
         YANG ietf-hardware.";
    }
    leaf model-name {
      type string;
      description
        "Имя модели, как описано в модуле YANG ietf-hardware.";
    }
    leaf firmware-rev {
      type string;
      description
        "firmware-rev, как описано в модуле YANG ietf-hardware.
         Отметим, что это поле НЕДОПУСТИМО включать, если
         устройство можно обновить, а MUD URL - нет.";
```

```
    }
    leaf software-rev {
      type string;
      description
        "software-rev, как описано в модуле YANG ietf-hardware.
         Отметим, что это поле НЕДОПУСТИМО включать, если
         устройство можно обновить, а MUD URL - нет.";       }
    leaf documentation {
      type inet:uri;
      description
        "URL для документации, относящейся к устройству и любым
         классам, используемым в файле MUD. Отметим, что диспетчерам
         MUD не требуется преобразовывать этот URL и достаточно
         просто предоставить его администратору. Разбор HTML не
         входит в число функций диспетчера MUD.";
    }
    leaf-list extensions {
      type string {
        length "1..40";
      }
      description
        "Список имён расширений, используемых в файле MUD. Имена
         регистрируются IANA, а расширения описываются в RFC.";
    }
    container from-device-policy {
      description
        "Правила, применяемые к трафику от устройства. Правила не
         обязательно предназначены для применения в одной точке и
         могут разбираться контроллером для любой соответствующей
         точки применения правил в сети или ином месте.";
      uses access-lists;
    }
    container to-device-policy {
      description
        "Правила, применяемые к трафику на устройство. Правила не
         обязательно предназначены для применения в одной точке и
         могут разбираться контроллером для любой соответствующей
         точки применения правил в сети или ином месте.";
      uses access-lists;
    }
  }

  grouping access-lists {
    description
      "Группировка списков доступа в контексте правил устройства.";
    container access-lists {
      description
        "Списки доступа, которые следует применять для трафика
         от устройства или для него.";
      list access-list {
        key "name";
        description
          "Каждая запись списка указывает ACL, который следует
           указывать в общей модели данных списков доступа. ACL
           идентифицируются по имени и типу.";
        leaf name {
          type leafref {
            path "/acl:acls/acl:acl/acl:name";
          }
          description
            "Имя ACL для этой записи.";
        }
      }
    }
  }

  augment "/acl:acls/acl:acl/acl:aces/acl:ace/acl:matches" {
    description
      "Добавление абстракций для избавления от необходимости
       использовать адреса IP.";
    container mud {
      description
        "Связанные с MUD сопоставления.";
      leaf manufacturer {
        type inet:host;
        description
          "Домен, предназначенный для сопоставления с разделом
           authority в MUD URL. Узел служит для указания одного
           или нескольких изготовителей, которым следует разрешать
           доступ к устройству.";
      }
      leaf same-manufacturer {
        type empty;
        description
          "Узел, сопоставляемый с разделом authority в MUD URL Вещи.
           Узел предназначен для предоставления доступа ко всем
```

```
            устройствам с таким разделом authority.";
        }
        leaf model {
          type inet:uri;
          description
            "Устройства с указанным типом модели будут соответствовать,
             при идентичности MUD URL.";
        }
        leaf local-networks {
          type empty;
          description
            "Адреса IP будут соответствовать этому узлу, если они
             считаются локальными (список локально заданных префиксов и
             масок, указывающий конкретную административную область.";
        }
        leaf controller {
          type inet:uri;
          description
            "Именует класс, с которым могут быть связаны адреса IP для
             сопоставления. Это может быть привязано к изготовителю или
             стандартному URN.";
        }
        leaf my-controller {
          type empty;
          description
            "Этот узел сопоставляется с элементами сети, настроенными в
             качестве контроллеров для этой Вещи, на основе MUD URL.";
        }
      }
    }
    augment "/acl:acls/acl:acl/acl:aces/acl:ace/acl:matches"
          + "/acl:l4/acl:tcp/acl:tcp" {
      description
        "add direction-initiated";
      leaf direction-initiated {
        type direction;
        description
          "Этот узел сопоставляется с направлением, в котором
           инициировано соединение. Способы определения направления
           рассмотрены в этом документе.";
      }
    }
  }
  <CODE ENDS>
```

8. Расширение для доменных имён в модели ACL

Этот модуль задаёт расширение модели IETF-ACL, позволяющее указывать доменные имена, путём добавления узла matches. Разные реализации могут применять свои методы поддержки сопоставления адресов IP с доменными именами. Однако назначением является разрешение (или запрет) по спискам доступа для ресурсов, указанных именами. Структура изменений показана ниже.

```
module: ietf-acldns
  augment /acl:acls/acl:acl/acl:aces/acl:ace/
    acl:matches/acl:l3/acl:ipv4/acl:ipv4:
    +--rw src-dnsname?   inet:host
    +--rw dst-dnsname?   inet:host
  augment /acl:acls/acl:acl/acl:aces/acl:ace/
    acl:matches/acl:l3/acl:ipv6/acl:ipv6:
    +--rw src-dnsname?   inet:host
    +--rw dst-dnsname?   inet:host
```

Выбор конкретных точек в модели списка управления доступом основывается на допущении наличия некого способа указания связанных с IP ресурсов, например, по именам DNS. Доменные имена в этом контексте соответствуют определению [RFC6991]. Дополнения реплицируются через IPv4 и IPv6, чтобы авторы файлов MUD могли контролировать версию IP, которую Вещь может применять.

Узлы модуля описаны ниже.

8.1. src-dnsname

Аргумент соответствует доменному имени источника, указанному inet:host. Для распознавания хостов могут использоваться разные методы. Важно, чтобы такое распознавание было совместимо с ACL, которые требуются для корректной работы Вещи.

8.2. dst-dnsname

Аргумент соответствует доменному имени адресата, указанному inet:host. Распознавание имён описано выше (параграф 8.1).

Отметим, что при использовании любого из аргументов в файле MUD в этот файл **недопустимо** включать в ACL src-dnsname, связанный с from-device-policy, или dst-dnsname, связанный с to-device-policy, поскольку доступ относится к конкретной Вещи.

8.3. Модель ietf-acldns

```
<CODE BEGINS>file "ietf-acldns@2019-01-28.yang"
module ietf-acldns {
  yang-version 1.1;
  namespace "urn:ietf:params:xml:ns:yang:ietf-acldns";
  prefix ietf-acldns;

  import ietf-access-control-list {
    prefix acl;
  }
  import ietf-inet-types {
    prefix inet;
  }

  organization
    "IETF OPSAWG (Operations and Management Area Working Group)";
  contact
    "WG Web: <https://datatracker.ietf.org/wg/opsawg/>
     WG List: opsawg@ietf.org

     Author: Eliot Lear
             lear@cisco.com

     Author: Ralph Droms
             rdroms@gmail.com

     Author: Dan Romascanu
             dromasca@gmail.com
    ";
  description
    "Этот модуль YANG задаёт дополнение к описанию IETF для списков
     доступа, позволяющее сопоставления по именам DNS.

     Авторские права (Copyright (c) 2019) принадлежат IETF Trust
     и лицам, указанным в качестве авторов кода. Все права защищены.

     Распространение и использование в исходной или двоичной форме с
     изменениями или без таковых разрешено в соответствии с лицензией
     Simplified BSD, изложенной в разделе 4  IETF Trust's Legal
     Provisions применительно к документам IETF
     (http://trustee.ietf.org/license-info).

     Эта версия данного модуля YANG является частью RFC 8520,
     где правовые вопросы рассмотрены более полно.";

  revision 2019-01-28 {
    description
      "Базовая версия расширения dnsname для модели ACL.";
    reference
      "RFC 8520: Manufacturer Usage Description Specification";
  }

  grouping dns-matches {
    description
      "Доменные имена для сопоставления.";
    leaf src-dnsname {
      type inet:host;
      description
        "Доменное имя для сопоставления.";
    }
    leaf dst-dnsname {
      type inet:host;
      description
        "Доменное имя для сопоставления.";
    }
  }

  augment "/acl:acls/acl:acl/acl:aces/acl:ace/acl:matches"
        + "/acl:l3/acl:ipv4/acl:ipv4" {
    description
      "Добавление доменных имён для сопоставления.";
    uses dns-matches;
  }
  augment "/acl:acls/acl:acl/acl:aces/acl:ace/acl:matches"
        + "/acl:l3/acl:ipv6/acl:ipv6" {
    description
      "Добавление доменных имён для сопоставления .";
    uses dns-matches;
  }
}
<CODE ENDS>
```

9. Пример файла MUD

В этом примере показаны два списка для управления исходящим доступом к облачному сервису на порту TCP 443.

```
{
  "ietf-mud:mud": {
    "mud-version": 1,
    "mud-url": "https://lighting.example.com/lightbulb2000",
    "last-update": "2019-01-28T11:20:51+01:00",
    "cache-validity": 48,
    "is-supported": true,
    "systeminfo": "The BMS Example Light Bulb",
    "from-device-policy": {
      "access-lists": {
        "access-list": [
          {
            "name": "mud-76100-v6fr"
          }
        ]
      }
    },
    "to-device-policy": {
      "access-lists": {
        "access-list": [
          {
            "name": "mud-76100-v6to"
          }
        ]
      }
    }
  },
  "ietf-access-control-list:acls": {
    "acl": [
      {
        "name": "mud-76100-v6to",
        "type": "ipv6-acl-type",
        "aces": {
          "ace": [
            {
              "name": "cl0-todev",
              "matches": {
                "ipv6": {
                  "ietf-acldns:src-dnsname": "test.example.com",
                  "protocol": 6
                },
                "tcp": {
                  "ietf-mud:direction-initiated": "from-device",
                  "source-port": {
                    "operator": "eq",
                    "port": 443
                  }
                }
              },
              "actions": {
                "forwarding": "accept"
              }
            }
          ]
        }
      },
      {
        "name": "mud-76100-v6fr",
        "type": "ipv6-acl-type",
        "aces": {
          "ace": [
            {
              "name": "cl0-frdev",
              "matches": {
                "ipv6": {
                  "ietf-acldns:dst-dnsname": "test.example.com",
                  "protocol": 6
                },
                "tcp": {
                  "ietf-mud:direction-initiated": "from-device",
                  "destination-port": {
                    "operator": "eq",
                    "port": 443
                  }
                }
              },
              "actions": {
                "forwarding": "accept"
              }
            }
          ]
        }
      }
    ]
  }
}
```

В этом примере объявлено два правила - одно для трафика от Вещи, другое - к ней. В каждом правиле указано имя списка доступа, применяемого для пакетов. В каждом из списков доступ разрешается для пакетов, связанных с доменным именем test.example.com[1]. В каждом из списков предполагается, что соединение инициировано Вещью.

10. Опции DHCP MUD URL

Опция IPv4 MUD URL имеет показанный ниже формат

```
+------+-----+------------------------------
| code | len |  MUDstring
+------+-----+------------------------------
```

Код OPTION_MUD_URL_V4 (161) выделен IANA, len - 1-октетное значение, указывающее размер MUDstring в октетах. Определение MUDstring приведено ниже.

```
MUDstring = mudurl [ " " reserved ]
mudurl = URI; URL [RFC3986] со схемой https [RFC7230]
reserved = 1*( OCTET ) ; [RFC5234]
```

Опции **недопустимо** занимать более 255 октетов. Если после MUD URL следует пробел, за ним помещается зарезервированная строка, которая будет определена в последующих спецификациях. Диспетчеры MUD, не понимающие это поле, **должны** игнорировать его.

Формат опции клиента IPv6 MUD URL показан ниже.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         OPTION_MUD_URL_V6     |        option-length          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                               MUDstring                       |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

OPTION_MUD_URL_V6

112.

option-length

Размер MUDstring (см. определение выше) в октетах.

Назначением этой опции является предоставление сети нового классификатора Вещи, а также предоставление некоторых рекомендуемых конфигураций маршрутизаторам, реализующим эту опцию. Однако решение об этих сведениях принимает сетевая система, управляемая администратором сети. Основной задачей этой опции является просто идентификация типа Вещи для сети структурированным способом, чтобы правила можно было легко найти с помощью имеющихся инструментов.

10.1. Поведение клиента

Клиент DHCPv4 **может** передавать опцию DHCPv4, а клиент DHCPv6 - опцию DHCPv6. Эти опции являются одиночными (singleton), как указано в [RFC7227]. Поскольку предполагается, что у клиента имеется не более одно MUD URL, связанного с ним, тот может передавать не более одной опции MUD URL по протоколу DHCPv4 и одной - по протоколу DHCPv6. При передаче обеих опций DHCP (v4 и v6) **должно** указываться одно значение URL.

10.2. Поведение сервера

Сервер DHCP может игнорировать эти опции или предпринимать на их основе действия. Приняв опцию, сервер будет пересылать URL и соответствующую информацию клиента (такую как адрес шлюзе или giaddr и запрошенный адрес IP, а также продолжительность аренды) системе управления сетью или извлечь само описание применения по URL.

Серверы DHCP могут реализовать функциональность MUD самостоятельно или передавать вместе с соответствующими сведениями системе управления сетью или диспетчеру MUD. Сервер DHCP, обрабатывающий MUD URL **должен** придерживаться процесса, описанного в [RFC2818] и [RFC5280] для проверки сертификата TLS web-сервера, где размещается файл MUD. Такие серверы извлекают файл, обрабатывают его, а также создают и устанавливают требуемую конфигурацию на соответствующем элементе сети. Серверам **следует** отслеживать на шлюзе изменения состояний данного интерфейса. Сервер DHCP без функциональности MUD, пересылающий MUD URL диспетчеру MUD, **должен** уведомлять диспетчер MUD о любом соответствующих изменениях состояния DHCP для клиента, таких как завершение срока или явное прекращение аренды.

В случае отказа сервера DHCP с функциональностью диспетчера MUD любым резервным механизмам **следует** включать состояние MUD, а серверу **следует** распознавать статус своих клиентов после запуска, как он делал бы это при отсутствии функциональности MUD. Если сервер DHCP пересылает информацию диспетчеру MUD, тот использует для получения информации резервные серверы DHCP или сбрасывает состояние на основе других сведений из сети, таких как статус порта коммутатора, отслеживаемый через SNMP, учёт Radius или аналогичные механизмы.

10.3. Требования к ретранслятору

К ретрансляторам не предъявляется дополнительных требований.

11. Расширение MUD URL X.509

В этом разделе задано некритическое расширение сертификата X.509, содержащее один URL, указывающий на доступное через описание MUD, относящееся к субъекту сертификата. Идентификатор URI должен представляться, как описано в параграфе 7.4 [RFC5280]. Идентификаторы ресурсов на других языках (Internationalized Resource Identifier или IRI) **должны** отображаться на URI, как указано в параграфе 3.1 [RFC3987], до их включения в расширение сертификата. Семантика URL определена в разделе 6 этого документа.

Значение id-pe выбирается в соответствии с рекомендациями параграфа 4.2.2 в [RFC5280].

[1]В оригинале ошибочно указано service.bms.example.com, см. https://www.rfc-editor.org/errata/eid7173. *Прим. перев.*

Эти расширения могут служить для направления приложений к доступным через сеть сведениям об эмитенте или субъекте.

MUD URL является именно таким и указывает доступные через сеть сведения о конкретном субъекте.

Кроме того, определено новое расширение id-pe-mudsigner, содержащее поле субъекта из сертификата подписи файла MUD. Обработка поля описана в параграфе 13.2.

Подпись предназначена для подтверждения действительности найденного на сервере файла MUD для данного устройства, независимо от других факторов. В разделе 16 приведены связанные с этим соображения безопасности.

Определён также новый тип содержимого (content-type) id-ct-mud. Хотя в настоящее время подписи отделены, при передаче файла MUD в криптографическом сообщении (Cryptographic Message Syntax или CMS) **следует** применять этот content-type.

В модуле используется импорт из [RFC5912] и [RFC6268]. Новые расширения показаны ниже.

```
<CODE BEGINS>
   MUDURLExtnModule-2016 { iso(1) identified-organization(3) dod(6)
                internet(1) security(5) mechanisms(5) pkix(7)
                id-mod(0) id-mod-mudURLExtn2016(88) }
    DEFINITIONS IMPLICIT TAGS ::= BEGIN

    -- Экспорт всего --
   IMPORTS
     -- RFC 5912
     EXTENSION
     FROM PKIX-CommonTypes-2009
          { iso(1) identified-organization(3) dod(6) internet(1)
            security(5) mechanisms(5) pkix(7) id-mod(0)
            id-mod-pkixCommon-02(57) }

     -- RFC 5912
     id-ct
     FROM PKIXCRMF-2009
          { iso(1) identified-organization(3) dod(6) internet(1)
            security(5)  mechanisms(5) pkix(7) id-mod(0)
            id-mod-crmf2005-02(55) }

     -- RFC 6268
     CONTENT-TYPE
     FROM CryptographicMessageSyntax-2010
       { iso(1) member-body(2) us(840) rsadsi(113549)
         pkcs(1) pkcs-9(9) smime(16) modules(0) id-mod-cms-2009(58) }

     -- RFC 5912
     id-pe, Name
     FROM PKIX1Explicit-2009
           { iso(1) identified-organization(3) dod(6) internet(1)
             security(5) mechanisms(5) pkix(7) id-mod(0)
             id-mod-pkix1-explicit-02(51) } ;

     --
     -- Расширения сертификата
     --
     MUDCertExtensions EXTENSION ::=
        { ext-MUDURL | ext-MUDsigner, ... }

     ext-MUDURL EXTENSION ::=
        { SYNTAX MUDURLSyntax IDENTIFIED BY id-pe-mud-url }

     id-pe-mud-url OBJECT IDENTIFIER ::= { id-pe 25 }

     MUDURLSyntax ::= IA5String

     ext-MUDsigner EXTENSION ::=
        { SYNTAX MUDsignerSyntax IDENTIFIED BY id-pe-mudsigner }

     id-pe-mudsigner OBJECT IDENTIFIER ::= { id-pe 30 }

     MUDsignerSyntax ::= Name

     --
     -- Типы содержимого CMS
     --
     MUDContentTypes CONTENT-TYPE ::=
        { ct-mud, ... }

      ct-mud CONTENT-TYPE ::=
        { -- Включение содержимого напрямую
          IDENTIFIED BY id-ct-mudtype }
        -- Двоичные данные в форме application/mud+json кодируются
        -- напрямую как подписанные данные (без кодирования ASN.1).

     id-ct-mudtype OBJECT IDENTIFIER ::= { id-ct 41 }
     END
<CODE ENDS>
```

Хотя это расширение может присутствовать как в сертификате изготовителя 802.AR (IDevID) или сертификате внедрения (LDevID), его наличие и передача не гарантируется. Реализациям диспетчеров MUD **рекомендуется** поддерживать таблицу сопоставления Вещей с MUD URL на основе IDevID.

12. Расширение MUD LLDP

Протокол обнаружения на канальном уровне (Link Layer Discovery Protocol или LLDP) IEEE802.1AB является локальным (one-hop), не зависящим от производителя протоколом канального уровня, используемым Вещами для анонсирования своих отождествлений, возможностей и соседей в локальной сети IEEE 802. Элемент TLV[1] позволяет задавать фирменные (vendor-specific) расширения. Агентство IANA зарегистрировало уникальный идентификатор организации (organizationally unique identifier или OUI) IEEE 802, определённый в соответствии с [RFC7042]. Расширение MUD LLDP использует определённый в этом документе субтип для передачи MUD URL. Формат кадра LLDP приведён ниже.

```
+---------+---------+----------+---------+-------------
|TLV Type |  len    |   OUI    |subtype  | MUDString
|  =127   |         |= 00 00 5E|  = 1    |
|(7 битов)|(9 битов)|(3 октета)|(1 октет)|(1-255 октетов)
+---------+---------+----------+---------+-------------
```

TLV Type = 127

Указывает фирменный элемент TLV.

len

Указывает размер строки TLV.

OUI = 00 00 5E

Уникальный идентификатор организации, выделенный IANA.

subtype = 1

Значение, выделенное IANA для MUDstring.

MUDstring

Строке **недопустимо** иметь размер более 255 октетов.

Назначение этого расширения состоит в предоставлении сети нового классификатора Вещи, а также некой рекомендуемой конфигурации для маршрутизаторов, реализующих правила. Однако решение о применении этих сведений полностью определяется сетевой системой, управляемой администратором. Основная функция расширения состоит в идентификации типа Вещи для сети структурированным способом, чтобы правила можно было легко найти с помощью имеющихся инструментов.

Хосты, маршрутизаторы и другие элементы сети, реализующие эту опцию, должны иметь не более одного связанного с ними MUD URL, поэтому они могут передавать не более одного значения MUD URL. Хосты, маршрутизаторы и другие элементы сети, реализующие эту опцию, могут игнорировать её или выполнять на основе опции те или иные действия, например, вносить сведения в соответствующие расширения LLDP MIB[2]. LLDP работает в одном направлении. Обмен блоками данных протокола (Link Layer Discovery Protocol Data Unit или LLDPDU) не происходит в форме запросов одной Вещи и откликов другой. Приём полученных Вещью сведений LLDP эта Вещь не подтверждает Вещи-отправителю. Определённое поведение сети не гарантируется. Когда Вещь воспринимает это расширение, она может переслать URL и соответствующие сведения от удалённой вещи диспетчеру MUD или извлечь описание использования путём распознавания URL в соответствии с обычной семантикой HTTP.

13. Создание и обработка подписанных файлов MUD

Поскольку файлы MUD содержат сведения, которые могут служить для настройки сетевых списков доступа, они могут быть конфиденциальными. Для предотвращения возможности подделки важно подписывать файлы. Для этого служит синтаксис кодированных сообщений (Cryptographic Message Syntax или CMS) [RFC5652] с кодированием DER.

13.1. Создание подписи файла MUD

Файл MUD **должен** быть подписан с использованием CMS как необрабатываемого двоичного объекта. Для повышения вероятности успешной проверки **следует** включать промежуточные сертификаты. Подпись хранится в указанном файлом MUD месте. Подписи передаются с применением типа содержимого application/pkcs7-signature. Например,

```
% openssl cms -sign -signer mancertfile -inkey mankey \
              -in mudfile -binary -outform DER -binary \
              -certfile intermediatecert -out mudfile.p7s
```

Примечание. Может потребоваться повторная подпись файла MUD, если срок действия прежней подписи истёк.

13.2. Проверка подписи файла MUD

Перед обработкой остальной части файла MUD диспетчер MUD **должен** получить подпись MUD путём извлечения поля mud-signature и проверки подписи в файле MUD. В сертификате подписи **должно** присутствовать расширение Key Usage с битом digitalSignature(0). При наличии расширения id-pe-mudsigner в сертификате устройства X.509 файл подписи MUD **должен** генерироваться с сертификатом, в котором subject соответствует содержимому id-pe-mudsigner. Если это условие не выполняется или не удаётся проверить цепочку доверия до известной привязки доверия, диспетчер MUD **должен** прекратить обработку файла MUD до получения разрешения от администратора.

Цель подписи состоит в назначение ответственного, репутация которого может помочь администратору при решении вопроса о восприятии данного файла MUD. Проверка репутации местоположения сервера, где размещён файл, уже стала обычным делом для web. Хотя вероятно, что изготовитель будет подписывать файл, но это необязательно, а иногда может быть нежелательным. С одной стороны, в некоторых средах интеграторы могут устанавливать свои сертификаты, с другой, может быть более важна ответственность за рекомендации, а не только взаимоотношения между Вещью и файлом. Например,

```
% openssl cms -verify -in mudfile.p7s -inform DER -content mudfile
```

Следует отметить дополнительную проверку общего корня доверия.

[1]Type-Length-Value - тип, размер, значение.
[2]Management Information Base - база сведений для управления.

14. Расширяемость

Одной из целей является понимание файлов MUD как можно более широким кругом систем. В сочетании с решением применять имеющиеся механизмы не остаётся возможности согласовывать расширения, а возможности их добавления ограничены. В связи с этим используется двухуровневая модель расширяемости, как указано ниже.

1. На грубом уровне версия протокола включается в MUD URL (этот документ задаёт версию 1). При использовании этой версии принимаются все изменения, а переходы на другие версии будут рассматриваться в будущих документах.

2. На более тонком уровне принимаются лишь расширения не влекущие дополнительных рисков для Вещи. В частности, добавление узлов в контейнер mud разрешается при понимании того, что эти расширения будут игнорироваться не понимающими их реализациями. Все такие расширения **нужно** стандартизовать через процесс IETF и они **должны** включаться в список extensions. Диспетчеры MUD **должны** игнорировать непонятные узлы YANG и **следует** создавать расширения, которые администратор может разрешить для предотвращения каких-либо несоответствий в правилах.

15. Вопросы внедрения

Поскольку MUD состоит из множества архитектурных блоков, можно применять различные варианты развёртывания. Одним из ключевых аспектов является место применения правил. Для защиты Вещи от других Вещей внутри локального развёртывания правила могут применяться на ближайшем коммутаторе или точке доступа. Для ограничения нежелательного трафика внутри сети целесообразно применять правила как можно ближе к Internet. В некоторых случаях применение правил на ближайшем этапе пересылки (hop) может оказаться невозможным. В этом случае повышается риск вторичного заражения (заражения устройств, расположенных близко одно к другому) возрастает для многих Вещей, которые могут взаимодействовать без защиты.

Предупреждение для некоторых классов. Включение Вещи в классы manufacturer и same-manufacturer может влиять на доступ к другим Вещам. Иными словами, такое включение может расширять список доступа на коммутаторах, соединённых с другими Вещами, в зависимости от управления доступом. Следует внимательно относиться к расширению списка доступа. Для сохранения разумных пределов расширения списка могут применяться дополнительные методы, такие как сегментация сети.

Поскольку на момент написания этого документа концепция MUD является новой, можно предположить, что на многих устройствах она ещё не реализована. Решение о предоставлении первому подключившемуся устройству широкого или ограниченного доступа остаётся за локальной системой. Кроме того, как отмечено во введении, при развёртывании может быть выбрано игнорирование политики MUD в целом и простое восприятие MUD URL как классификатора для применения в локальной политике.

Ниже приведены сведения о сроке действия и использовании доменных имён.

16. Вопросы безопасности

В зависимости от способа выдачи MUD URL Вещь может быть способна обманывать пользователя, получая дополнительный доступ в сеть. Это может выполняться разными способами при выдаче MUD URL через DHCP или LLDP. Например, неправомерное включение в класс (такой как same-manufacturer), будет давать доступ к устройствам, таким как my-controller, или к ресурсам Internet, который иначе был бы запрещён. Такие возможности зависят ото конкретного развёртывания. Реализации **следует** делать настраиваемыми для запрета дополнительного доступа к устройствам с использованием MUD URL, которые не выдаются защищённым путём, например, в сертификате. Реализациям **не следует** предоставлять расширенные разрешения (помимо предоставляемых устройствам без политики MUD) устройствам, которые не привязывают строго своё отождествление к передачам L2/L3. При использовании диспетчером MUD незащищённых методов, в классы **не следует** включать устройства, применяющие защищённые и незащищённые методы, чтобы предотвратить атаки с повышением полномочий, и **недопустимо** включать устройства с одним MUD URL, полученным с помощью строгих и слабых методов аутентификации.

Устройства могут подделывать сведения об источнике (L2/L3). При внедрении следует применять подобающие средства защиты для привязки взаимодействий к аутентификации. Для аутентификации 802.1X одним из средств является IEEE 802.1AE (MACsec) [IEEE8021AE]. Похожий подход может применяться с 802.11i (Wi-Fi Protected Access 2 (WPA2)) [IEEE80211i]. Для иных технологий нижнего уровня доступны другие варианты. Реализациям, применяющим ориентированный на сессии доступ без криптографической привязки, следует озаботиться удалением состояния при обнаружении разрыва сессии по любой причине.

Мошеннический удостоверяющий центр (certification authority или CA) может подписывать сертификаты с тем же именем субъекта, которое указано в поле MUDsigner сертификата изготовителя, разрешая, тем самым, подмену файла MUD для устройства. Имеется два способа смягчения таких угроз. Во-первых, смена подписывающей стороны может быть отмечена диспетчером MUD как особая ситуация. Во-вторых, при смене файла MUD диспетчеру MUD **следует** запрашивать разрешение у администратора (это следует делать в любом случае). При любых обстоятельствах диспетчер MUD **должен** поддерживать для этих целей кэш доверенных CA. При обнаружении мошенника его **следует** удалять. Дополнительные меры смягчения описаны ниже.

При отсутствии сертификатов Вещи, заявляющие принадлежность к конкретному производителю, **не следует** включать в группу этого производителя без той или иной проверки. Это актуально при использовании диспетчером MUD таких примитивов, как manufacturer, для доступа Вещей определённого типа. Системы управления сетью могут быть способны получать «отпечатки» (fingerprint) Вещей. В таких случаях MUD URL может служить классификатором для разрешения или запрета. Отпечатки могут давать и другие преимущества, например, при использовании сертификатов 802.1AR, которые сами не могут меняться и отпечатки позволяют добавить в MUDstring артефакты в форме резервного поля, описанного в разделе 10. Значение таких артефактов будут рассмотрено в последующих работах.

Диспетчерам MUD **не следует** воспринимать описание применения для Вещей с тем же MAC-адресом, который указывает смену полномочности (authority) URL, без дополнительной проверки (например, администратором сети). Новые Вещи, представляющие не проверенный на подлинность MUD URL, **следует** проверять тем или иным внешним способом, если они получают расширенный доступ в сеть.

Недобросовестный изготовитель может неправомерно применить анализатор файлов MUD, чтобы воспользоваться уязвимостью. Для борьбы с такими угрозами рекомендуется два подхода. Первый заключается в проверке того, что подписавшая файл MUD сторона известна и является доверенной для диспетчера MUD. Второй заключается в предварительном сканировании файла для проверки возможности его разбора и некого уровня правдоподобия. Файлы MUD вероятно будут относительно небольшими, чтобы сделать это. Числу ACE, используемых любой данной Вещью, следует также быть относительно небольшим. Полезно также может быть извлечение MUD URL лишь с известных файлов, имеющих достойную репутацию в web или домене.

При работе с URL требуется использовать доменные имена. Если имя владельца доменного имени меняется, новый владелец может предоставить файлы MUD, которые диспетчеры MUD сочтут действительными. Диспетчерам MUD **следует** кэшировать сертификаты, используемые сервером файлов MUD. При извлечении нового сертификата (по какой-либо причине) диспетчеру MUD следует проверить, не сменился ли владелец домена. Подходящим программным способом такой проверки является смена серверов доменных имён. Если фактический файл MUD изменён, диспетчер MUD **может** обратиться к базе данных WHOIS для просмотра сведений о владельце домена. Если произошли изменения или по каким-то причинам невозможно определить смену владельца, может потребоваться дополнительная проверка. Отметим, что в данном случае не рассматриваются ситуации, когда Вещь произведена достаточно давно, а в эксплуатацию введена недавно, или был установлен новый диспетчер MUD.

Публикация Вещью MUD URL раскрывает Вещь и даёт злоумышленникам представление о возможных уязвимостях. Хотя от MUD URL не требуется уникальность для конкретной Вещи, публикация URL в сочетании с другими сведениями может помочь наблюдателям идентифицировать людей. Для решения этих проблем разработчикам следует принимать во внимание дополнительные сведения, анонсируемые с помощью таких механизмов, как Multicast DNS (mDNS) [RFC6762[1]], а также способы идентификации Вещи в иных случаях, возможно, по поведению при подключении к сети, предназначению Вещи для индивидуального использования или передачи персональных данных, и затем применять подходящие методы минимизации данных. Один из подходов заключается в применении TEAP [RFC7170] в качестве средства обмена информацией с полномочными компонентами сети. Элементы сети также могут помочь в ограничении доступа к MUD URL с помощью таких механизмов, как DHCPv6-Shield [RFC7610].

Существует опасность слежения за самим диспетчером MUD с целью определения подключённых к сети Вещей. Для снижения риска диспетчеры MUD могут применять доверенные TLS-прокси, которые будут агрегировать информацию.

Следует отметить, что соображения безопасности из параграфа 3.7 в [RFC8407] в этом случае неприменимы, поскольку сериализованная форма YANG не предназначена для доступа через NETCONF. Однако для тех, кто пытается внедрить эту модель в элемент сети про протоколу настройки сети (Network Configuration Protocol или NETCONF), все объекты в каждой модели, описанной в этом документе, обладают характеристики безопасности, аналогичными указанным в [RFC8519]. Основным назначением MUD является настройка доступа, что не позволяет использовать MUD для разрушительных действий неуполномоченных сторон.

17. Взаимодействия с IANA

17.1. Регистрация модулей YANG

Указанные ниже модули YANG включены в реестр YANG Module Names.

```
Name: ietf-mud
URN: urn:ietf:params:xml:ns:yang:ietf-mud
Prefix: ietf-mud
Registrant contact: The IESG
Reference: RFC 8520

Name: ietf-acldns
URI: urn:ietf:params:xml:ns:yang:ietf-acldns
Prefix: ietf-acldns
Registrant contact: The IESG
Reference: RFC 8520
```

17.2. Регистрация URI

Агентство IANA добавило в реестр IETF XML registry указанные ниже записи.

```
URI: urn:ietf:params:xml:ns:yang:ietf-acldns
Registrant Contact: The IESG.
XML: N/A. Запрошенный URI является пространством имён XML.

URI: urn:ietf:params:xml:ns:yang:ietf-mud
Registrant Contact: The IESG.
XML: N/A. Запрошенный URI является пространством имён XML.
```

17.3. Опции DHCPv4 и DHCPv6

Агентство IANA выделило значение OPTION_MUD_URL_V4 (161) в реестр Dynamic Host Configuration Protocol (DHCP) and Bootstrap Protocol (BOOTP) Parameters и OPTION_MUD_URL_V6 (112) в реестр Dynamic Host Configuration Protocol for IPv6 (DHCPv6), как описано в разделе 10.

17.4. Расширения PKIX

Агенство IANA выделило указанные ниже значения.

- Модуль ASN.1 MUDURLExtnModule-2016 (88) в реестре SMI Security for PKIX Module Identifier (1.3.6.1.5.5.7.0).
- Идентификатор объекта id-pe-mud-url (25) в реестре SMI Security for PKIX Certificate Extension (1.3.6.1.5.5.7.1).
- Идентификатор объекта id-pe-mudsigner (30) в реестре SMI Security for PKIX Certificate Extension.

[1] В оригинале ошибочно указано RFC6872, см. https://www.rfc-editor.org/errata/eid5702. *Прим. перев.*

- Идентификатор объекта id-ct-mudtype (41) в реестре SMI Security for S/MIME CMS Content Type.

Использование этих значений описано в разделе 11.

17.5. Регистрация типов носителей для файлов MUD

Для передачи фалов MUD задан указанный ниже тип носителя.

Имя типа: application.

Имя субтипа: mud+json.

Требуемые параметры: N/A.

Необязательные параметры: N/A.

Кодирование: 8 битов, значения application/mud+json представляются объектами JSON, должна применяться кодировка UTF-8 [RFC3629].

Вопросы безопасности: см. раздел «Вопросы безопасности» в RFC 8520 и раздел 12 в [RFC8259].

Вопросы функциональной совместимости: N/A

Опубликованная спецификация: RFC 8520

Приложения, использующие этот тип носителя: диспетчеры MUD, описанные в RFC 8520.

Вопросы идентификаторов фрагментов: N/A

Дополнительные сведения:

Magic number(s): N/A

Расширения файлов: N/A

Код типа файлов Macintosh: N/A

Контактные данные:

Eliot Lear <lear@cisco.com>, Ralph Droms <rdroms@gmail.com>,

Dan Romascanu <dromasca@gmail.com>

Предполагаемое использование: COMMON

Ограничения на использование: нет

Авторы:

Eliot Lear <lear@cisco.com>

Ralph Droms <rdroms@gmail.com>

Dan Romascanu <dromasca@gmail.com>

Контролёр изменений: IESG

Временная регистрация? (только для стандартных деревьев): нет.

17.6. Реестр IANA для субтипов LLDP TLV

Агентство IANA создало новый реестр IANA Link Layer Discovery Protocol (LLDP) TLV Subtypes в IEEE 802 Numbers с процедурой выделения значений Expert Review [RFC8126]. Максимальное число записей в реестре - 256. Исходное содержимое реестра указано ниже.

```
Значение субтипа LLDP: 1 (остальные значения указаны как Unassigned)
Описание: ссылка MUD URL
Документ: RFC 8520
```

17.7. Общеизвестное имя MUD URN

В соответствии с [RFC3553] добавлен указанный ниже реестр параметров.

```
Имя реестра: MUD Well-Known Uniform[1] Resource Name (URN)
Спецификация: RFC 8520
Репозиторий: https://www.iana.org/assignments/mud
Значение индекса: Кодируется аналогично именам служб для портов TCP/UDP
                  в соответствии с параграфом 5.1 в [RFC6335]
```

В реестр MUD Well-Known Uniform1 Resource Name (URN) добавлена указанная ниже запись.

```
urn:ietf:params:mud:dns указывает службу, заданную [RFC1123],
urn:ietf:params:mud:ntp - службу, заданную [RFC5905].
```

17.8. Реестр расширений

Агентство IANA организовало указанный ниже реестр расширений.

```
Имя реестра: MUD Extensions
Процедура регистрации: Standards Action
Документ: RFC 8520
Имя расширения: строка UTF-8, не более 40 символов.
```

[1] В оригинале ошибочно указано Universal, см. https://www.rfc-editor.org/errata/eid5664. *Прим. перев.*

Само расширение **должно** следовать правилам, указанным в этой спецификации. IANA выдаёт упреждающие назначения в соответствии с [RFC7120].

18. Литература

18.1. Нормативные документы

[IEEE8021AB] IEEE, "IEEE Standard for Local and Metropolitan Area Networks-- Station and Media Access Control Connectivity Discovery", IEEE 802.1AB.

[RFC1123] Braden, R., Ed., "Requirements for Internet Hosts - Application and Support", STD 3, RFC 1123, DOI 10.17487/RFC1123, October 1989, <https://www.rfc-editor.org/info/rfc1123>.

[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>.

[RFC2131] Droms, R., "Dynamic Host Configuration Protocol", RFC 2131, DOI 10.17487/RFC2131, March 1997, <https://www.rfc-editor.org/info/rfc2131>.

[RFC2818] Rescorla, E., "HTTP Over TLS", RFC 2818, DOI 10.17487/RFC2818, May 2000, <https://www.rfc-editor.org/info/rfc2818>.

[RFC3629] Yergeau, F., "UTF-8, a transformation format of ISO 10646", STD 63, RFC 3629, DOI 10.17487/RFC3629, November 2003, <https://www.rfc-editor.org/info/rfc3629>.

[RFC3748] Aboba, B., Blunk, L., Vollbrecht, J., Carlson, J., and H. Levkowetz, Ed., "Extensible Authentication Protocol (EAP)", RFC 3748, DOI 10.17487/RFC3748, June 2004, <https://www.rfc-editor.org/info/rfc3748>.

[RFC3986] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax", STD 66, RFC 3986, DOI 10.17487/RFC3986, January 2005, <https://www.rfc-editor.org/info/rfc3986>.

[RFC3987] Duerst, M. and M. Suignard, "Internationalized Resource Identifiers (IRIs)", RFC 3987, DOI 10.17487/RFC3987, January 2005, <https://www.rfc-editor.org/info/rfc3987>.

[RFC5234] Crocker, D., Ed. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", STD 68, RFC 5234, DOI 10.17487/RFC5234, January 2008, <https://www.rfc-editor.org/info/rfc5234>.

[RFC5280] Cooper, D., Santesson, S., Farrell, S., Boeyen, S., Housley, R., and W. Polk, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 5280, DOI 10.17487/RFC5280, May 2008, <https://www.rfc-editor.org/info/rfc5280>.

[RFC5652] Housley, R., "Cryptographic Message Syntax (CMS)", STD 70, RFC 5652, DOI 10.17487/RFC5652, September 2009, <https://www.rfc-editor.org/info/rfc5652>.

[RFC5905] Mills, D., Martin, J., Ed., Burbank, J., and W. Kasch, "Network Time Protocol Version 4: Protocol and Algorithms Specification", RFC 5905, DOI 10.17487/RFC5905, June 2010, <https://www.rfc-editor.org/info/rfc5905>.

[RFC5912] Hoffman, P. and J. Schaad, "New ASN.1 Modules for the Public Key Infrastructure Using X.509 (PKIX)", RFC 5912, DOI 10.17487/RFC5912, June 2010, <https://www.rfc-editor.org/info/rfc5912>.

[RFC6268] Schaad, J. and S. Turner, "Additional New ASN.1 Modules for the Cryptographic Message Syntax (CMS) and the Public Key Infrastructure Using X.509 (PKIX)", RFC 6268, DOI 10.17487/RFC6268, July 2011, <https://www.rfc-editor.org/info/rfc6268>.

[RFC6335] Cotton, M., Eggert, L., Touch, J., Westerlund, M., and S. Cheshire, "Internet Assigned Numbers Authority (IANA) Procedures for the Management of the Service Name and Transport Protocol Port Number Registry", BCP 165, RFC 6335, DOI 10.17487/RFC6335, August 2011, <https://www.rfc-editor.org/info/rfc6335>.

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

[RFC7120] Cotton, M., "Early IANA Allocation of Standards Track Code Points", BCP 100, RFC 7120, DOI 10.17487/RFC7120, January 2014, <https://www.rfc-editor.org/info/rfc7120>.

[RFC7227] Hankins, D., Mrugalski, T., Siodelski, M., Jiang, S., and S. Krishnan, "Guidelines for Creating New DHCPv6 Options", BCP 187, RFC 7227, DOI 10.17487/RFC7227, May 2014, <https://www.rfc-editor.org/info/rfc7227>.

[RFC7230] Fielding, R., Ed. and J. Reschke, Ed., "Hypertext Transfer Protocol (HTTP/1.1): Message Syntax and Routing", RFC 7230, DOI 10.17487/RFC7230, June 2014, <https://www.rfc-editor.org/info/rfc7230>.

[RFC7231] Fielding, R., Ed. and J. Reschke, Ed., "Hypertext Transfer Protocol (HTTP/1.1): Semantics and Content", RFC 7231, DOI 10.17487/RFC7231, June 2014, <https://www.rfc-editor.org/info/rfc7231>.

[RFC7610] Gont, F., Liu, W., and G. Van de Velde, "DHCPv6-Shield: Protecting against Rogue DHCPv6 Servers", BCP 199, RFC 7610, DOI 10.17487/RFC7610, August 2015, <https://www.rfc-editor.org/info/rfc7610>.

[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>.

[RFC7951] Lhotka, L., "JSON Encoding of Data Modeled with YANG", RFC 7951, DOI 10.17487/RFC7951, August 2016, <https://www.rfc-editor.org/info/rfc7951>.

[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>.

[RFC8259] Bray, T., Ed., "The JavaScript Object Notation (JSON) Data Interchange Format", STD 90, RFC 8259, DOI 10.17487/RFC8259, December 2017, <https://www.rfc-editor.org/info/rfc8259>.

[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>.

[RFC8348] Bierman, A., Bjorklund, M., Dong, J., and D. Romascanu, "A YANG Data Model for Hardware Management", RFC 8348, DOI 10.17487/RFC8348, March 2018, <https://www.rfc-editor.org/info/rfc8348>.

[RFC8415] Mrugalski, T., Siodelski, M., Volz, B., Yourtchenko, A., Richardson, M., Jiang, S., Lemon, T., and T. Winters, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 8415, DOI 10.17487/RFC8415, November 2018, <https://www.rfc-editor.org/info/rfc8415>.

[RFC8519] Jethanandani, M., Agarwal, S., Huang, L., and D. Blair, "YANG Data Model for Network Access Control Lists (ACLs)", RFC 8519, DOI 10.17487/RFC8519, March 2019, <https://www.rfc-editor.org/info/rfc8519>.

18.2. Дополнительная литература

[FW95] Chapman, D. and E. Zwicky, "Building Internet Firewalls", First Edition, November 1995.

[IEEE80211i] IEEE, "IEEE Standard for information technology - Telecommunications and information exchange between systems-Local and metropolitan area networks-Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Amendment 6: Medium Access Control (MAC) Security Enhancements", IEEE 802.11i.

[IEEE8021AE] IEEE, "IEEE Standard for Local and metropolitan area networks-Media Access Control (MAC) Security", IEEE 802.1AE.

[IEEE8021AR] IEEE, "IEEE Standard for Local and metropolitan area networks - Secure Device Identity", IEEE 802.1AR.

[IEEE8021X] IEEE, "IEEE Standard for Local and metropolitan area networks--Port-Based Network Access Control", IEEE 802.1X.

[RFC1984] IAB and IESG, "IAB and IESG Statement on Cryptographic Technology and the Internet", BCP 200, RFC 1984, DOI 10.17487/RFC1984, August 1996, <https://www.rfc-editor.org/info/rfc1984>.

[RFC3553] Mealling, M., Masinter, L., Hardie, T., and G. Klyne, "ACK IETF URN Sub-namespace for Registered Protocol Parameters", BCP 73, RFC 3553, DOI 10.17487/RFC3553, June 2003, <https://www.rfc-editor.org/info/rfc3553>.

[RFC6092] Woodyatt, J., Ed., "Recommended Simple Security Capabilities in Customer Premises Equipment (CPE) for Providing Residential IPv6 Internet Service", RFC 6092, DOI 10.17487/RFC6092, January 2011, <https://www.rfc-editor.org/info/rfc6092>.

[RFC6762] Cheshire, S. and M. Krochmal, "Multicast DNS", RFC 6762, DOI 10.17487/RFC6762, February 2013, <https://www.rfc-editor.org/info/rfc6762>

[RFC7042] Eastlake 3rd, D. and J. Abley, "IANA Considerations and IETF Protocol and Documentation Usage for IEEE 802 Parameters", BCP 141, RFC 7042, DOI 10.17487/RFC7042, October 2013, <https://www.rfc-editor.org/info/rfc7042>.

[RFC7170] Zhou, H., Cam-Winget, N., Salowey, J., and S. Hanna, "Tunnel Extensible Authentication Protocol (TEAP) Version 1", RFC 7170, DOI 10.17487/RFC7170, May 2014, <https://www.rfc-editor.org/info/rfc7170>.

[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>.

[RFC7452] Tschofenig, H., Arkko, J., Thaler, D., and D. McPherson, "Architectural Considerations in Smart Object Networking", RFC 7452, DOI 10.17487/RFC7452, March 2015, <https://www.rfc-editor.org/info/rfc7452>.

[RFC7488] Boucadair, M., Penno, R., Wing, D., Patil, P., and T. Reddy, "Port Control Protocol (PCP) Server Selection", RFC 7488, DOI 10.17487/RFC7488, March 2015, <https://www.rfc-editor.org/info/rfc7488>.

[RFC8126] Cotton, M., Leiba, B., and T. Narten, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 8126, DOI 10.17487/RFC8126, June 2017, <https://www.rfc-editor.org/info/rfc8126>.

[RFC8343] Bjorklund, M., "A YANG Data Model for Interface Management", RFC 8343, DOI 10.17487/RFC8343, March 2018, <https://www.rfc-editor.org/info/rfc8343>.

[RFC8407] Bierman, A., "Guidelines for Authors and Reviewers of Documents Containing YANG Data Models", BCP 216, RFC 8407, DOI 10.17487/RFC8407, October 2018, <https://www.rfc-editor.org/info/rfc8407>.

Приложение A. Принятые по умолчанию узлы MUD

Ниже показана часть файла MUD, разрешающая трафик DNS к контроллеру, зарегистрированному с URN urn:ietf:params:mud:dns, и трафик NTP к контроллеру, зарегистрированному с urn:ietf:params:mud:ntp. Это считается принятым по умолчанию поведением и ACE фактически добавляются к другим записям ace в файле MUD. Для блокирования DNS или NTP повторяется оператор сопоставления с заменой при пересылке (forwarding) действия accept на drop. Поскольку ACE обрабатываются в порядке получения, принятые по умолчанию значения не будут достигнуты. Диспетчер MUD может выполнить дополнительную оптимизацию, просто не включая принятые по умолчанию значения, если они переопределены.

Ниже приведены 4 записи списка acl, реализующие принятые по умолчанию узлы MUD, две для IPv4 и две для IPv6 (по одной для каждого направления в обеих версиях IP). Отметим, что имя списка доступа и имя ace не требуется сохранять или как-то применять в реализациях, они указаны лишь для полноты.

```
    "ietf-access-control-list:acls": {
       "acl": [
         {
           "name": "mud-59776-v4to",
```

```
      "type": "ipv4-acl-type",
      "aces": {
        "ace": [
          {
            "name": "ent0-todev",
            "matches": {
              "ietf-mud:mud": {
                "controller": "urn:ietf:params:mud:dns"
              },
              "ipv4": {
                "protocol": 17
              },
              "udp": {
                "source-port": {
                  "operator": "eq",
                  "port": 53
                }
              }
            },
            "actions": {
              "forwarding": "accept"
            }
          },
          {
            "name": "ent1-todev",
            "matches": {
              "ietf-mud:mud": {
                "controller": "urn:ietf:params:mud:ntp"
              },
              "ipv4": {
                "protocol": 17
              },
              "udp": {
                "source-port": {
                  "operator": "eq",
                  "port": 123
                }
              }
            },
            "actions": {
              "forwarding": "accept"
            }
          }
        ]
      }
    },
    {
      "name": "mud-59776-v4fr",
      "type": "ipv4-acl-type",
      "aces": {
        "ace": [
          {
            "name": "ent0-frdev",
            "matches": {
              "ietf-mud:mud": {
                "controller": "urn:ietf:params:mud:dns"
              },
              "ipv4": {
                "protocol": 17
              },
              "udp": {
                "destination-port": {
                  "operator": "eq",
                  "port": 53
                }
              }
            },
            "actions": {
              "forwarding": "accept"
            }
          },
          {
            "name": "ent1-frdev",
            "matches": {
              "ietf-mud:mud": {
                "controller": "urn:ietf:params:mud:ntp"
              },
              "ipv4": {
                "protocol": 17
              },
              "udp": {
                "destination-port": {
                  "operator": "eq",
                  "port": 123
                }
              }
```

```
            },
            "actions": {
              "forwarding": "accept"
            }
          }
        ]
      }
    },
    {
      "name": "mud-59776-v6to",
      "type": "ipv6-acl-type",
      "aces": {
        "ace": [
          {
            "name": "ent0-todev",
            "matches": {
              "ietf-mud:mud": {
                "controller": "urn:ietf:params:mud:dns"
              },
              "ipv6": {
                "protocol": 17
              },
              "udp": {
                "source-port": {
                  "operator": "eq",
                  "port": 53
                }
              }
            },
            "actions": {
              "forwarding": "accept"
            }
          },
          {
            "name": "ent1-todev",
            "matches": {
              "ietf-mud:mud": {
                "controller": "urn:ietf:params:mud:ntp"
              },
              "ipv6": {
                "protocol": 17
              },
              "udp": {
                "source-port": {
                  "operator": "eq",
                  "port": 123
                }
              }
            },
            "actions": {
              "forwarding": "accept"
            }
          }
        ]
      }
    },
    {
      "name": "mud-59776-v6fr",
      "type": "ipv6-acl-type",
      "aces": {
        "ace": [
          {
            "name": "ent0-frdev",
            "matches": {
              "ietf-mud:mud": {
                "controller": "urn:ietf:params:mud:dns"
              },
              "ipv6": {
                "protocol": 17
              },
              "udp": {
                "destination-port": {
                  "operator": "eq",
                  "port": 53
                }
              }
            },
            "actions": {
              "forwarding": "accept"
            }
          },
          {
            "name": "ent1-frdev",
            "matches": {
              "ietf-mud:mud": {
                "controller": "urn:ietf:params:mud:ntp"
```

```
                },
                "ipv6": {
                  "protocol": 17
                },
                "udp": {
                  "destination-port": {
                    "operator": "eq",
                    "port": 123
                  }
                }
              },
              "actions": {
                "forwarding": "accept"
              }
            }
          ]
        }
      }
    ]
  }
```

Приложение B. Пример расширения DETNET-indicator

В этом примере базовая модель MUD дополняется для индикации устройством реализации DETNET. Если устройство заявляет, что оно не использует DETNET, а затем пытается применить его, может генерироваться уведомление или особый случай. Пример предназначен лишь для иллюстрации.

Имя расширения: Example-Extension (для использования в списке расширений).

Стандарт: RFC 8520 (без регистрации примера).

Это расширение дополняет модель MUD, включая один узел с использованием приведённого ниже модуля с деревом

```
module: ietf-mud-detext-example
  augment /ietf-mud:mud:
    +--rw is-detnet-required?   boolean
```

Модель приведена ниже.

```
<CODE BEGINS>file "ietf-mud-detext-example@2019-01-28.yang"
module ietf-mud-detext-example {
  yang-version 1.1;
  namespace "urn:ietf:params:xml:ns:yang:ietf-mud-detext-example";
  prefix ietf-mud-detext-example;

  import ietf-mud {
    prefix ietf-mud;
  }

  organization
    "IETF OPSAWG (Operations and Management Area Working Group)";
  contact
    "WG Web: <https://datatracker.ietf.org/wg/opsawg/>
     WG List: opsawg@ietf.org

     Author: Eliot Lear
             lear@cisco.com

     Author: Ralph Droms
             rdroms@gmail.com

     Author: Dan Romascanu
             dromasca@gmail.com
    ";
  description
    "Пример расширения модуля MUD для индикации поддержки DETNET.";

  revision 2019-01-28 {
    description
      "Исходный выпуск.";
    reference
      "RFC 8520: Manufacturer Usage Description Specification";
  }

  augment "/ietf-mud:mud" {
    description
      "Добавление простого расширения для указания производителем
       необходимости DETNET.";
    leaf is-detnet-required {
      type boolean;
      description
        "Значение true указывает необходимость поддержки DETNET
         для корректной работы.";
    }
  }
}
<CODE ENDS>
```

При использовании показанного выше примера расширение будет выражаться в показанной ниже форме.

```
{
  "ietf-mud:mud": {
    "mud-version": 1,
    "mud-url": "https://lighting.example.com/lightbulb2000",
    "last-update": "2019-01-28T11:20:51+01:00",
    "cache-validity": 48,
    "extensions": [
        "ietf-mud-detext-example"
     ],
    "ietf-mud-detext-example:is-detnet-required": "false",
    "is-supported": true,
    "systeminfo": "The BMS Example Light Bulb",
    "from-device-policy": {
      "access-lists": {
        "access-list": [
          {
            "name": "mud-76100-v6fr"
          }
        ]
      }
    },
    "to-device-policy": {
      "access-lists": {
        "access-list": [
          {
            "name": "mud-76100-v6to"
          }
        ]
      }
    }
  },
  "ietf-access-control-list:acls": {
    "acl": [
      {
        "name": "mud-76100-v6to",
        "type": "ipv6-acl-type",
        "aces": {
          "ace": [
            {
              "name": "cl0-todev",
              "matches": {
                "ipv6": {
                  "ietf-acldns:src-dnsname": "test.example.com",
                  "protocol": 6
                },
                "tcp": {
                  "ietf-mud:direction-initiated": "from-device",
                  "source-port": {
                    "operator": "eq",
                    "port": 443
                  }
                }
              },
              "actions": {
                "forwarding": "accept"
              }
            }
          ]
        }
      },
      {
        "name": "mud-76100-v6fr",
        "type": "ipv6-acl-type",
        "aces": {
          "ace": [
            {
              "name": "cl0-frdev",
              "matches": {
                "ipv6": {
                  "ietf-acldns:dst-dnsname": "test.example.com",
                  "protocol": 6
                },
                "tcp": {
                  "ietf-mud:direction-initiated": "from-device",
                  "destination-port": {
                    "operator": "eq",
                    "port": 443
                  }
                }
              },
              "actions": {
                "forwarding": "accept"
              }
            }
          ]
        }
```

```
          }
        ]
      }
    }
```

Благодарности

Авторы признательны Einar Nilsen-Nygaard, в одиночку обновившему модель в соответствии с обновлённой моделью ACL, а также Bernie Volz, Tom Gindin, Brian Weis, Sandeep Kumar, Thorsten Dahm, John Bashinski, Steve Rich, Jim Bieda, Dan Wing, Joe Clarke, Henk Birkholz, Adam Montville, Jim Schaad, Robert Sparks за ценные советы и рецензии. Russ Housley полностью переписал раздел 11, сделав модуль полноценным. Adrian Farrel предоставил основу для рассмотрения вопросов приватности, Kent Watsen - обзор архитектуры и модель YANG. Ответственность за ошибки в данной работе полностью принимают на себя авторы.

Адреса авторов

Eliot Lear
Cisco Systems
Richtistrasse 7
Wallisellen CH-8304
Switzerland
Phone: +41 44 878 9200
Email: lear@cisco.com

Ralph Droms
Google
355 Main St., 5th Floor
Cambridge, MA 02142
United States of America
Phone: +1 978 376 3731
Email: rdroms@gmail.com

Dan Romascanu
Phone: +972 54 5555347
Email: dromasca@gmail.com

Перевод на русский язык

Николай Малых

nmalykh@protokols.ru