

A YANG Data Model for DHCPv6 Configuration

Модель данных YANG для настройки DHCPv6

Аннотация

Этот документ описывает модель данных YANG для настройки и управления серверами, ретрансляторами и клиентами протокола динамической настройки конфигурации хостов IPv6 (Dynamic Host Configuration Protocol for IPv6 или DHCPv6) (RFC 8415).

Статус документа

Документ относится к категории Internet Standards Track.

Документ является результатом работы IETF¹ и представляет согласованный взгляд сообщества IETF. Документ прошёл открытое обсуждение и был одобрен для публикации IESG². Дополнительную информацию о стандартах Internet можно найти в разделе 2 в RFC 7841.

Информация о текущем статусе документа, найденных ошибках и способах обратной связи доступна по ссылке <https://www.rfc-editor.org/info/rfc9243>.

Авторские права

Copyright (c) 2022. Авторские права принадлежат IETF Trust и лицам, указанным в качестве авторов документа. Все права защищены.

К документу применимы права и ограничения, указанные в BCP 78 и IETF Trust Legal Provisions и относящиеся к документам IETF (<http://trustee.ietf.org/license-info>), на момент публикации данного документа. Прочтите упомянутые документы внимательно. Фрагменты программного кода, включённые в этот документ, распространяются в соответствии с упрощённой лицензией BSD, как указано в параграфе 4.e документа IETF Trust Legal Provisions, без каких-либо гарантий (как указано в Simplified BSD License).

Оглавление

1. Введение.....	2
1.1. Область действия.....	2
1.2. Расширяемость модуля YANG для сервера DHCPv6.....	2
1.2.1. Определения опций DHCPv6.....	2
2. Терминология.....	3
2.1. Уровни требований.....	3
3. Диаграммы деревьев DHCPv6.....	3
3.1. Диаграмма дерева сервера DHCPv6.....	3
3.2. Диаграмма дерева ретранслятора DHCPv6.....	6
3.3. Диаграмма дерева клиента DHCPv6.....	8
4. Модули YANG DHCPv6.....	10
4.1. Базовый модуль YANG DHCPv6.....	10
4.2. Модуль YANG для сервера DHCPv6.....	15
4.3. Модуль YANG для ретранслятора DHCPv6.....	26
4.4. Модуль YANG для клиента DHCPv6.....	31
5. Вопросы безопасности.....	40
6. Взаимодействие с IANA.....	41
6.1. Регистрация URI.....	41
6.2. Регистрация модулей YANG.....	41
7. Литература.....	41
7.1. Нормативные документы.....	41
7.2. Дополнительная литература.....	42
Приложение А. Примеры деревьев данных.....	42
А.1. Примеры конфигурации сервера DHCPv6.....	42
А.2. Пример конфигурации ретранслятора DHCPv6.....	44
А.3. Пример конфигурации клиента DHCPv6.....	45
Приложение В. Пример добавления определений опций DHCPv6.....	45
Приложение С. Пример фирменного модуля конфигурации сервера.....	47
Приложение D. Пример определения конфигурации селектора класса.....	50
Благодарности.....	54
Участники работы.....	54
Адрес автора.....	54

¹Internet Engineering Task Force - комиссия по решению инженерных задач Internet.

²Internet Engineering Steering Group - комиссия по инженерным разработкам Internet.

1. Введение

Протокол DHCPv6 [RFC8415] служит для предоставления конфигурации и других параметров клиентам сетей IPv6. Этот документ задаёт модуль YANG [RFC7950] для настройки и управления «элементами» DHCPv6 (серверы, ретрансляторы, клиенты), использующими протокол управления сетью (Network Configuration Protocol или NETCONF) [RFC6241] или RESTCONF [RFC8040]. Для каждого элемента задан свой модуль, а «базовый» модуль содержит определения типов (typedef) и группировки (grouping), применяемые модулями элементов. В Приложении А представлены примеры XML для каждого элемента и рассмотрено их взаимодействие.

Модули для ретранслятора и клиента обеспечивают конфигурацию, применяемую к интерфейсам устройств. Это делается с помощью модуля YANG ietf-interfaces [RFC8343] и ссылок interface-refs на соответствующие интерфейсы.

Следует отметить, что DHCPv6 сам по себе не является протоколом настройки клиента, поэтому назначением этого документа является предоставление замены выделенных DHCPv6 адресов и параметров соответствующими параметрами NETCONF и YANG. Модуль клиента DHCPv6 предназначен для настройки и мониторинга функции клиента DHCPv6 и не заменяет настройку адреса и параметров DHCPv6.

Модули YANG в этом документе соответствуют архитектуре хранилищ данных управления сетью (Network Management Datastore Architecture или NMDA) [RFC8342].

1.1. Область действия

В [RFC8415] описана текущая версия базового протокола DHCPv6. Было опубликовано множество дополнительных спецификаций, расширяющих функциональность элементов DHCPv6 и добавляющих новые опции. Модули YANG в этом документе не пытаются охватить все расширения, а, скорее, моделируют функции и опции DHCPv6 из [RFC8415]. Особое внимание уделялось расширяемости модулей, чтобы к ним можно было легко добавить функциональность, требуемую определённой реализацией или вариантом развёртывания.

1.2. Расширяемость модуля YANG для сервера DHCPv6

Модули в этом документе лишь пытаются моделировать поведение DHCPv6 и не включают настройку и управления конкретными реализациями. Различия между реализациями слишком велики, чтобы пытаться стандартизовать их так, чтобы это было полезно и не вносило ограничения.

Однако признано, что настройка и управление для конкретной реализации являются важной частью развёртывания и эксплуатации DHCP. Поэтому в Приложении С приведён пример модуля YANG для настройки зависящих от реализации функций с использованием дополнений (augment) базового модуля ietf-dhcpv6-server.yang.

В DHCPv6 применяется концепция выбора класса (class selection) для сообщений, получаемых сервером. Это идентификация и классификация сообщений на основе большого числа параметров, позволяющая предоставить корректные сведения о подготовке, например, путём выделения префикса из корректного пула или предоставления набора опций, относящихся к реализации клиента конкретным производителем. В процессе разработки этого документа были исследованы реализации и обнаружено, что эти функции существенно различаются, хотя и присущи всем реализациям. Поэтому настройка функции выбора класса не была включена в модуль сервера DHCPv6, чтобы производители могли задать свои модули YANG. В Приложении D приведён пример такого модуля, демонстрирующий объединение с основным модулем ietf-dhcpv6-server.yang.

1.2.1. Определения опций DHCPv6

Было определено множество опций DHCPv6 в дополнение к заданным в [RFC8415]. Поскольку реализации сильно различаются в плане поддержки опций DHCPv6, был принят решение о включении в этот документ лишь опций DHCPv6, определённых в [RFC8415]. Из этих опций для модели были выбраны лишь те, которые требуют настройки оператором. Например, опция OPTION_IA_NA (3) создаётся сервером DHCP по запросу клиента. Содержимое полей опции основано на множестве входных параметров, которые сервер будет применять при получении запроса (например, таймеры, T1/T2, связанные с пулом адресов). В результате для опции нет полей, которые можно настраивать напрямую, и она не включена в модель. В таблице 1 приведены моделируемые опции DHCPv6, элементы, для которых они моделируются, и указаны соответствующие модули YANG.

Опция	Таблица 1. Смоделированные опции DHCPv6.		Модуль
	Server	Relay Client	
OPTION_ORO (6) Option Request		X	ietf-dhcpv6-client.yang
OPTION_PREFERENCE (7) Preference	X		ietf-dhcpv6-server.yang
OPTION_AUTH (11) Authentication	X	X	ietf-dhcpv6-common.yang
OPTION_UNICAST (12) Server Unicast	X		ietf-dhcpv6-server.yang
OPTION_RAPID_COMMIT (14) Rapid Commit	X	X	ietf-dhcpv6-common.yang
OPTION_USER_CLASS (15) User Class		X	ietf-dhcpv6-client.yang
OPTION_VENDOR_CLASS (16) Vendor Class		X	ietf-dhcpv6-client.yang
OPTION_VENDOR_OPTS (17) Vendor-specific Information	X	X	ietf-dhcpv6-common.yang
OPTION_INTERFACE_ID (18) Interface-Id Option		X	ietf-dhcpv6-relay.yang
OPTION_RECONF_MSG (19) Reconfigure Message	X		ietf-dhcpv6-server.yang

OPTION_RECONF_ACCEPT (20) Reconfigure Accept Option	X	X	ietf-dhcpv6-client.yang
OPTION_INFORMATION_REFRESH_TIME (32) Information Refresh Time	X		ietf-dhcpv6-server.yang
OPTION_SOL_MAX_RT (82) sol max rt	X		ietf-dhcpv6-server.yang
OPTION_INF_MAX_RT (83) inf max rt	X		ietf-dhcpv6-server.yang

Другие определения опций можно добавить в дополнительных модулях YANG с помощью операторов augment для соответствующих модулей элементов из этого документа. В Приложении В представлен пример модуля, показывающий, как можно таким способом расширить определения опций DHCPv6. Даны также некоторые рекомендации по созданию модулей YANG для дополнительных опций DHCPv6.

2. Терминология

Предполагается знакомство читателя с языком моделирования данных YANG [RFC7950].

Модули YANG с этом документе соответствуют архитектуре NMDA [RFC8342]. Символы, применяемые в диаграммах деревьев, описаны в [RFC8340].

Предполагается знакомство читателя с терминологией DHCPv6, заданной в [RFC8415] и других документах.

2.1. Уровни требований

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

3. Диаграммы деревьев DHCPv6

3.1. Диаграмма дерева сервера DHCPv6

Диаграмма дерева на рисунке 1 иллюстрирует модуль сервера DHCPv6, а также включает базовый модуль из параграфа 4.1.

```

module: ietf-dhcpv6-server
  +--rw dhcpv6-server
    +--rw enabled?          boolean
    +--rw server-duid?     dhcp6:duid
    +--rw vendor-config
    +--rw option-sets
      | +--rw option-set* [option-set-id]
      |   +--rw option-set-id          string
      |   +--rw description?          string
      |   +--rw preference-option
      |   | +--rw pref-value?  uint8
      |   +--rw auth-option
      |   | +--rw algorithm?    uint8
      |   | +--rw rdm?          uint8
      |   | +--rw replay-detection?  uint64
      |   | +--rw (protocol)?
      |   |   +--:(conf-token)
      |   |   | +--rw token-auth-information?  binary
      |   |   +--:(rkap)
      |   |   +--rw datatype?      uint8
      |   |   +--rw auth-info-value?  binary
      |   +--rw server-unicast-option
      |   | +--rw server-address?  inet:ipv6-address
      |   +--rw rapid-commit-option!
      |   +--rw vendor-specific-information-options
      |   | +--rw vendor-specific-information-option*
      |   |   [enterprise-number]
      |   |   +--rw enterprise-number  uint32
      |   |   +--rw vendor-option-data* [sub-option-code]
      |   |   +--rw sub-option-code    uint16
      |   |   +--rw sub-option-data?  binary
      |   +--rw reconfigure-message-option
      |   | +--rw msg-type?  uint8
      |   +--rw reconfigure-accept-option!
      |   +--rw info-refresh-time-option
      |   | +--rw info-refresh-time?  dhcp6:timer-seconds32
      |   +--rw sol-max-rt-option
      |   | +--rw sol-max-rt-value?  dhcp6:timer-seconds32
      |   +--rw inf-max-rt-option
      |   | +--rw inf-max-rt-value?  dhcp6:timer-seconds32
      +--rw class-selector
      +--rw allocation-ranges
        +--rw option-set-id*    leafref
        +--rw valid-lifetime?   dhcp6:timer-seconds32
        +--rw renew-time?       dhcp6:timer-seconds32
        +--rw rebind-time?      dhcp6:timer-seconds32
        +--rw preferred-lifetime?  dhcp6:timer-seconds32
        +--rw rapid-commit?     boolean

```

```

+--rw allocation-range* [id]
|   +--rw id string
|   +--rw description? string
|   +--rw network-prefix inet:ipv6-prefix
|   +--rw option-set-id* leafref
|   +--rw valid-lifetime? dhcp6:timer-seconds32
|   +--rw renew-time? dhcp6:timer-seconds32
|   +--rw rebind-time? dhcp6:timer-seconds32
|   +--rw preferred-lifetime? dhcp6:timer-seconds32
|   +--rw rapid-commit? boolean
+--rw address-pools {na-assignment}?
|   +--rw address-pool* [pool-id]
|   |   +--rw pool-id string
|   |   +--rw pool-prefix
|   |   |   inet:ipv6-prefix
|   |   +--rw start-address
|   |   |   inet:ipv6-address-no-zone
|   |   +--rw end-address
|   |   |   inet:ipv6-address-no-zone
|   |   +--rw max-address-utilization? dhcp6:threshold
|   |   +--rw option-set-id* leafref
|   |   +--rw valid-lifetime?
|   |   |   dhcp6:timer-seconds32
|   |   +--rw renew-time?
|   |   |   dhcp6:timer-seconds32
|   |   +--rw rebind-time?
|   |   |   dhcp6:timer-seconds32
|   |   +--rw preferred-lifetime?
|   |   |   dhcp6:timer-seconds32
|   |   +--rw rapid-commit? boolean
|   +--rw host-reservations
|   |   +--rw host-reservation* [reserved-addr]
|   |   |   +--rw client-duid? dhcp6:duid
|   |   |   +--rw reserved-addr
|   |   |   |   inet:ipv6-address
|   |   |   +--rw option-set-id* leafref
|   |   |   +--rw valid-lifetime?
|   |   |   |   dhcp6:timer-seconds32
|   |   |   +--rw renew-time?
|   |   |   |   dhcp6:timer-seconds32
|   |   |   +--rw rebind-time?
|   |   |   |   dhcp6:timer-seconds32
|   |   |   +--rw preferred-lifetime?
|   |   |   |   dhcp6:timer-seconds32
|   |   |   +--rw rapid-commit? boolean
|   +--ro active-leases
|   |   +--ro total-count uint64
|   |   +--ro allocated-count uint64
|   |   +--ro active-lease* [leased-address]
|   |   |   +--ro leased-address
|   |   |   |   inet:ipv6-address
|   |   |   +--ro client-duid? dhcp6:duid
|   |   |   +--ro ia-id uint32
|   |   |   +--ro allocation-time?
|   |   |   |   yang:date-and-time
|   |   |   +--ro last-renew-rebind?
|   |   |   |   yang:date-and-time
|   |   |   +--ro preferred-lifetime?
|   |   |   |   dhcp6:timer-seconds32
|   |   |   +--ro valid-lifetime?
|   |   |   |   dhcp6:timer-seconds32
|   |   |   +--ro lease-t1?
|   |   |   |   dhcp6:timer-seconds32
|   |   |   +--ro lease-t2?
|   |   |   |   dhcp6:timer-seconds32
|   |   |   +--ro status
|   |   |   |   +--ro code? uint16
|   |   |   |   +--ro message? string
+--rw prefix-pools {prefix-delegation}?
|   +--rw prefix-pool* [pool-id]
|   |   +--rw pool-id string
|   |   +--rw pool-prefix
|   |   |   inet:ipv6-prefix
|   |   +--rw client-prefix-length uint8
|   |   +--rw max-pd-space-utilization? dhcp6:threshold
|   |   +--rw option-set-id* leafref
|   |   +--rw valid-lifetime?
|   |   |   dhcp6:timer-seconds32
|   |   +--rw renew-time?
|   |   |   dhcp6:timer-seconds32
|   |   +--rw rebind-time?
|   |   |   dhcp6:timer-seconds32
|   |   +--rw preferred-lifetime?
|   |   |   dhcp6:timer-seconds32
|   |   +--rw rapid-commit? boolean
|   +--rw host-reservations

```

```

| | | +--rw prefix-reservation* [reserved-prefix]
| | | | +--rw client-duid?          dhcp6:duid
| | | | +--rw reserved-prefix
| | | | |   inet:ipv6-prefix
| | | | | +--rw reserved-prefix-len?  uint8
| | | | +--rw option-set-id*         leafref
| | | | +--rw valid-lifetime?
| | | | |   dhcp6:timer-seconds32
| | | | +--rw renew-time?
| | | | |   dhcp6:timer-seconds32
| | | | +--rw rebind-time?
| | | | |   dhcp6:timer-seconds32
| | | | +--rw preferred-lifetime?
| | | | |   dhcp6:timer-seconds32
| | | | +--rw rapid-commit?         boolean
| | +--ro active-leases
| | | +--ro total-count            uint64
| | | +--ro allocated-count       uint64
| | | +--ro active-lease* [leased-prefix]
| | | | +--ro leased-prefix
| | | | |   inet:ipv6-prefix
| | | | | +--ro client-duid?        dhcp6:duid
| | | | | +--ro ia-id                uint32
| | | | | +--ro allocation-time?
| | | | | |   yang:date-and-time
| | | | | +--ro last-renew-rebind?
| | | | | |   yang:date-and-time
| | | | | +--ro preferred-lifetime?
| | | | | |   dhcp6:timer-seconds32
| | | | | +--ro valid-lifetime?
| | | | | |   dhcp6:timer-seconds32
| | | | | +--ro lease-t1?
| | | | | |   dhcp6:timer-seconds32
| | | | | +--ro lease-t2?
| | | | | |   dhcp6:timer-seconds32
| | | | | +--ro status
| | | | | | +--ro code?             uint16
| | | | | | +--ro message?        string
+--rw statistics
| +--rw discontinuity-time?       yang:date-and-time
| +--ro solicit-count?           yang:counter32
| +--ro advertise-count?         yang:counter32
| +--ro request-count?           yang:counter32
| +--ro confirm-count?           yang:counter32
| +--ro renew-count?             yang:counter32
| +--ro rebind-count?           yang:counter32
| +--ro reply-count?             yang:counter32
| +--ro release-count?           yang:counter32
| +--ro decline-count?           yang:counter32
| +--ro reconfigure-count?       yang:counter32
| +--ro information-request-count? yang:counter32
| +--ro discarded-message-count? yang:counter32

```

rpcs:

```

+---x delete-address-lease {na-assignment}?
| +---w input
| | +---w lease-address-to-delete  leafref
| +--ro output
| | +--ro return-message?  string
+---x delete-prefix-lease {prefix-delegation}?
+---w input
| +---w lease-prefix-to-delete  leafref
+--ro output
| +--ro return-message?  string

```

notifications:

```

+---n address-pool-utilization-threshold-exceeded
| {na-assignment}?
| +--ro pool-id                leafref
| +--ro total-pool-addresses   uint64
| +--ro max-allocated-addresses uint64
| +--ro allocated-address-count uint64
+---n prefix-pool-utilization-threshold-exceeded
| {prefix-delegation}?
| +--ro pool-id                leafref
| +--ro total-pool-prefixes   uint64
| +--ro max-allocated-prefixes uint64
| +--ro allocated-prefixes-count uint64
+---n invalid-client-detected
| +--ro message-type?          enumeration
| +--ro duid?                  dhcp6:duid
| +--ro description?           string
+---n decline-received {na-assignment}?
| +--ro duid?                  dhcp6:duid
| +--ro declined-resources* []
| +--ro (resource-type)?

```

```

|      +--:(declined-address)
|      | +--ro address?   inet:ipv6-address
|      +--:(declined-prefix)
|      | +--ro prefix?   inet:ipv6-prefix
+---n non-success-code-sent
    +--ro duid?         dhcp6:duid
    +--ro status
        +--ro code?      uint16
        +--ro message?   string

```

Рисунок 1. Структура модуля данных сервера DHCPv6.

enabled

Включает или отключает функцию сервера DHCPv6.

dhcpv6-server

Контейнер для конфигурации сервера DHCPv6.

server-duid

Каждый сервер должен иметь уникальный идентификатор (DHCP Unique Identifier или DUID) для представления себя клиентам. DUID состоит из 2-октетного поля типа и поля содержимого размером до 128 октетов. В настоящее время определены 4 типа DUID в [RFC8415] и [RFC6355]. DUID можно настроить с использованием формата одного из этих типов или формата unstructured. Определения типов DUID импортируются из модуля ietf-dhcpv6-common.yang. Ссылки [IANA-HARDWARE-TYPES] и [IANA-PEN] указывают соответствующие типы DUID.

vendor-config

Контейнер для связанных с реализацией узлов YANG для модулей дополнения конфигурации устройства. Пример такого модуля приведён в Приложении C.

option-sets

Сервер можно настроить с множеством наборов опций. Это группы опций DHCPv6 с общими параметрами, которые могут быть предоставлены клиенту по запросу. Поле option-set-id служит для указания набора в конфигурации сервера.

option-set

Конфигурационные параметры для опций DHCPv6. Здесь задан исходный набор определений опций, а дополнительные опции, которые могут относиться также к транслятору или клиенту, импортируются из модуля ietf-dhcpv6-common. При необходимости могут добавляться другие модули опций DHCPv6. Полный список опций DHCPv6 приведён в [IANA-DHCPV6-OPTION-CODES].

class-selector

Служит для размещения зависящих от реализации узлов YANG фирменных для добавляемых селекторов класса. Пример приведён в Приложении D.

allocation-ranges

Для выделения адресов и префиксов применяется иерархическая модель. Контейнер верхнего уровня allocation-ranges содержит глобальные параметры конфигурации. В нем размещается список allocation-range для задания префиксов IPv6 и связанных с ними дополнительных параметров.

address-pools

Применяется для пулов адресов IA_NA (Identity Association for Non-temporary Addresses) и IA_TA (Identity Association for Temporary Addresses) с контейнером для определения резервирований хостов. Здесь же хранится информация о состоянии действующей аренды для каждого пула.

prefix-pools

Указывает пулы, используемые для делегирования префиксов клиентам. Можно также настраивать статическое резервирование для хостов. Делегирование префиксов поддерживают не все реализации серверов DHCPv6, поэтому применяется оператор feature.

Сведения о RPC

delete-address-lease

Позволяет удалить аренду отдельного адреса IPv6 из базы данных сервера. В соответствии с [BCP18] следует по возможности включать в выходное сообщение идентификатор языка.

delete-prefix-lease

Позволяет удалить аренду отдельного префикса IPv6 из базы данных сервера. В соответствии с [BCP18] следует по возможности включать в выходное сообщение идентификатор языка.

Сведения об уведомлениях

address/prefix-pool-utilization-threshold-exceeded

Выдается когда число выделенных из пула адресов или префиксов превышает заданных порог.

invalid-client-detected

Выдается при обнаружении сервером непригодного клиента. Может включать описание ошибки и тип сообщения, вызвавшего генерацию уведомления.

decline-received

Выдается при получении от клиента сообщения DHCPv6 Decline.

non-success-code-sent

Выдается при наличии статусного сообщения для отказа. Коды статуса берутся из [IANA-DHCPV6-STATUS-CODES].

3.2. Диаграмма дерева ретранслятора DHCPv6

Диаграмма дерева на рисунке 2 иллюстрирует модуль сервера DHCPv6, а также включает базовый модуль из параграфа 4.1. RPC в модуле взяты из [RFC8987].

```

module: ietf-dhcpv6-relay
  +--rw dhcpv6-relay
    +--rw enabled?          boolean
    +--rw relay-if* [if-name]
      | +--rw if-name          if:interface-ref
      | +--rw enabled?        boolean
      | +--rw destination-address* inet:ipv6-address

```

```

| +--rw link-address?          inet:ipv6-address
| +--rw relay-options
| | +--rw auth-option
| | | +--rw algorithm?          uint8
| | | +--rw rdm?                uint8
| | | +--rw replay-detection?   uint64
| | | +--rw (protocol)?
| | | | +--:(conf-token)
| | | | | +--rw token-auth-information?  binary
| | | | +--:(rkap)
| | | | +--rw datatype?         uint8
| | | | +--rw auth-info-value?  binary
| | +--rw interface-id-option
| | | +--rw interface-id?      binary
| +--rw statistics
| | +--rw discontinuity-time?
| | | yang:date-and-time
| | +--ro solicit-received-count?
| | | yang:counter32
| | +--ro advertise-sent-count?
| | | yang:counter32
| | +--ro request-received-count?
| | | yang:counter32
| | +--ro confirm-received-count?
| | | yang:counter32
| | +--ro renew-received-count?
| | | yang:counter32
| | +--ro rebind-received-count?
| | | yang:counter32
| | +--ro reply-sent-count?
| | | yang:counter32
| | +--ro release-received-count?
| | | yang:counter32
| | +--ro decline-received-count?
| | | yang:counter32
| | +--ro reconfigure-sent-count?
| | | yang:counter32
| | +--ro information-request-received-count?
| | | yang:counter32
| | +--ro unknown-message-received-count?
| | | yang:counter32
| | +--ro unknown-message-sent-count?
| | | yang:counter32
| | +--ro discarded-message-count?
| | | yang:counter32
| +--rw prefix-delegation! {prefix-delegation}?
| | +--ro pd-leases* [ia-pd-prefix]
| | | +--ro ia-pd-prefix          inet:ipv6-prefix
| | | +--ro last-renew?          yang:date-and-time
| | | +--ro client-peer-address? inet:ipv6-address
| | | +--ro client-duid?         dhcp6:duid
| | | +--ro server-duid?        dhcp6:duid
| +--rw statistics
| | +--ro relay-forward-sent-count?
| | | yang:counter32
| | +--ro relay-forward-received-count?
| | | yang:counter32
| | +--ro relay-reply-received-count?
| | | yang:counter32
| | +--ro relay-forward-unknown-sent-count?
| | | yang:counter32
| | +--ro relay-forward-unknown-received-count?
| | | yang:counter32
| | +--ro discarded-message-count?
| | | yang:counter32

rpcs:
+---x clear-prefix-entry {prefix-delegation}?
| +---w input
| | +---w lease-prefix    leafref
| +--ro output
| | +--ro return-message? string
+---x clear-client-prefixes {prefix-delegation}?
| +---w input
| | +---w client-duid    dhcp6:duid
| +--ro output
| | +--ro return-message? string
+---x clear-interface-prefixes {prefix-delegation}?
+---w input
| +---w interface      -> /dhcpv6-relay/relay-if/if-name
+--ro output
| +--ro return-message? string

notifications:
+---n relay-event
+--ro topology-change

```

```

+--ro relay-if-name?
|
|   -> /dhcpv6-relay/relay-if/if-name
+--ro last-ipv6-addr?  inet:ipv6-address

```

Рисунок 2. Структура модуля данных ретранслятора DHCPv6.

enabled

Включает или отключает функцию ретранслятора DHCPv6.

dhcpv6-relay

Контейнер для конфигурации ретранслятора DHCPv6.

relay-if

Поскольку транслятор может иметь несколько интерфейсов в сторону клиентов, они задаются в списке. Лист if-name служит ключом и имеется interface-ref для подходящего интерфейса, определённого в модуле ietf-interfaces.

enabled

Включает или выключает функции ретранслятора DHCPv6 на конкретном интерфейсе.

destination-addresses

Список индивидуальных и/или групповых адресов IPv6, куда будут транслироваться клиентские сообщения.

link-address

Значение, помещаемое ретранслятором в поле link-address сообщений Relay-Forward.

prefix-delegation

Поскольку делегирование префиксов поддерживают не все реализации ретрансляторов DHCPv6, применяется оператор feature.

pd-leases

Доступные лишь для чтения узлы со сведениями об активных делегированных арендах префиксов.

relay-options

Параметры конфигурации для опций DHCPv6, которые может передавать ретранслятор. Здесь задан исходный набор применимых опций, а дополнительные опции, которые могут относиться также к клиенту и серверу, импортируются из модуля ietf-dhcpv6-common. Информация для Authentication Option (OPTION_AUTH (11)) взята из [IANA-DHCPV6-AUTH-NAMESPACES] и [RFC3118]. При необходимости могут добавляться другие модули опций DHCPv6 с помощью операторов augment. Полный список опций DHCPV6 приведён в [IANA-DHCPV6-OPTION-CODES].

Сведения о RPC

clear-prefix-entry

Позволяет удалить делегированную аренду префикса на ретрансляторе. В соответствии с [BCP18] в выходное сообщение следует по возможности включать идентификатор языка.

clear-client-prefixes

Позволяет удалить делегированную аренду для одного клиента (указан DUID) на ретрансляторе. В соответствии с [BCP18] в выходное сообщение следует по возможности включать идентификатор языка.

clear-interface-prefixes

Позволяет удалить все делегированные записи аренды на интерфейсе ретранслятора. В соответствии с [BCP18] в выходное сообщение следует по возможности включать идентификатор языка.

Сведения об уведомлении

topology-change

Выдается при изменении топологии для агента ретрансляции, т. е. при перенастройке интерфейса в сторону клиентов.

3.3. Диаграмма дерева клиента DHCPv6

Диаграмма дерева на рисунке 3 иллюстрирует модуль сервера DHCPv6, а также включает базовый модуль из параграфа 4.1.

```

module: ietf-dhcpv6-client
+--rw dhcpv6-client
+--rw enabled?      boolean
+--rw client-if* [if-name]
+--rw if-name      if:interface-ref
+--rw enabled?     boolean
+--rw interface-duid?  dhcp6:duid
|   {(non-temp-addr or prefix-delegation or temp-addr)
|   and anon-profile}?
+--rw client-configured-options
| +--rw option-request-option
| | +--rw oro-option*  uint16
| | +--rw rapid-commit-option!
| | +--rw user-class-option!
| | +--rw user-class-data-instance*
| |   [user-class-data-id]
| |   +--rw user-class-data-id  uint8
| |   +--rw user-class-data?   binary
| +--rw vendor-class-option
| | +--rw vendor-class-option-instances*
| |   [enterprise-number]
| |   +--rw enterprise-number  uint32
| |   +--rw vendor-class-data-element*
| |     [vendor-class-data-id]
| |     +--rw vendor-class-data-id  uint8
| |     +--rw vendor-class-data?   binary
| +--rw vendor-specific-information-options
| | +--rw vendor-specific-information-option*
| |   [enterprise-number]
| |   +--rw enterprise-number  uint32
| |   +--rw vendor-option-data* [sub-option-code]

```

```

| | +--rw sub-option-code uint16
| | +--rw sub-option-data? binary
| +--rw reconfigure-accept-option!
+--rw ia-na* [ia-id] {non-temp-addr}?
| +--rw ia-id uint32
| +--rw ia-na-options
| +--ro lease-state
| | +--ro ia-na-address? inet:ipv6-address
| | +--ro lease-t1? dhcp:timer-seconds32
| | +--ro lease-t2? dhcp:timer-seconds32
| | +--ro preferred-lifetime? dhcp:timer-seconds32
| | +--ro valid-lifetime? dhcp:timer-seconds32
| | +--ro allocation-time? yang:date-and-time
| | +--ro last-renew-rebind? yang:date-and-time
| | +--ro server-duid? dhcp:duid
| | +--ro status
| | | +--ro code? uint16
| | | +--ro message? string
+--rw ia-ta* [ia-id] {temp-addr}?
| +--rw ia-id uint32
| +--rw ia-ta-options
| +--ro lease-state
| | +--ro ia-ta-address? inet:ipv6-address
| | +--ro preferred-lifetime? dhcp:timer-seconds32
| | +--ro valid-lifetime? dhcp:timer-seconds32
| | +--ro allocation-time? yang:date-and-time
| | +--ro last-renew-rebind? yang:date-and-time
| | +--ro server-duid? dhcp:duid
| | +--ro status
| | | +--ro code? uint16
| | | +--ro message? string
+--rw ia-pd* [ia-id] {prefix-delegation}?
| +--rw ia-id uint32
| +--rw prefix-length-hint? uint8
| +--rw ia-pd-options
| +--ro lease-state
| | +--ro ia-pd-prefix? inet:ipv6-prefix
| | +--ro lease-t1? dhcp:timer-seconds32
| | +--ro lease-t2? dhcp:timer-seconds32
| | +--ro preferred-lifetime? dhcp:timer-seconds32
| | +--ro valid-lifetime? dhcp:timer-seconds32
| | +--ro allocation-time? yang:date-and-time
| | +--ro last-renew-rebind? yang:date-and-time
| | +--ro server-duid? dhcp:duid
| | +--ro status
| | | +--ro code? uint16
| | | +--ro message? string
+--rw statistics
| +--rw discontinuity-time? yang:date-and-time
| +--ro solicit-count? yang:counter32
| +--ro advertise-count? yang:counter32
| +--ro request-count? yang:counter32
| +--ro confirm-count? yang:counter32
| +--ro renew-count? yang:counter32
| +--ro rebind-count? yang:counter32
| +--ro reply-count? yang:counter32
| +--ro release-count? yang:counter32
| +--ro decline-count? yang:counter32
| +--ro reconfigure-count? yang:counter32
| +--ro information-request-count? yang:counter32
| +--ro discarded-message-count? yang:counter32

```

notifications:

```

+---n invalid-ia-address-detected
| {non-temp-addr or temp-addr}?
| +--ro ia-id uint32
| +--ro ia-na-t1-timer? uint32
| +--ro ia-na-t2-timer? uint32
| +--ro invalid-address? inet:ipv6-address
| +--ro preferred-lifetime? uint32
| +--ro valid-lifetime? uint32
| +--ro ia-options? binary
| +--ro description? string
+---n transmission-failed
| +--ro failure-type enumeration
| +--ro description? string
+---n unsuccessful-status-code
| +--ro server-duid dhcp:duid
| +--ro status
| | +--ro code? uint16
| | +--ro message? string
+---n server-duid-changed
| {non-temp-addr or prefix-delegation or temp-addr}?
| +--ro new-server-duid dhcp:duid
| +--ro previous-server-duid dhcp:duid
| +--ro lease-ia-na?

```

```

|         -> /dhcpv6-client/client-if/ia-na/ia-id
|         {non-temp-addr}?
+---ro lease-ia-ta?
|         -> /dhcpv6-client/client-if/ia-ta/ia-id
|         {temp-addr}?
+---ro lease-ia-pd?
|         -> /dhcpv6-client/client-if/ia-pd/ia-id
|         {prefix-delegation}?

```

Рисунок 3. Структура модуля данных клиента DHCPv6.

enabled

Включает или отключает функцию клиента DHCPv6.

dhcpv6-client

Контейнер для конфигурации клиента DHCPv6.

client-if

Клиент может иметь несколько интерфейсов, запрашивающих конфигурацию через DHCP, они указываются в списке. Лист if-name служит ключом, а interface-ref указывает подходящий интерфейс, заданный в модуле YANG ietf-interfaces.

enabled

Включает или отключает функции клиента DHCPv6 для заданного интерфейса.

client-duid/interface-duid

DUID служит для идентификации клиента сервером или ретранслятором. DUID состоит из 2-октетного поля типа и строки содержимого размером 1-128 октетов. В настоящее время определены 4 типа DUID в [RFC8415] и [RFC6355]. DUID можно настроить с использованием формата одного из этих типов или формата unstructured. Определения типов DUID импортируются из модуля ietf-dhcpv6-common.yang. Ссылки [IANA-HARDWARE-TYPES] и [IANA-PEN] указывают соответствующие типы DUID. настройка DUID для клиента нужна лишь при запросе адресов или префиксов у сервера. Наличие листьев client-duid или interface-duid определяется включением хотя бы одной из функций (feature) non-temp-addr, temp-addr, prefix-delegation. При включении функции anon-profile [RFC7844] можно настроить уникальный DUID на интерфейсе со включённым DHCP, используя лист interface-duid, для иных случаев применяется глобальный лист client-duid.

client-configured-options

Содержит параметры конфигурации для опций DHCPv6, которые может передавать клиент. Начальный набор опций задан здесь, а дополнительные опции, которые могут также относиться к серверу и ретранслятору, импортируются из модуля ietf-dhcpv6-common. При необходимости можно задать дополнительные модули с опциями DHCPv6 с помощью операторов augment. Полный список опций DHCPv6 приведён в [IANA-DHCPV6-OPTION-CODES].

ia-na, ia-ta, ia-pd

Узлы конфигурации для запроса одной или более аренды каждого типа. Доступные лишь для чтения узлы, относящиеся к активной аренде для каждого типа, также размещаются здесь с кодами состояния из [IANA-DHCPV6-STATUS-CODES]. Поскольку эти типы могут поддерживаться не всеми реализациями клиентов DHCPv6, применяются операторы feature. DHCP без поддержки состояния (параграф 6.1 в [RFC8415]) настраивается при отключении всех функций (feature) для адресов и префиксов.

Сведения об уведомлениях

invalid-ia-detected

Выдается при доказанной недействительности отождествления ассоциации клиента. Возможные причины включают дублирование адресов, недействительные адреса и т. п.

retransmission-failed

This is raised when the retransmission mechanism defined in [RFC8415] has failed.

4. Модули YANG DHCPv6

4.1. Базовый модуль YANG DHCPv6

```

<CODE BEGINS> file "ietf-dhcpv6-common@2022-06-20.yang"
module ietf-dhcpv6-common {
  yang-version 1.1;
  namespace "urn:ietf:params:xml:ns:yang:ietf-dhcpv6-common";
  prefix dhc6;

  organization
    "IETF Dynamic Host Configuration (DHC) Working Group";
  contact
    "WG Web: <https://datatracker.ietf.org/wg/dhc/>
    WG List: <mailto:dhcwg@ietf.org>
    Author: Yong Cui <yong@csnet1.cs.tsinghua.edu.cn>
    Author: Linhui Sun <lh.sunlinh@gmail.com>
    Editor: Ian Farrer <ian.farrer@telekom.de>
    Author: Sladjana Zeichlin <sladjana.zechlin@telekom.de>
    Author: Zihao He <hezihao9512@gmail.com>
    Author: Michal Nowikowski <godfryd@isc.org>";
  description
    "Этот модуль YANG задаёт базовые компоненты для настройки и
    управления DHCPv6.

    Ключевые слова ДОЛЖНО, НЕДОПУСТИМО, ТРЕБУЕТСЯ, НУЖНО, НЕ НУЖНО,
    СЛЕДУЕТ, НЕ СЛЕДУЕТ, РЕКОМЕНДУЕТСЯ, НЕ РЕКОМЕНДУЕТСЯ, МОЖНО,
    НЕОБЯЗАТЕЛЬНО в этом документе трактуются в соответствии с
    ВСП 14 (RFC 2119) (RFC 8174) тогда и только тогда, когда они
    указаны заглавными буквами, как показано здесь.

    Авторские права (Copyright (c) 2022) принадлежат IETF Trust и

```

лицам, указанным как авторы. Все права защищены.

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

Эта версия модуля YANG является частью RFC 9243, где правовые аспекты приведены более полно.";

```

revision 2022-06-20 {
  description
    "Исходный выпуск.";
  reference
    "RFC 9243: A YANG Data Model for DHCPv6 Configuration";
}

typedef threshold {
  type uint8 {
    range "1..100";
  }
  description
    "Пороговое значение в процентах.";
}

typedef timer-seconds32 {
  type uint32;
  units "seconds";
  description
    "Значение таймера в секундах (32 бита).";
}

typedef duid-base {
  type string {
    pattern '([0-9a-fA-F]{2}){3,130}';
  }
  description
    "Каждый сервер и клиент DHCP имеет DUID с 2-октетным полем типа
    и полем содержимого от 1 до 128 октетов. Тип duid-base
    применяется другими типами с дополнительными ограничениями.

    В настоящее время определены 4 типа DUID (RFC 8415 и 6355) -
    DUID-LLT, DUID-EN, DUID-LL, DUID-UUID. DUID-unstructured
    представляет DUID, не соответствующие этим форматам.

    Тип string служит для представления шестнадцатеричных
    значений DUID, чтобы можно было применить ограничения.";
  reference
    "RFC 8415: Dynamic Host Configuration Protocol for
    IPv6 (DHCPv6), Section 11
    RFC 6355: Definition of the UUID-Based DHCPv6 Unique
    Identifier (DUID-UUID), Section 4";
}

typedef duid-llt {
  type duid-base {
    pattern '0001'
      + '[0-9a-fA-F]{12,}';
  }
  description
    "DUID типа 1 на основе адреса канального уровня и времени
    (Link-Layer Address Plus Time или DUID-LLT). Состоит из 2
    октетов типа оборудования, выделенных IANA, 4 октетов времени
    создания DUID (число секунд 1 1 января 2000 г UTC по модулю
    2^32) и адреса канального уровня без разделителей, например,

    | 0001 | 0006 | 28490058 | 00005E005300 |

    Здесь указан тип 1 DUID (0x01), тип оборудования 0x06 (IEEE
    Hardware Type), время создания 0x28490058 (см. выше) и адрес
    канального уровня 0x5E005300 (EUI-48 00-00-5E-00-53-00).";
  reference
    "RFC 8415: Dynamic Host Configuration Protocol for
    IPv6 (DHCPv6), Section 11.2
    IANA 'Hardware Types' registry
    <https://www.iana.org/assignments/arp-parameters>";
}

typedef duid-en {
  type duid-base {
    pattern '0002'
      + '[0-9a-fA-F]{8,}';
  }
  description
    "DUID типа 2 на основе Enterprise Number (DUID-EN)

```

производителя. Состоит из 4-октетов Private Enterprise Number, выделенного IANA, и выделенного производителем значения. Например,
| 0002 | 00007ED9 | 0CC084D303000912 |

Здесь указано 2 октета типа DUID (0x02), 4 октета Enterprise Number (0x7ED9) и 8 октетов идентификатора (0x0CC084D303000912).";

reference

"RFC 8415: Dynamic Host Configuration Protocol for IPv6 (DHCPv6), Section 11.3
IANA 'Private Enterprise Numbers' registry
<<https://www.iana.org/assignments/enterprise-numbers>>"

}

```
typedef duid-ll {
```

```
  type duid-base {
```

```
    pattern '0003'
```

```
      + '([0-9a-fA-F]){4,}';
```

```
  }
```

```
  description
```

"DUID типа 3 на основе адреса канального уровня (DUID-LL). Состоит из 2 октетов типа оборудования, выделенных IANA, и адреса канального уровня без разделителей. Например,
| 0003 | 0006 | 00005E005300 |

Этот пример включает 2 октета типа DUID (0x03), тип оборудования 0x06 (IEEE Hardware Type) и адрес канального уровня 0x5E005300 (EUI-48 00-00-5E-00-53-00).";

reference

"RFC 8415: Dynamic Host Configuration Protocol for IPv6 (DHCPv6), Section 11.4
IANA 'Hardware Types' registry
<<https://www.iana.org/assignments/arp-parameters>>"

}

```
typedef duid-uuid {
```

```
  type duid-base {
```

```
    pattern '0004'
```

```
      + '[0-9a-fA-F]{32}';
```

```
  }
```

```
  description
```

"DUID типа 4 на основе UUID (DUID-UUID). Содержит 16 октетов 128 битового идентификатора UUID. Например,
| 0004 | 9f03b182705747e38a1e422910078642 |

Этот пример содержит 2 октета типа DUID (0x04) и UUID 9f03b182-7057-47e3-8a1e-422910078642.";

reference

"RFC 8415: Dynamic Host Configuration Protocol for IPv6 (DHCPv6), Section 11.5
RFC 6355: Definition of the UUID-Based DHCPv6 Unique Identifier (DUID-UUID)";

}

```
typedef duid-unstructured {
```

```
  type duid-base {
```

```
    pattern '(000[1-4].*)' {
```

```
      modifier "invert-match";
```

```
  }
```

```
}
```

```
description
```

"Применяется для DUID иных форматов (не 1 - 4), например,
| 7b6a164d325946539dc540fb539bc430 |

Здесь применяется произвольное 16-октетное значение. Единственным ограничением является отличие первых 2 октетов от 0x01-0x04 во избежание путаницы с типами (duid-llt, duid-en, duid-ll, duid-uuid).";

reference

"RFC 8415: Dynamic Host Configuration Protocol for IPv6 (DHCPv6), Section 11";

}

```
typedef duid {
```

```
  type union {
```

```
    type duid-llt;
```

```
    type duid-en;
```

```
    type duid-ll;
```

```
    type duid-uuid;
```

```
    type duid-unstructured;
```

```
  }
```

```
description
```

"Представляет неструктурированный идентификатор DUID.";

reference

"RFC 8415: Dynamic Host Configuration Protocol for

```

    IPv6 (DHCPv6), Section 11";
}

/*
 * Группировки
 */
grouping status {
  description
    "Сведения о последнем коде состояния, переданном сервером
    или полученным клиентом.";
  reference
    "RFC 8415: Dynamic Host Configuration Protocol
    for IPv6 (DHCPv6), Section 7.5.";
  container status {
    description
      "Сведения о коде состояния для успеха или отказа операций,
      запрошенных в сообщениях.";
    leaf code {
      type uint16;
      description
        "Числовой код статуса, представленный в опции. Список кодов
        приведён в реестре Status Codes, доступном по ссылке
        <https://www.iana.org/assignments/dhcpv6-parameters>.";
    }
    leaf message {
      type string;
      description
        "Строка UTF-8, пригодная для вывода пользователю.
        НЕДОПУСТИМО завершение null-символом.";
    }
  }
}

grouping auth-option-group {
  description
    "OPTION_AUTH (11) Authentication Option.";
  reference
    "RFC 8415: Dynamic Host Configuration Protocol
    for IPv6 (DHCPv6), Section 21.11
    RFC 3118: Authentication for DHCP Messages
    IANA 'Dynamic Host Configuration Protocol (DHCP)
    Authentication Option Name Spaces' registry
    <https://www.iana.org/assignments/auth-namespaces>";
  container auth-option {
    description
      "OPTION_AUTH (11) Authentication Option.";
    leaf algorithm {
      type uint8;
      description
        "Алгоритм, применяемый для протокола аутентификации.";
    }
    leaf rdm {
      type uint8;
      description
        "Метод обнаружения повторов (Replay Detection Method или
        RDM), применяемый в этой опции Authentication.";
    }
    leaf replay-detection {
      type uint64;
      description
        "Сведения об обнаружении повтора для RDM.";
    }
  }
  choice protocol {
    description
      "Протокол аутентификации, применяемый в опции. Значения
      1 (отложенная аутентификация) и 2 (Delayed Authentication
      (устарело)) не применимы и не моделируются.";
    case conf-token {
      leaf token-auth-information {
        type binary;
        description
          "Значение 0. Сведения для аутентификации, заданные
          протоколом и алгоритмом, применяются в опции
          Authentication.";
      }
    }
    case rkap {
      description
        "Значение 3. Для протокола Reconfigure Key Authentication
        (RKAP) обеспечивается защита от неверной настройки
        клиента, вызванная сообщением Reconfigure от враждебного
        сервера DHCP.";
      leaf datatype {
        type uint8 {
          range "1 .. 2";
        }
      }
    }
  }
}

```

```

        description
            "Тип данных в поле Value этой опции.
            1 Reconfigure key (применяется в сообщении Reply).
            2 дайджест HMAC-MD5 для сообщения (применяется в
              сообщении Reconfigure).";
    }
    leaf auth-info-value {
        type binary {
            length "16";
        }
        description
            "Данные, указанные полем Type (16 октетов).";
    }
}
}
}

grouping rapid-commit-option-group {
    description
        "OPTION_RAPID_COMMIT (14) Rapid Commit Option.";
    reference
        "RFC 8415: Dynamic Host Configuration Protocol for
        IPv6 (DHCPv6), Section 21.14";
    container rapid-commit-option {
        presence "Разрешать передачу этой опции";
        description
            "OPTION_RAPID_COMMIT (14) Rapid Commit Option.";
    }
}

grouping vendor-specific-information-option-group {
    description
        "OPTION_VENDOR_OPTS (17) Vendor-specific Information
        Option.";
    reference
        "RFC 8415: Dynamic Host Configuration Protocol
        for IPv6 (DHCPv6), Section 21.17";
    container vendor-specific-information-options {
        description
            "OPTION_VENDOR_OPTS (17) Vendor-specific Information
            Option.";
        list vendor-specific-information-option {
            key "enterprise-number";
            description
                "Опция Vendor-specific Information может присутствовать
                неоднократно в сообщении. Каждая запись списка задает
                содержимое экземпляра опции.";
            leaf enterprise-number {
                type uint32;
                description
                    "Enterprise Number производителя от IANA.";
                reference
                    "IANA 'Private Enterprise Numbers' registry
                    <https://www.iana.org/assignments/enterprise-numbers>";
            }
            list vendor-option-data {
                key "sub-option-code";
                description
                    "Опции производителя, интерпретируемые фирменными
                    функциями клиента и сервера.";
                leaf sub-option-code {
                    type uint16;
                    description
                        "Код субопции.";
                }
                leaf sub-option-data {
                    type binary;
                    description
                        "Область данных субопции.";
                }
            }
        }
    }
}

grouping reconfigure-accept-option-group {
    description
        "OPTION_RECONF_ACCEPT (20) Reconfigure Accept Option.
        Клиент применяет опцию Reconfigure Ассепт для аносирования
        серверу своего отношения к сообщениям Reconfigure, а сервер -
        для указания клиенту своего отношения к приёму сообщений
        Reconfigure. Без этой опции клиент не принимает сообщения
        Reconfigure по умолчанию. Наличие узла включает опцию.";
    reference
        "RFC 8415: Dynamic Host Configuration Protocol

```

```

    for IPv6 (DHCPv6), Section 21.20";
    container reconfigure-accept-option {
      presence "Разрешает передачу этой опции";
      description
        "OPTION_RECONF_ACCEPT (20) Reconfigure Accept Option.";
    }
  }
}
<CODE ENDS>

```

4.2. Модуль YANG для сервера DHCPv6

Этот модуль импортирует определения типов из [RFC6991] и модуля из [RFC8343].

```

<CODE BEGINS> file "ietf-dhcpv6-server@2022-06-20.yang"
module ietf-dhcpv6-server {
  yang-version 1.1;
  namespace "urn:ietf:params:xml:ns:yang:ietf-dhcpv6-server";
  prefix dhc6-srv;

  import ietf-inet-types {
    prefix inet;
    reference
      "RFC 6991: Common YANG Data Types";
  }
  import ietf-yang-types {
    prefix yang;
    reference
      "RFC 6991: Common YANG Data Types";
  }
  import ietf-dhcpv6-common {
    prefix dhc6;
    reference
      "RFC 9243: A YANG Data Model for DHCPv6 Configuration";
  }
  import ietf-netconf-acm {
    prefix nacm;
    reference
      "RFC 8341: Network Configuration Access Control Model";
  }

  organization
    "IETF Dynamic Host Configuration (DHC) Working Group";
  contact
    "WG Web: <https://datatracker.ietf.org/wg/dhc/>
    WG List: <mailto:dhcwg@ietf.org>
    Author: Yong Cui <yong@csnet1.cs.tsinghua.edu.cn>
    Author: Linhui Sun <lh.sunlinh@gmail.com>
    Editor: Ian Farrer <ian.farrer@telekom.de>
    Author: Sladjana Zeichlin <sladjana.zechlin@telekom.de>
    Author: Zihao He <hezihao9512@gmail.com>
    Author: Michal Nowikowski <godfryd@isc.org>";
  description
    "Этот модуль YANG определяет компоненты настройки и
    управления для серверов DHCPv6.

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

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

    Эта версия модуля YANG является частью RFC 9243, где правовые
    аспекты приведены более полно.";

  revision 2022-06-20 {
    description
      "Исходный выпуск.";
    reference
      "RFC 9243: A YANG Data Model for DHCPv6 Configuration";
  }

  /*
  * Свойства (функции)
  */

  feature na-assignment {
    description
      "Указывает, что сервер DHCPv6 реализует выделение постоянных
      (non-temporary) адресов.";
    reference
      "RFC 8415: Dynamic Host Configuration Protocol for
      IPv6 (DHCPv6), Section 6.2";
  }
}

```

```
feature prefix-delegation {
  description
    "Указывает, что сервер DHCPv6 реализует делегирование
    префиксов.";
  reference
    "RFC 8415: Dynamic Host Configuration Protocol for
    IPv6 (DHCPv6), Section 6.3";
}

/*
 * Группировки
 */

grouping resource-config {
  description
    "Узлы, применяемые на разных уровнях иерархии адресации
    сервера DHCPv6.";
  leaf-list option-set-id {
    type leafref {
      path "/dhcpv6-server/option-sets/option-set/option-set-id";
    }
    description
      "Поле ID для набора опций DHCPv6 (option-set) для
      предоставления клиентам с использованием allocation-range.";
  }
  leaf valid-lifetime {
    type dhcp6:timer-seconds32;
    description
      "Срок действия Identity Association (IA).";
    reference
      "RFC 8415: Dynamic Host Configuration Protocol for
      IPv6 (DHCPv6), Section 12.1";
  }
  leaf renew-time {
    type dhcp6:timer-seconds32;
    description
      "Время обновления (T1).";
    reference
      "RFC 8415: Dynamic Host Configuration Protocol for
      IPv6 (DHCPv6), Section 4.2";
  }
  leaf rebind-time {
    type dhcp6:timer-seconds32;
    description
      "Время перепривязки (T2).";
    reference
      "RFC 8415: Dynamic Host Configuration Protocol for
      IPv6 (DHCPv6), Section 4.2";
  }
  leaf preferred-lifetime {
    type dhcp6:timer-seconds32;
    description
      "Предпочтительный срок действия IA.";
    reference
      "RFC 8415: Dynamic Host Configuration Protocol for
      IPv6 (DHCPv6), Section 12.1";
  }
  leaf rapid-commit {
    type boolean;
    description
      "Разрешает обмен цепочками из 2 сообщений.";
    reference
      "RFC 8415: Dynamic Host Configuration Protocol for
      IPv6 (DHCPv6), Section 5.1";
  }
}

grouping lease-information {
  description
    "Сведения о привязке для каждого клиента, получившего
    адрес или префикс IPv6.";
  leaf client-duid {
    type dhcp6:duid;
    description
      "DUID клиента.";
    reference
      "RFC 8415: Dynamic Host Configuration Protocol for
      IPv6 (DHCPv6), Section 11";
  }
  leaf ia-id {
    type uint32;
    mandatory true;
    description
      "IAID клиента.";
    reference
```

```
"RFC 8415: Dynamic Host Configuration Protocol for
IPv6 (DHCPv6), Section 12";
}
leaf allocation-time {
  type yang:date-and-time;
  description
    "Дата и время предоставления аренды.";
  reference
    "RFC 8415: Dynamic Host Configuration Protocol for
    IPv6 (DHCPv6), Section 18";
}
leaf last-renew-rebind {
  type yang:date-and-time;
  description
    "Время последнего успешного обновление или перепривязки.";
  reference
    "RFC 8415: Dynamic Host Configuration Protocol for
    IPv6 (DHCPv6), Section 18";
}
leaf preferred-lifetime {
  type dhcp6:timer-seconds32;
  description
    "Предпочтительный срок действия в секундах.";
  reference
    "RFC 8415: Dynamic Host Configuration Protocol for
    IPv6 (DHCPv6), Section 6";
}
leaf valid-lifetime {
  type dhcp6:timer-seconds32;
  description
    "Срок действия аренды в секундах.";
  reference
    "RFC 8415: Dynamic Host Configuration Protocol for
    IPv6 (DHCPv6), Section 6";
}
leaf lease-t1 {
  type dhcp6:timer-seconds32;
  description
    "Интервал, после которого клиенту нужно связаться с сервером,
    где были получены адреса из IA_NA, для продления срока
    действия адресов, выделенных для IA_PD.";
  reference
    "RFC 8415: Dynamic Host Configuration Protocol for
    IPv6 (DHCPv6), Section 4.2";
}
leaf lease-t2 {
  type dhcp6:timer-seconds32;
  description
    "Интервал, после которого клиенту нужно связаться с любым
    доступным сервером для продления срока действия адресов,
    выделенных IA_PD.";
  reference
    "RFC 8415: Dynamic Host Configuration Protocol for
    IPv6 (DHCPv6), Section 4.2";
}
}
uses dhcp6:status;
}

grouping message-statistics {
  description
    "Счётчики для сообщений DHCPv6.";
  leaf discontinuity-time {
    type yang:date-and-time;
    description
      "Время, последнего разрыва одного или нескольких счётчиков
      сервера DHCPv6. Если разрывов с последней реинициализации
      локальной подсистемы управления не было, узел содержит
      время реинициализации локальной подсистемы управления.";
  }
  leaf solicit-count {
    type yang:counter32;
    config false;
    description
      "Число принятых сообщений Solicit (1).";
  }
  leaf advertise-count {
    type yang:counter32;
    config false;
    description
      "Число переданных сообщений Advertise (2).";
  }
  leaf request-count {
    type yang:counter32;
    config false;
    description
      "Число принятых сообщений Request (3).";
  }
}
```

```
}
leaf confirm-count {
  type yang:counter32;
  config false;
  description
    "Число принятых сообщений Confirm (4).";
}
leaf renew-count {
  type yang:counter32;
  config false;
  description
    "Число принятых сообщений Renew (5).";
}
leaf rebind-count {
  type yang:counter32;
  config false;
  description
    "Число принятых сообщений Rebind (6).";
}
leaf reply-count {
  type yang:counter32;
  config false;
  description
    "Число переданных сообщений Reply (7).";
}
leaf release-count {
  type yang:counter32;
  config false;
  description
    "Число принятых сообщений Release (8).";
}
leaf decline-count {
  type yang:counter32;
  config false;
  description
    "Число принятых сообщений Decline (9).";
}
leaf reconfigure-count {
  type yang:counter32;
  config false;
  description
    "Число переданных сообщений Reconfigure (10).";
}
leaf information-request-count {
  type yang:counter32;
  config false;
  description
    "Число принятых сообщений Information-request (11).";
}
leaf discarded-message-count {
  type yang:counter32;
  config false;
  description
    "Число сообщений, отброшенных по любым причинам.";
}
}

grouping preference-option-group {
  description
    "OPTION_PREFERENCE (7) Preference Option.";
  reference
    "RFC 8415: Dynamic Host Configuration Protocol for
    IPv6 (DHCPv6), Section 21.8";
  container preference-option {
    description
      "OPTION_PREFERENCE (7) Preference Option.";
    leaf pref-value {
      type uint8;
      description
        "Уровень предпочтения для сервера в этом сообщении.
        1-октетное целое число без знака.";
    }
  }
}

grouping server-unicast-option-group {
  description
    "OPTION_UNICAST (12) Server Unicast Option.";
  reference
    "RFC 8415: Dynamic Host Configuration Protocol for
    IPv6 (DHCPv6), Section 21.12";
  container server-unicast-option {
    description
      "OPTION_UNICAST (12) Server Unicast Option.";
    leaf server-address {
      type inet:ipv6-address;
    }
  }
}
```

```

    description
      "128-битовый адрес, по которому клиенту следует передавать
      сообщения, доставляемые индивидуально (unicast).";
  }
}

grouping reconfigure-message-option-group {
  description
    "OPTION_RECONF_MSG (19) Reconfigure Message Option.";
  reference
    "RFC 8415: Dynamic Host Configuration Protocol for
    IPv6 (DHCPv6), Section 21.19";
  container reconfigure-message-option {
    description
      "OPTION_RECONF_MSG (19) Reconfigure Message Option.";
    leaf msg-type {
      type uint8;
      description
        "5 для Renew, 6 для Rebind, 11 для Information-request.";
    }
  }
}

grouping info-refresh-time-option-group {
  description
    "OPTION_INFORMATION_REFRESH_TIME (32) Information Refresh
    Time Option.";
  reference
    "RFC 8415: Dynamic Host Configuration Protocol for
    IPv6 (DHCPv6), Section 21.23";
  container info-refresh-time-option {
    description
      "OPTION_INFORMATION_REFRESH_TIME (32) Information Refresh
      Time Option.";
    leaf info-refresh-time {
      type dhcp6:timer-seconds32;
      description
        "Верхняя граница времени ожидания клиентом перед
        обновлением сведений, полученных от сервера DHCP.";
    }
  }
}

grouping sol-max-rt-option-group {
  description
    "OPTION_SOL_MAX_RT (82) SOL_MAX_RT Option (тайм-аут
    Max Solicit).";
  reference
    "RFC 8415: Dynamic Host Configuration Protocol for
    IPv6 (DHCPv6), Section 21.24";
  container sol-max-rt-option {
    description
      "OPTION_SOL_MAX_RT (82) SOL_MAX_RT Option.";
    leaf sol-max-rt-value {
      type dhcp6:timer-seconds32;
      description
        "Значение тайм-аута Maximum Solicit.";
    }
  }
}

grouping inf-max-rt-option-group {
  description
    "OPTION_INF_MAX_RT (83) INF_MAX_RT Option (тайм-аут Max
    Information-request).";
  reference
    "RFC 8415: Dynamic Host Configuration Protocol for
    IPv6 (DHCPv6), Section 21.25";
  container inf-max-rt-option {
    description
      "OPTION_INF_MAX_RT (83) INF_MAX_RT Option.";
    leaf inf-max-rt-value {
      type dhcp6:timer-seconds32;
      description
        "Тайм-аут Maximum Information-request.";
    }
  }
}

/*
 * Узлы данных
 */

container dhcpv6-server {
  description

```

```

"Узлы конфигурации для сервера DHCPv6.";
reference
  "RFC 8415: Dynamic Host Configuration Protocol for
  IPv6 (DHCPv6), Section 18.3";
leaf enabled {
  type boolean;
  description
    "Включает функции сервера DHCP.";
}
leaf server-duid {
  type dhcp:duid;
  description
    "DUID of the server.";
  reference
    "RFC 8415: Dynamic Host Configuration Protocol for
    IPv6 (DHCPv6), Section 11";
}
container vendor-config {
  description
    "Контейнер для фирменных дополнений и зависящих от
    реализации узлов данных конфигурации.";
}
container option-sets {
  description
    "Сервер может разрешать настройку различных наборов опций для
    клиентов, соответствующих определенным параметрам, таким как
    топологическое местоположение и тип клиента. Список
    option-set содержит набор опций и содержимое, возвращаемое
    клиентам.";
  reference
    "RFC 8415: Dynamic Host Configuration Protocol for
    IPv6 (DHCPv6), Section 21";
  list option-set {
    key "option-set-id";
    description
      "Определения YANG для опций DHCPv6 содержатся в отдельных
      модулях и добавляются в этот контейнер, когда нужно.";
    leaf option-set-id {
      type string;
      description
        "Идентификатор набора опций.";
    }
    leaf description {
      type string;
      description
        "Необязательное поле для дополнительных сведений,
        относящихся к набору опций.";
    }
    uses preference-option-group;
    uses dhcp:auth-option-group;
    uses server-unicast-option-group;
    uses dhcp:rapid-commit-option-group;
    uses dhcp:vendor-specific-information-option-group;
    uses reconfigure-message-option-group;
    uses dhcp:reconfigure-accept-option-group;
    uses info-refresh-time-option-group;
    uses sol-max-rt-option-group;
    uses inf-max-rt-option-group;
  }
}
container class-selector {
  description
    "Серверы DHCPv6 применяют функцию class-selector для
    идентификации и классификации входящих сообщений от клиентов
    для предоставления тем корректной конфигурации. Механизмы
    реализации этой функции различаются в реализациях, поэтому
    их невозможно включить в этот модуль. Контейнер служит для
    дополнений class-selector YANG от разработчиков.";
}
container allocation-ranges {
  description
    "Эта модель основана на иерархии выделения адресов и
    параметров. Верхний уровень является «глобальным» и задаётся
    как контейнер для всех allocation-ranges. В его иерархии
    размещаются отдельные allocation-ranges.";
  uses resource-config;
  list allocation-range {
    key "id";
    description
      "Диапазоны сетей, указываемые ключом id.";
    leaf id {
      type string;
      mandatory true;
      description
        "Уникальный идентификатор диапазона выделения.";
    }
  }
}

```

```

leaf description {
  type string;
  description
    "Описание диапазона выделения.";
}
leaf network-prefix {
  type inet:ipv6-prefix;
  mandatory true;
  description
    "Префикс сети.";
}
uses resource-config;
container address-pools {
  if-feature "na-assignment";
  description
    "Конфигурация для пулов адресов сервера DHCPv6.";
  list address-pool {
    key "pool-id";
    description
      "Список пулов адресов для выделения клиентам,
      различаемых по pool-id.";
    leaf pool-id {
      type string;
      mandatory true;
      description
        "Уникальный идентификатор пула.";
    }
    leaf pool-prefix {
      type inet:ipv6-prefix;
      mandatory true;
      description
        "Префикс IPv6 для пула, которому следует быть в
        заданном network-prefix, если тот имеется.";
    }
    leaf start-address {
      type inet:ipv6-address-no-zone;
      mandatory true;
      description
        "Начальный адрес пула IPv6.";
    }
    leaf end-address {
      type inet:ipv6-address-no-zone;
      mandatory true;
      description
        "Конечный адрес пула IPv6.";
    }
    leaf max-address-utilization {
      type dhc6:threshold;
      description
        "Максимальное число адресов из пула, которые можно
        выделить одновременно, в процентах от доступного
        числа адресов (end-address - start-address + 1)
        с округлением вверх. Служит для установки значения
        для address-pool-utilization-threshold-exceeded.";
    }
  }
  uses resource-config;
  container host-reservations {
    description
      "Конфигурация резервирований хостов из пула.";
    list host-reservation {
      key "reserved-addr";
      description
        "Список резервирований хостов.";
      leaf client-duid {
        type dhc6:duid;
        description
          "DUID клиента для резервирования.";
      }
      leaf reserved-addr {
        type inet:ipv6-address;
        description
          "Зарезервированный адрес IPv6.";
      }
      uses resource-config;
    }
  }
  container active-leases {
    config false;
    description
      "Состояние для активной аренды клиентов.";
    leaf total-count {
      type uint64;
      mandatory true;
      description
        "Общее число адресов в пуле.";
    }
  }
}

```



```
        + "/active-leases/active-lease/leased-prefix";
    }
    mandatory true;
    description
        "Префикс IPv6 активной аренды, удаляемой с сервера.";
}
}
output {
    leaf return-message {
        type string;
        description
            "Отклик от сервера. По возможности в сообщении следует
            включать идентификатор языка.";
        reference
            "BCP 18 (RFC 2277) IETF Policy on Character Sets
            and Languages, Section 4.2";
    }
}
}
/*
 * Уведомления
 */

notification address-pool-utilization-threshold-exceeded {
    if-feature "na-assignment";
    description
        "Использование пула адресов превышает порог, заданный
        max-address-utilization.";
    leaf pool-id {
        type leafref {
            path "/dhcpv6-server/allocation-ranges/"
                + "allocation-range/address-pools/address-pool"
                + "/pool-id";
        }
        mandatory true;
        description
            "Leafref для пула адресов, связанного с уведомлением.";
    }
    leaf total-pool-addresses {
        type uint64;
        mandatory true;
        description
            "Общее число адресов в пуле (end-address -
            start-address + 1).";
    }
    leaf max-allocated-addresses {
        type uint64;
        mandatory true;
        description
            "Максимальное число адресов, которые можно одновременно
            выделить из пула. Это значение может быть меньше общего
            числа адресов. Рассчитывается как max-address-utilization
            (в процентах) от total-pool-addresses с округлением вверх.";
    }
    leaf allocated-address-count {
        type uint64;
        mandatory true;
        description
            "Общее число адресов, выделенных из пула.";
    }
}

notification prefix-pool-utilization-threshold-exceeded {
    if-feature "prefix-delegation";
    description
        "Передаётся при превышении порога использования пула,
        заданного max-pd-space-utilization.";
    leaf pool-id {
        type leafref {
            path "/dhcpv6-server/allocation-ranges"
                + "/allocation-range/prefix-pools/prefix-pool/pool-id";
        }
        mandatory true;
        description
            "Уникальный идентификатор пула.";
    }
    leaf total-pool-prefixes {
        type uint64;
        mandatory true;
        description
            "Общее число префиксов в пуле.";
    }
    leaf max-allocated-prefixes {
        type uint64;
        mandatory true;
    }
}
```

```

description
  "Максимальное число префиксов, которые можно одновременно
  выделить из пула. Это значение может быть меньше общего
  числа префиксов. Рассчитывается как max-prefix-utilization
  (в процентах) от total-pool-prefixes с округлением вверх.";
}
leaf allocated-prefixes-count {
  type uint64;
  mandatory true;
  description
    "Число выделенных из пула префиксов.";
}
}

notification invalid-client-detected {
  description
    "Передаётся при обнаружении сервером непригодного клиента.";
  leaf message-type {
    type enumeration {
      enum solicit {
        description
          "Сообщение Solicit (1).";
      }
      enum request {
        description
          "Сообщение Request (3).";
      }
      enum confirm {
        description
          "Сообщение Confirm (4).";
      }
      enum renew {
        description
          "Сообщение Renew (5).";
      }
      enum rebind {
        description
          "Сообщение Rebind (6).";
      }
      enum release {
        description
          "Сообщение Release (8).";
      }
      enum decline {
        description
          "Сообщение Decline (9).";
      }
      enum info-request {
        description
          "Сообщение Information request (11).";
      }
    }
  }
  description
    "Тип полученного сервером сообщения, вызвавшего ошибку.";
}
leaf duid {
  type dhcp6:duid;
  description
    "DUID клиента.";
}
leaf description {
  type string;
  description
    "Описание события (например, код ошибки или сообщение
    журнала).";
}
}

notification decline-received {
  if-feature "na-assignment";
  description
    "Передаётся при получении сервером Decline (9) от клиента.";
  leaf duid {
    type dhcp6:duid;
    description
      "DUID клиента.";
  }
  list declined-resources {
    description
      "Список отвергнутых адресов и префиксов.";
    choice resource-type {
      description
        "Тип отвергнутого ресурса.";
      case declined-address {
        leaf address {
          type inet:ipv6-address;

```


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

Эта версия модуля YANG является частью RFC 9243, где правовые аспекты приведены более полно.";

```

revision 2022-06-20 {
  description
    "Исходный выпуск.";
  reference
    "RFC 9243: A YANG Data Model for DHCPv6 Configuration";
}

/*
 * Свойства (функции)
 */

feature prefix-delegation {
  description
    "Включается, если ретранслятор работает как делегирующий
    маршрутизатор для префиксов DHCPv6.";
  reference
    "RFC 8415: Dynamic Host Configuration Protocol for
    IPv6 (DHCPv6), Section 6.3";
}

/*
 * Группировки
 */

grouping pd-lease-state {
  description
    "State data for the relay.";
  list pd-leases {
    key "ia-pd-prefix";
    config false;
    description
      "Сведения об активном делегировании префикса IA_PD.";
    leaf ia-pd-prefix {
      type inet:ipv6-prefix;
      description
        "Делегированный префикс.";
    }
    leaf last-renew {
      type yang:date-and-time;
      description
        "Время последнего обновления делегированного префикса.";
    }
    leaf client-peer-address {
      type inet:ipv6-address;
      description
        "Партнерский адрес арендующего клиента.";
    }
    leaf client-duid {
      type dhcp6:duid;
      description
        "DUID арендующего клиента.";
    }
    leaf server-duid {
      type dhcp6:duid;
      description
        "DUID делегирующего сервера.";
    }
  }
}

grouping message-statistics {
  description
    "Счётчики для разных типов сообщений DHCPv6.";
  leaf discontinuity-time {
    type yang:date-and-time;
    description
      "Время, последнего разрыва одного или нескольких счётчиков
      ретранслятора DHCPv6. Если разрывов после реинициализации
      локальной подсистемы управления не было, узел содержит
      время реинициализации локальной подсистемы управления.";
  }
  leaf solicit-received-count {
    type yang:counter32;
    config false;
    description
      "Число принятых сообщений Solicit (1).";
  }
}

```

```
}
leaf advertise-sent-count {
  type yang:counter32;
  config false;
  description
    "Число переданных сообщений Advertise (2).";
}
leaf request-received-count {
  type yang:counter32;
  config false;
  description
    "Число принятых сообщений Request (3).";
}
leaf confirm-received-count {
  type yang:counter32;
  config false;
  description
    "Число принятых сообщений Confirm (4).";
}
leaf renew-received-count {
  type yang:counter32;
  config false;
  description
    "Число принятых сообщений Renew (5).";
}
leaf rebind-received-count {
  type yang:counter32;
  config false;
  description
    "Число принятых сообщений Rebind (6).";
}
leaf reply-sent-count {
  type yang:counter32;
  config false;
  description
    "Число переданных сообщений Reply (7).";
}
leaf release-received-count {
  type yang:counter32;
  config false;
  description
    "Число принятых сообщений Release (8).";
}
leaf decline-received-count {
  type yang:counter32;
  config false;
  description
    "Число принятых сообщений Decline (9).";
}
leaf reconfigure-sent-count {
  type yang:counter32;
  config false;
  description
    "Число переданных сообщений Reconfigure (10).";
}
leaf information-request-received-count {
  type yang:counter32;
  config false;
  description
    "Число принятых сообщений Information-request (11).";
}
leaf unknown-message-received-count {
  type yang:counter32;
  config false;
  description
    "Число принятых сообщений неизвестного типа.";
}
leaf unknown-message-sent-count {
  type yang:counter32;
  config false;
  description
    "Число переданных сообщений неизвестного типа.";
}
leaf discarded-message-count {
  type yang:counter32;
  config false;
  description
    "Число сообщений, отброшенных по любой причине.";
}
}

grouping global-statistics {
  description
    "Глобальная статистика для устройства.";
  leaf relay-forward-sent-count {
    type yang:counter32;
```

```

    config false;
    description
      "Число переданных сообщений Relay-forward (12).";
  }
  leaf relay-forward-received-count {
    type yang:counter32;
    config false;
    description
      "Число принятых сообщений Relay-forward (12).";
  }
  leaf relay-reply-received-count {
    type yang:counter32;
    config false;
    description
      "Число принятых сообщений Relay-reply (13).";
  }
  leaf relay-forward-unknown-sent-count {
    type yang:counter32;
    config false;
    description
      "Число переданных сообщений Relay-forward (12) с
      сообщением неизвестного типа.";
  }
  leaf relay-forward-unknown-received-count {
    type yang:counter32;
    config false;
    description
      "Число принятых сообщений Relay-forward (12) с
      сообщением неизвестного типа.";
  }
  leaf discarded-message-count {
    type yang:counter32;
    config false;
    description
      "Число сообщений, отброшенных по любой причине.";
  }
}

grouping interface-id-option-group {
  description
    "OPTION_INTERFACE_ID (18) Interface-Id Option.";
  reference
    "RFC 8415: Dynamic Host Configuration Protocol for
    IPv6 (DHCPv6), Section 21.18";
  container interface-id-option {
    description
      "OPTION_INTERFACE_ID (18) Interface-Id Option.";
    leaf interface-id {
      type binary;
      description
        "Необработываемое (opaque) значение произвольного размера,
        создаваемое агентом ретрансляции для идентификации одного
        из интерфейсов этого агента.";
    }
  }
}

/*
 * Узлы данных
 */

container dhcpv6-relay {
  description
    "Контейнер с узлами данных конфигурации для ретранслятора.";
  reference
    "RFC 8415: Dynamic Host Configuration Protocol for
    IPv6 (DHCPv6), Section 19";
  leaf enabled {
    type boolean;
    description
      "Глобально включает функцию ретрансляции DHCP.";
  }
  list relay-if {
    key "if-name";
    description
      "Список интерфейсов, настроенных для ретрансляции DHCPv6.";
    leaf if-name {
      type if:interface-ref;
      description
        "interface-ref для интерфейса ретранслятора.";
    }
  }
  leaf enabled {
    type boolean;
    description
      "Включает функцию ретрансляции DHCP на этом интерфейсе.";
  }
}

```

```
leaf-list destination-address {
  type inet:ipv6-address;
  description
    "На каждом агенте ретрансляции DHCPv6 может быть задан
    список адресов получателей для ретранслируемых сообщений.
    Список может включать индивидуальные, групповые и другие
    действительные адреса.";
}
leaf link-address {
  type inet:ipv6-address;
  description
    "Адрес, который сервер может использовать для идентификации
    канала, где расположен клиент.";
}
container relay-options {
  description
    "Определения опций DHCPv6, которые может передавать
    ретранслятор, добавляются сюда из других модулей YANG.";
  uses dhcp:auth-option-group;
  uses interface-id-option-group;
}
container statistics {
  description
    "Счётчики сообщений DHCPv6 для интерфейса ретранслятора.";
  uses message-statistics;
}
container prefix-delegation {
  if-feature "prefix-delegation";
  presence "Разрешает делегирование префиксов для интерфейса.";
  description
    "Контролирует и хранит сведения о статусе делегирования
    префиксов.";
  uses pd-lease-state;
}
}
container statistics {
  description
    "Глобальные счётчики DHCPv6 для ретранслятора.";
  uses global-statistics;
}
}
/*
 * RPC
 */
rpc clear-prefix-entry {
  nacm:default-deny-all;
  if-feature "prefix-delegation";
  description
    "Очищает запись для активного делегирования префикса на
    ретрансляторе.";
  reference
    "RFC 8987: DHCPv6 Prefix Delegating Relay Requirements,
    Section 4.4";
  input {
    leaf lease-prefix {
      type leafref {
        path "/dhcpv6-relay/relay-if/prefix-delegation"
          + "/pd-leases/ia-pd-prefix";
      }
      mandatory true;
      description
        "Префикс IPv6 активной аренды, удаляемой с ретранслятора.";
    }
  }
  output {
    leaf return-message {
      type string;
      description
        "Отклик от сервера, включающий по возможности идентификатор
        языка.";
      reference
        "BCP 18 (RFC 2277) IETF Policy on Character Sets
        and Languages, Section 4.2";
    }
  }
}
}
rpc clear-client-prefixes {
  nacm:default-deny-all;
  if-feature "prefix-delegation";
  description
    "Очищает все активные записи префиксов для одного клиента.";
  reference
    "RFC 8987: DHCPv6 Prefix Delegating Relay Requirements,
```

```

    Section 4.4";
input {
    leaf client-duid {
        type dhcp6:duid;
        mandatory true;
        description
            "DUID клиента.";
    }
}
output {
    leaf return-message {
        type string;
        description
            "Отклик от сервера, включающий по возможности идентификатор
            языка.";
        reference
            "BCP 18 (RFC 2277) IETF Policy on Character Sets
            and Languages, Section 4.2";
    }
}
}

rpc clear-interface-prefixes {
    nacm:default-deny-all;
    if-feature "prefix-delegation";
    description
        "Очищает все привязки делегированных префиксов на интерфейсе
        ретранслятора.";
    reference
        "RFC 8987: DHCPv6 Prefix Delegating Relay Requirements,
        Section 4.4";
    input {
        leaf interface {
            type leafref {
                path "/dhcpv6-relay/relay-if/if-name";
            }
            mandatory true;
            description
                "Указывает интерфейс ретранслятора, для которого удаляются
                все активные привязки делегированных префиксов.";
        }
    }
    output {
        leaf return-message {
            type string;
            description
                "Отклик от сервера, включающий по возможности идентификатор
                языка.";
            reference
                "BCP 18 (RFC 2277) IETF Policy on Character Sets
                and Languages, Section 4.2";
        }
    }
}

/*
 * Уведомления
 */

notification relay-event {
    description
        "Уведомления о событиях ретранслятора DHCPv6.";
    container topology-change {
        description
            "Передаётся при удалении интерфейса с конфигурацией или
            состоянием DHCPv6 из if:interface-refs.";
        leaf relay-if-name {
            type leafref {
                path "/dhcpv6-relay/relay-if/if-name";
            }
            description
                "Имя удаляемого интерфейса.";
        }
        leaf last-ipv6-addr {
            type inet:ipv6-address;
            description
                "Последний адрес IPv6, настроенный на интерфейсе.";
        }
    }
}
}
<CODE ENDS>

```

4.4. Модуль YANG для клиента DHCPv6

Этот модуль импортирует определения типов из [RFC6991] и модули из [RFC8343].

```

<CODE BEGINS> file "ietf-dhcpv6-client@2022-06-20.yang"
module ietf-dhcpv6-client {
  yang-version 1.1;
  namespace "urn:ietf:params:xml:ns:yang:ietf-dhcpv6-client";
  prefix dhc6-clnt;

  import ietf-inet-types {
    prefix inet;
    reference
      "RFC 6991: Common YANG Data Types";
  }
  import ietf-yang-types {
    prefix yang;
    reference
      "RFC 6991: Common YANG Data Types";
  }
  import ietf-dhcpv6-common {
    prefix dhc6;
    reference
      "RFC 9243: A YANG Data Model for DHCPv6 Configuration";
  }
  import ietf-interfaces {
    prefix if;
    reference
      "RFC 8343: A YANG Data Model for Interface Management";
  }
}

```

organization

"IETF Dynamic Host Configuration (DHC) Working Group";

contact

"WG Web: <<https://datatracker.ietf.org/wg/dhc/>>
 WG List: <<mailto:dhcwg@ietf.org>>
 Author: Yong Cui <yong@csnet1.cs.tsinghua.edu.cn>
 Author: Linhui Sun <lh.sunlinh@gmail.com>
 Editor: Ian Farrer <ian.farrer@telekom.de>
 Author: Sladjana Zeichlin <sladjana.zechlin@telekom.de>
 Author: Zihao He <hezihao9512@gmail.com>
 Author: Michal Nowikowski <godfryd@isc.org>";

description

"Этот модуль YANG определяет компоненты настройки и управления для клиентов DHCPv6.

Ключевые слова ДОЛЖНО, НЕДОПУСТИМО, ТРЕБУЕТСЯ, НУЖНО, НЕ НУЖНО, СЛЕДУЕТ, НЕ СЛЕДУЕТ, РЕКОМЕНДУЕТСЯ, НЕ РЕКОМЕНДУЕТСЯ, МОЖНО, НЕОБЯЗАТЕЛЬНО в этом документе трактуются в соответствии с ВСП 14 (RFC 2119) (RFC 8174) тогда и только тогда, когда они указаны заглавными буквами, как показано здесь.

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

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

Эта версия модуля YANG является частью RFC 9243, где правовые аспекты приведены более полно.";

```

revision 2022-06-20 {
  description
    "Исходный выпуск.";
  reference
    "RFC 9243: A YANG Data Model for DHCPv6 Configuration";
}

```

```

/*
 * Свойства (функции)
 */

```

```

feature non-temp-addr {
  description
    "Клиент поддерживает выделение постоянных адресов DHCPv6.";
  reference
    "RFC 8415: Dynamic Host Configuration Protocol for IPv6 (DHCPv6), Section 6.2";
}

```

```

feature temp-addr {
  description
    "Клиент поддерживает выделение временных адресов DHCPv6.";
  reference
    "RFC 8415: Dynamic Host Configuration Protocol for IPv6 (DHCPv6), Section 6.5";
}

```

```
feature prefix-delegation {
  description
    "Клиент реализует делегирование префиксов DHCPv6.";
  reference
    "RFC 8415: Dynamic Host Configuration Protocol for
    IPv6 (DHCPv6), Section 6.3";
}

feature anon-profile {
  description
    "Клиент поддерживает профили анонимности DHCP.";
  reference
    "RFC 7844: Anonymity Profiles for DHCP Clients";
}

/*
 * Группировки
 */

grouping message-statistics {
  description
    "Счётчики сообщений DHCPv6.";
  leaf discontinuity-time {
    type yang:date-and-time;
    description
      "Время, последнего разрыва одного или нескольких счётчиков
      клиента DHCPv6. Если разрывов после реинициализации
      локальной подсистемы управления не было, узел содержит
      время реинициализации локальной подсистемы управления.";
  }
  leaf solicit-count {
    type yang:counter32;
    config false;
    description
      "Число переданных сообщений Solicit (1).";
  }
  leaf advertise-count {
    type yang:counter32;
    config false;
    description
      "Число принятых сообщений Advertise (2).";
  }
  leaf request-count {
    type yang:counter32;
    config false;
    description
      "Число переданных сообщений Request (3).";
  }
  leaf confirm-count {
    type yang:counter32;
    config false;
    description
      "Число переданных сообщений Confirm (4).";
  }
  leaf renew-count {
    type yang:counter32;
    config false;
    description
      "Число переданных сообщений Renew (5).";
  }
  leaf rebind-count {
    type yang:counter32;
    config false;
    description
      "Число переданных сообщений Rebind (6).";
  }
  leaf reply-count {
    type yang:counter32;
    config false;
    description
      "Число принятых сообщений Reply (7).";
  }
  leaf release-count {
    type yang:counter32;
    config false;
    description
      "Число переданных сообщений Release (8).";
  }
  leaf decline-count {
    type yang:counter32;
    config false;
    description
      "Число переданных сообщений Decline (9).";
  }
  leaf reconfigure-count {
```

```
    type yang:counter32;
    config false;
    description
      "Число принятых сообщений Reconfigure (10).";
  }
  leaf information-request-count {
    type yang:counter32;
    config false;
    description
      "Число переданных сообщений Information-request (11).";
  }
  leaf discarded-message-count {
    type yang:counter32;
    config false;
    description
      "Число сообщений, отброшенных по любой причине.";
  }
}

grouping lease-state {
  description
    "Сведения об активной аренде IA_NA.";
  leaf preferred-lifetime {
    type dhcp6:timer-seconds32;
    description
      "Предпочтительный срок аренды адреса в секундах.";
  }
  leaf valid-lifetime {
    type dhcp6:timer-seconds32;
    description
      "Действительный срок аренды адреса в секундах.";
  }
  leaf allocation-time {
    type yang:date-and-time;
    description
      "Дата и время первой аренды адреса.";
  }
  leaf last-renew-rebind {
    type yang:date-and-time;
    description
      "Время последнего обновления или перепривязки
      арендованного адреса.";
  }
  leaf server-duid {
    type dhcp6:duid;
    description
      "DUID сервера аренды.";
  }
  uses dhcp6:status;
}

grouping option-request-option-group {
  description
    "OPTION_ORO (6) Option Request Option. Клиент ДОЛЖЕН включать
    опцию Option Request в сообщения Solicit, Request, Renew,
    Rebind, Information-request для информирования сервера об
    опциях, которые он хочет получать от сервера.";
  reference
    "RFC 8415: Dynamic Host Configuration Protocol for
    IPv6 (DHCPv6), Sections 21.23, 21.24, 21.25, & 21.7";
  container option-request-option {
    description
      "OPTION_ORO (6) Option Request Option.";
    leaf-list oro-option {
      type uint16;
      description
        "Список запрошенных клиентом опций, указанный кодом опции.
        Список ДОЛЖЕН включать код для опции SOL_MAX_RT (82) при
        включении в сообщения Solicit. Если эта опция передана в
        сообщении Information-request, ДОЛЖНЫ включаться коды
        OPTION_INFORMATION_REFRESH_TIME (32) и INF_MAX_RT (83).";
    }
  }
}

grouping user-class-option-group {
  description
    "OPTION_USER_CLASS (15) User Class Option";
  reference
    "RFC 8415: Dynamic Host Configuration Protocol for
    IPv6 (DHCPv6), Section 21.15";
  container user-class-option {
    presence "Настройка опции";
    description
      "OPTION_USER_CLASS (15) User Class Option.";
    list user-class-data-instance {
```

```

    key "user-class-data-id";
    min-elements 1;
    description
      "Классы пользователей, в которые входит клиент.";
    leaf user-class-data-id {
      type uint8;
      description
        "Идентификатор данных класса пользователей.";
    }
    leaf user-class-data {
      type binary;
      description
        "Необрабатываемое (opaque) поле, представляющее
        User Class, к которому относится пользователь.";
    }
  }
}

grouping vendor-class-option-group {
  description
    "OPTION_VENDOR_CLASS (16) Vendor Class Option.";
  reference
    "RFC 8415: Dynamic Host Configuration Protocol
    for IPv6 (DHCPv6), Section 21.16";
  container vendor-class-option {
    description
      "OPTION_VENDOR_CLASS (16) Vendor Class Option.";
    list vendor-class-option-instances {
      key "enterprise-number";
      description
        "Опция Vendor Class может указываться в сообщении не один
        раз. Каждая запись указывает содержимое 1 экземпляра.";
      leaf enterprise-number {
        type uint32;
        description
          "Enterprise Number производителя от IANA.";
      }
      list vendor-class-data-element {
        key "vendor-class-data-id";
        description
          "Классы производителя, в которые входит клиент.";
        leaf vendor-class-data-id {
          type uint8;
          description
            "Идентификатор данных Vendor Class.";
        }
        leaf vendor-class-data {
          type binary;
          description
            "Необрабатываемое (opaque) поле, представляющее
            Vendor Class, к которому относится пользователь.";
        }
      }
    }
  }
}

/*
 * Узлы данных
 */
container dhcpv6-client {
  description
    "Конфигурация и состояние клиента DHCPv6.";
  leaf enabled {
    type boolean;
    default "true";
    description
      "Глобально включает функции клиента DHCP.";
  }
  leaf client-duid {
    if-feature "(non-temp-addr or prefix-delegation "
      + "or temp-addr) and not anon-profile";
    type dhcp6:duid;
    description
      "DUID клиента, который будет применяться на всех
      интерфейсах с поддержкой DHCPv6.";
    reference
      "RFC 8415: Dynamic Host Configuration Protocol for
      IPv6 (DHCPv6), Section 11";
  }
  list client-if {
    key "if-name";
    description
      "Список интерфейсов, для которых клиент запрашивает
      конфигурацию DHCPv6.";
  }
}

```

```
leaf if-name {
  type if:interface-ref;
  mandatory true;
  description
    "Ссылка на запись интерфейса, к которой относится
    запрошенная конфигурация.";
}
leaf enabled {
  type boolean;
  default "true";
  description
    "Включает функции клиента DHCP для этого интерфейса.";
}
leaf interface-duid {
  if-feature "(non-temp-addr or prefix-delegation "
    + "or temp-addr) and anon-profile";
  type dhcp6:duid;
  description
    "DUID клиента для интерфейсов, использующих профили
    анонимности DHCP.";
  reference
    "RFC 7844: Anonymity Profiles for DHCP Clients,
    Section 3";
}
container client-configured-options {
  description
    "Определения для опций DHCPv6, которые можно передавать
    клиенту. Дополнительные опции могут добавляться сюда из
    других модулей YANG, когда это нужно.";
  uses option-request-option-group;
  uses dhcp6:rapid-commit-option-group;
  uses user-class-option-group;
  uses vendor-class-option-group;
  uses dhcp6:vendor-specific-information-option-group;
  uses dhcp6:reconfigure-accept-option-group;
}
list ia-na {
  if-feature "non-temp-addr";
  key "ia-id";
  description
    "Конфигурация, связанная с IA_NA.";
  reference
    "RFC 8415: Dynamic Host Configuration Protocol
    for IPv6 (DHCPv6), Section 13.1";
  leaf ia-id {
    type uint32;
    description
      "Уникальный идентификатор для IA_NA.";
    reference
      "RFC 8415: Dynamic Host Configuration Protocol
      for IPv6 (DHCPv6), Section 12";
  }
  container ia-na-options {
    description
      "Точка дополнения для опций, которые клиент может
      передавать в поле опций IA_NA в OPTION_IA_NA.";
  }
  container lease-state {
    config false;
    description
      "Сведения об активной аренде IA_NA.";
    leaf ia-na-address {
      type inet:ipv6-address;
      description
        "Арендуемый адрес.";
    }
  }
  leaf lease-t1 {
    type dhcp6:timer-seconds32;
    description
      "Интервал, по истечении которого клиенту следует
      связаться с сервером, где были получены адреса из
      IA_NA, для продления срока их действия.";
  }
  leaf lease-t2 {
    type dhcp6:timer-seconds32;
    description
      "Интервал, по истечении которого клиенту следует
      связаться с любым доступным сервером для продления
      срока действия адресов, выделенных для IA_NA.";
  }
  uses lease-state;
}
}
list ia-ta {
  if-feature "temp-addr";
  key "ia-id";
```

```
description
"Конфигурация, связанная с адресами IA_TA.";
reference
"RFC 8415: Dynamic Host Configuration Protocol for
IPv6 (DHCPv6), Section 13.2";
leaf ia-id {
type uint32;
description
"Уникальный идентификатор IA_TA.";
reference
"RFC 8415: Dynamic Host Configuration Protocol
for IPv6 (DHCPv6), Section 12";
}
container ia-ta-options {
description
"Точка для добавления опций, которые клиент может
передавать в поле опций IA_TA в OPTION_IA_TA.";
}
container lease-state {
config false;
description
"Сведения об активной аренде IA_TA.";
leaf ia-ta-address {
type inet:ipv6-address;
description
"Арендуемый адрес.";
}
uses lease-state;
}
}
list ia-pd {
if-feature "prefix-delegation";
key "ia-id";
description
"Конфигурация для IA_PD.";
reference
"RFC 8415: Dynamic Host Configuration Protocol for
IPv6 (DHCPv6), Section 13.3";
leaf ia-id {
type uint32;
description
"Уникальный идентификатор IA_PD.";
reference
"RFC 8415: Dynamic Host Configuration Protocol
for IPv6 (DHCPv6), Section 12";
}
leaf prefix-length-hint {
type uint8 {
range "1..128";
}
description
"Рекомендуемое значение размера префикса в сообщениях
серверу для указания размера делегируемого префикса.";
reference
"RFC 8415: Dynamic Host Configuration Protocol
for IPv6 (DHCPv6), Section 18.2.1";
}
container ia-pd-options {
description
"Точка для добавления опций, которые клиент может
передавать в поле опций IA_PD в OPTION_IA_TA.";
}
container lease-state {
config false;
description
"Сведения о делегированном префиксе IA_PD.";
leaf ia-pd-prefix {
type inet:ipv6-prefix;
description
"Арендуемый делегированный префикс.";
}
leaf lease-t1 {
type dhcp6:timer-seconds32;
description
"Интервал, по истечении которого клиенту следует
связаться с сервером, где были получены адреса из
IA_NA, для продления срока действия адресов IA_PD.";
}
leaf lease-t2 {
type dhcp6:timer-seconds32;
description
"Интервал, по истечении которого клиенту следует
связаться с любым доступным сервером для продления
срока действия адресов IA_PD.";
}
uses lease-state;
}
```

```
    }
  }
  container statistics {
    description
      "Счётчики сообщений DHCPv6 для клиента.";
    uses message-statistics;
  }
}

/*
 * Уведомления
 */

notification invalid-ia-address-detected {
  if-feature "non-temp-addr or temp-addr";
  description
    "Передаётся, когда адрес, полученный в опции IA, сочтен
    недействительным. Причиной может быть дублирование адресов
    или иной недействительный адрес.";
  reference
    "RFC 8415: Dynamic Host Configuration Protocol for
    IPv6 (DHCPv6), Section 18.2.10.1";
  leaf ia-id {
    type uint32;
    mandatory true;
    description
      "IAID.";
  }
  leaf ia-na-t1-timer {
    type uint32;
    description
      "Значение поля T1 для выделения постоянного адреса
      (OPTION_IA_NA).";
  }
  leaf ia-na-t2-timer {
    type uint32;
    description
      "Значение поля preferred-lifetime для выделения постоянного
      адреса (OPTION_IA_NA).";
  }
  leaf invalid-address {
    type inet:ipv6-address;
    description
      "Адрес IP, сочтённый недействительным.";
  }
  leaf preferred-lifetime {
    type uint32;
    description
      "Значение поля preferred-lifetime в OPTION_IAADDR.";
  }
  leaf valid-lifetime {
    type uint32;
    description
      "Значение поля valid-lifetime в OPTION_IAADDR.";
  }
  leaf ia-options {
    type binary;
    description
      "Копия содержимого поля опций адреса IA.";
  }
  leaf description {
    type string;
    description
      "Описание причины недействительности IA.";
  }
}

notification transmission-failed {
  description
    "Передаётся при отказе передачи или повтора сообщения.";
  reference
    "RFC 8415: Dynamic Host Configuration Protocol for
    IPv6 (DHCPv6), Section 7.6";
  leaf failure-type {
    type enumeration {
      enum solicit-timeout {
        description
          "Превышен тайм-аут Max Solicit (SOL_MAX_RT).";
      }
      enum request-timeout {
        description
          "Превышен тайм-аут Max Request (REQ_MAX_RT).";
      }
      enum request-retries-exceeded {
        description

```

```

    "Превышено число попыток Max Request (REC_MAX_RC).";
  }
  enum confirm-duration-exceeded {
    description
      "Превышена продолжительность Max Confirm (CNF_MAX_RD).";
  }
  enum renew-timeout {
    description
      "Превышен тайм-аут Max Renew (REN_MAX_RT).";
  }
  enum rebind-timeout {
    description
      "Превышен тайм-аут Max Rebind (REB_MAX_RT).";
  }
  enum info-request-timeout {
    description
      "Превышен тайм-аут Max Information-request (INF_MAX_RT)";
  }
  enum release-retries-exceeded {
    description
      "Превышено число попыток Max Release (REL_MAX_RC).";
  }
  enum decline-retries-exceeded {
    description
      "Превышено число попыток Max Decline (DEC_MAX_RT).";
  }
}
mandatory true;
description
  "Описание отказа.";
}
leaf description {
  type string;
  description
    "Сведения об отказе, такие как число попыток и значения
    таймеров.";
}
}

notification unsuccessful-status-code {
  description
    "Передаётся при получении клиентом сообщения с кодом
    неудачи в опции Status Code.";
  reference
    "RFC 8415: Dynamic Host Configuration Protocol for
    IPv6 (DHCPv6), Section 21.13";
  leaf server-duid {
    type dhcp6:duid;
    mandatory true;
    description
      "DUID сервера, передавшего код неудачи.";
  }
  uses dhcp6:status;
}

notification server-duid-changed {
  if-feature "non-temp-addr or prefix-delegation or "
    + "temp-addr";
  description
    "Передаётся при получении клиентом аренды от сервера, DUID
    которого отличается от сохранённого клиентом (например, в
    отклике на сообщение Rebind).";
  reference
    "RFC 8415: Dynamic Host Configuration Protocol for
    IPv6 (DHCPv6), Section 18.2.5";
  leaf new-server-duid {
    type dhcp6:duid;
    mandatory true;
    description
      "DUID нового сервера.";
  }
  leaf previous-server-duid {
    type dhcp6:duid;
    mandatory true;
    description
      "DUID прежнего сервера.";
  }
  leaf lease-ia-na {
    if-feature "non-temp-addr";
    type leafref {
      path "/dhcpv6-client/client-if/ia-na/ia-id";
    }
    description
      "Ссылка на аренду IA_NA.";
  }
  leaf lease-ia-ta {

```

```

    if-feature "temp-addr";
    type leafref {
      path "/dhcpv6-client/client-if/ia-ta/ia-id";
    }
    description
      "Ссылка на аренду IA_TA.";
  }
  leaf lease-ia-pd {
    if-feature "prefix-delegation";
    type leafref {
      path "/dhcpv6-client/client-if/ia-pd/ia-id";
    }
    description
      "Ссылка на аренду IA_PD.";
  }
}
}
<CODE ENDS>

```

5. Вопросы безопасности

Заданные этим документом модули YANG определяют схему для данных, предназначенную для доступа через сеть с использованием протоколов управления, таких как NETCONF [RFC6241] или RESTCONF [RFC8040]. Нижним уровнем NETCONF служит защищённый транспорт с обязательной поддержкой SSH (Secure Shell) [RFC6242]. Нижним уровнем RESTCONF служит протокол HTTPS с обязательной поддержкой защиты на транспортном уровне (TLS) [RFC8446].

Модель доступа к конфигурации сети (NACM – Network Configuration Access Control Model) [RFC8341] обеспечивает возможность разрешить доступ лишь определённым пользователям NETCONF или RESTCONF к заранее заданному подмножеству операций NETCONF или RESTCONF и содержимого.

В модулях данных YANG определено множество узлов данных, которые разрешают запись, создание и удаление (т. е. config true, как принято по умолчанию). Эти узлы могут быть конфиденциальными или уязвимыми в некоторых сетевых средах. Запись в такие узлы (например, edit-config) без должной защиты может негативно влиять на работу сети. Ниже перечислены ветви и узлы ieft-dhcpv6-server.yang, которые могут быть конфиденциальными или уязвимыми.

dhcp6-srv/vendor-config

dhcp6-srv/allocation-ranges

Атаки на службы (Denial-of-Service или DoS), такие как отключения сервера DHCP или удаление конфигурации пула адресов или префиксов.

dhcp6-srv/option-sets

Различные атаки на основе изменения содержимого опций DHCPv6, ведущего к некоторым угрозам безопасности или приватности. Эти опции могут перенаправлять клиентов на серверы, контролируемые злоумышленником, например, путём изменения адреса сервера а DNS в опции DHCP для указания мошеннического сервера.

Ниже перечислены ветви и узлы ieft-dhcpv6-relay.yang, которые могут быть конфиденциальными или уязвимыми.

dhcp6-rly/relay-if

DoS-атаки на основе запрета функций ретрансляции DHCP или смены destination-address не отсутствующий адрес.

dhcp6-rly/relay-if

Изменение destination-address для передачи сообщений мошенническому серверу DHCPv6.

Некоторые из операций RPC в этих модулях YANG могут быть чувствительными или уязвимыми в той или иной сетевой среде. Важно контролировать доступ к таким операциям. Эти RPC используют паст:default-deny-all. Ниже указаны операции модуля ieft-dhcpv6-relay.yang, которые могут быть чувствительными или уязвимыми.

dhcp6-rly/clear-prefix-entry

dhcp6-rly/clear-client-prefixes

dhcp6-rly/clear-interface-prefixes

Удаление или очистка активной аренды адреса или префикса, вызывающее DoS-атаку, поскольку трафик больше не будет маршрутизироваться клиенту.

Злоумышленник, передающий сообщения DHCPv6, которые заставляют сервер отправлять уведомления invalid-client-detected и decline-received, может организовать DoS-атаку. Такую атаку может смягчить отказ клиента NETCONF от подписки на затронутые атакой уведомления.

Некоторые из доступных для чтения узлов в модулях YANG могут быть конфиденциальными или уязвимыми в той или иной сетевой среде. Важно контролировать доступ к таким объектам (например, get, get-config, notification). Ниже перечислены ветви и узлы, которые могут быть конфиденциальными или уязвимыми.

Ниже указаны ветви и узлы, которые можно злонамеренно использовать для отслеживания активности или отпечатков типа устройства на хосте.

dhcp6-srv/allocation-ranges/allocation-range/address-pools/address-pool/active-leases

Информация на сервере о клиентах с активной арендой.

dhcp6-rly/relay-if/prefix-delegation/

Информация на ретрансляторе о клиентах с активной арендой.

Сведения, о заданных для сервера пулах адресов и префиксов могут применяться злоумышленниками для сетевой разведки [RFC7707]. Ниже указаны ветви и узлы данных, которые могут применяться для таких целей.

dhcp6-srv/allocation-ranges/allocation-range/address-pools/address-pool/pool-prefix

Сведения о диапазонах выделяемых клиентам адресов.

dhcp6-srv/allocation-ranges/allocation-range/prefix-pools/prefix-pool/pool-prefix

Сведения о диапазонах выделяемых клиентам префиксов.

В [RFC7844] описаны профили анонимности для клиентов DHCP, которые могут препятствовать отслеживанию клиентов в посещаемой сети. Поддержку этого можно включить реализацией функции anon-profile в модуля клиента.

В [RFC7824] рассмотрены вопросы приватности DHCPv6, которые применимы и здесь.

6. Взаимодействие с IANA

Этот документ регистрирует 4 URI и 4 модуля YANG.

6.1. Регистрация URI

В соответствии с этим документом агентство IANA зарегистрировало 4 URI в субреестре ns реестра IETF XML Registry [RFC3688].

```
URI: urn:ietf:params:xml:ns:yang:ietf-dhcpv6-server
Registrant Contact: The IESG.
XML: N/A; запрошенный URI является пространством имён XML.
```

```
URI: urn:ietf:params:xml:ns:yang:ietf-dhcpv6-relay
Registrant Contact: The IESG.
XML: N/A; запрошенный URI является пространством имён XML.
```

```
URI: urn:ietf:params:xml:ns:yang:ietf-dhcpv6-client
Registrant Contact: The IESG.
XML: N/A; запрошенный URI является пространством имён XML.
```

```
URI: urn:ietf:params:xml:ns:yang:ietf-dhcpv6-common
Registrant Contact: The IESG.
XML: N/A; запрошенный URI является пространством имён XML.
```

6.2. Регистрация модулей YANG

В соответствии с этим документом агентство IANA зарегистрировало 4 модуля YANG в субреестре YANG Module Names [RFC6020] реестра YANG Parameters.

```
name: ietf-dhcpv6-server
namespace: urn:ietf:params:xml:ns:yang:ietf-dhcpv6-server
maintained by IANA: N
prefix: dhc6-srv
reference: RFC 9243
```

```
name: ietf-dhcpv6-relay
namespace: urn:ietf:params:xml:ns:yang:ietf-dhcpv6-relay
maintained by IANA: N
prefix: dhc6-rly
reference: RFC 9243
```

```
name: ietf-dhcpv6-client
namespace: urn:ietf:params:xml:ns:yang:ietf-dhcpv6-client
maintained by IANA: N
prefix: dhc6-clnt
reference: RFC 9243
```

```
name: ietf-dhcpv6-common
namespace: urn:ietf:params:xml:ns:yang:ietf-dhcpv6-common
maintained by IANA: N
prefix: dhc6
reference: RFC 9243
```

7. Литература

7.1. Нормативные документы

- [BCP18] Alvestrand, H., "IETF Policy on Character Sets and Languages", BCP 18, RFC 2277, January 1998. <<https://www.rfc-editor.org/info/bcp18>>
- [IANA-DHCPV6-AUTH-NAMESPACES] IANA, "Dynamic Host Configuration Protocol (DHCP) Authentication Option Name Spaces", <<https://www.iana.org/assignments/auth-namespaces>>.
- [IANA-DHCPV6-OPTION-CODES] IANA, "Option Codes", <<https://www.iana.org/assignments/dhcpv6-parameters>>.
- [IANA-DHCPV6-STATUS-CODES] IANA, "DHCPv6 Status Codes", <<https://www.iana.org/assignments/dhcpv6-parameters>>.
- [IANA-HARDWARE-TYPES] IANA, "Hardware Types", <<https://www.iana.org/assignments/arp-parameters>>.
- [IANA-PEN] IANA, "Private Enterprise Numbers", <<https://www.iana.org/assignments/enterprise-numbers>>.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC3118] Droms, R., Ed. and W. Arbaugh, Ed., "Authentication for DHCP Messages", RFC 3118, DOI 10.17487/RFC3118, June 2001, <<https://www.rfc-editor.org/info/rfc3118>>.
- [RFC3688] Mealling, M., "The IETF XML Registry", BCP 81, RFC 3688, DOI 10.17487/RFC3688, January 2004, <<https://www.rfc-editor.org/info/rfc3688>>.

- [RFC6020] Bjorklund, M., Ed., "YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)", [RFC 6020](#), DOI 10.17487/RFC6020, October 2010, <<https://www.rfc-editor.org/info/rfc6020>>.
- [RFC6241] Enns, R., Ed., Bjorklund, M., Ed., Schoenwaelder, J., Ed., and A. Bierman, Ed., "Network Configuration Protocol (NETCONF)", [RFC 6241](#), DOI 10.17487/RFC6241, June 2011, <<https://www.rfc-editor.org/info/rfc6241>>.
- [RFC6242] Wasserman, M., "Using the NETCONF Protocol over Secure Shell (SSH)", [RFC 6242](#), DOI 10.17487/RFC6242, June 2011, <<https://www.rfc-editor.org/info/rfc6242>>.
- [RFC6355] Narten, T. and J. Johnson, "Definition of the UUID-Based DHCPv6 Unique Identifier (DUID-UUID)", RFC 6355, DOI 10.17487/RFC6355, August 2011, <<https://www.rfc-editor.org/info/rfc6355>>.
- [RFC6991] Schoenwaelder, J., Ed., "Common YANG Data Types", [RFC 6991](#), DOI 10.17487/RFC6991, July 2013, <<https://www.rfc-editor.org/info/rfc6991>>.
- [RFC7844] Huitema, C., Mrugalski, T., and S. Krishnan, "Anonymity Profiles for DHCP Clients", RFC 7844, DOI 10.17487/RFC7844, May 2016, <<https://www.rfc-editor.org/info/rfc7844>>.
- [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>>.
- [RFC8040] Bierman, A., Bjorklund, M., and K. Watsen, "RESTCONF Protocol", [RFC 8040](#), DOI 10.17487/RFC8040, January 2017, <<https://www.rfc-editor.org/info/rfc8040>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, [RFC 8174](#), DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.
- [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>>.
- [RFC8341] Bierman, A. and M. Bjorklund, "Network Configuration Access Control Model", STD 91, [RFC 8341](#), DOI 10.17487/RFC8341, March 2018, <<https://www.rfc-editor.org/info/rfc8341>>.
- [RFC8342] Bjorklund, M., Schoenwaelder, J., Shafer, P., Watsen, K., and R. Wilton, "Network Management Datastore Architecture (NMDA)", [RFC 8342](#), DOI 10.17487/RFC8342, March 2018, <<https://www.rfc-editor.org/info/rfc8342>>.
- [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>>.
- [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>>.
- [RFC8446] Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.3", [RFC 8446](#), DOI 10.17487/RFC8446, August 2018, <<https://www.rfc-editor.org/info/rfc8446>>.
- [RFC8987] Farrer, I., Kottapalli, N., Hunek, M., and R. Patterson, "DHCPv6 Prefix Delegating Relay Requirements", RFC 8987, DOI 10.17487/RFC8987, February 2021, <<https://www.rfc-editor.org/info/rfc8987>>.

7.2. Дополнительная литература

- [GROUPINGS-TLS] Watsen, K., "YANG Groupings for TLS Clients and TLS Servers", Work in Progress, Internet-Draft, draft-ietf-netconf-tls-client-server-28, 24 May 2022, <<https://datatracker.ietf.org/doc/html/draft-ietf-netconf-tls-client-server-28>>.
- [RFC3319] Schulzrinne, H. and B. Volz, "Dynamic Host Configuration Protocol (DHCPv6) Options for Session Initiation Protocol (SIP) Servers", RFC 3319, DOI 10.17487/RFC3319, July 2003, <<https://www.rfc-editor.org/info/rfc3319>>.
- [RFC7707] Gont, F. and T. Chown, "Network Reconnaissance in IPv6 Networks", RFC 7707, DOI 10.17487/RFC7707, March 2016, <<https://www.rfc-editor.org/info/rfc7707>>.
- [RFC7824] Krishnan, S., Mrugalski, T., and S. Jiang, "Privacy Considerations for DHCPv6", RFC 7824, DOI 10.17487/RFC7824, May 2016, <<https://www.rfc-editor.org/info/rfc7824>>.

Приложение А. Примеры деревьев данных

В этом приложении приведены примеры XML для деревьев данных разных элементов DHCPv6.

А.1. Примеры конфигурации сервера DHCPv6

Приведённый ниже пример показывает базовую конфигурацию сервера, задавая:

- включение функций сервера DHCP;
- DUID сервера;
- набор опций (id=1) с настройкой опции Solicit Max Retry Timeout (SOL_MAX_RT (82));

```

- один диапазон сети (2001:db8::/32);
- один пул адресов, с начальным и конечным адресом, таймерами аренды и option-set-id 1 для указанного выше набора опций.
<dhcpv6-server
  xmlns="urn:iETF:params:xml:ns:yang:iETF-dhcpv6-server">
  <enabled>true</enabled>
  <server-duid>000200090CC084D303000912</server-duid>
  <vendor-config/>
  <option-sets>
    <option-set>
      <option-set-id>1</option-set-id>
      <description>Example DHCP option set</description>
      <sol-max-rt-option>
        <sol-max-rt-value>3600</sol-max-rt-value>
      </sol-max-rt-option>
    </option-set>
  </option-sets>
  <class-selector/>
  <allocation-ranges>
    <valid-lifetime>54000</valid-lifetime>
    <renew-time>7200</renew-time>
    <rebind-time>32400</rebind-time>
    <preferred-lifetime>43200</preferred-lifetime>
    <allocation-range>
      <id>1</id>
      <description>example-allocation-range</description>
      <network-prefix>2001:db8::/32</network-prefix>
      <option-set-id>1</option-set-id>
      <address-pools>
        <address-pool>
          <pool-id>1</pool-id>
          <pool-prefix>2001:db8:1:1::/64</pool-prefix>
          <start-address>2001:db8:1:1::1000</start-address>
          <end-address>2001:db8:1:1::2000</end-address>
          <max-address-utilization>50</max-address-utilization>
          <option-set-id>1</option-set-id>
        </address-pool>
      </address-pools>
    </allocation-range>
  </allocation-ranges>
</dhcpv6-server>

```

Рисунок 4. Пример XML для базовой конфигурации сервера.

Следующий фрагмент показывает статическое резервирование хоста в пуле адресов. Таймеры аренды хоста задают значения больше, чем для адресов с динамическим выделением.

```

<address-pools>
  <address-pool>
    <pool-id>1</pool-id>
    <pool-prefix>2001:db8:1:1::/64</pool-prefix>
    <start-address>2001:db8:1:1::1000</start-address>
    <end-address>2001:db8:1:1::2000</end-address>
    <max-address-utilization>50</max-address-utilization>
    <option-set-id>1</option-set-id>
    <host-reservations>
      <host-reservation>
        <reserved-addr>2001:db8:1:1::1001</reserved-addr>
        <client-duid>00052001db81</client-duid>
        <option-set-id>1</option-set-id>
        <valid-lifetime>604800</valid-lifetime>
        <renew-time>86400</renew-time>
        <rebind-time>172800</rebind-time>
        <preferred-lifetime>345600</preferred-lifetime>
      </host-reservation>
    </host-reservations>
  </address-pool>
</address-pools>

```

Рисунок 5. Пример XML для конфигурации сервера с резервированием для хостов.

В следующем фрагменте показан диапазон сети и пул для использования при делегировании префиксов клиентам. В примере каждый клиент получает префикс /56. Для max-pd-space-utilization задано значение 80%, поэтому уведомление prefix-pool-utilization-threshold-exceeded будет передаваться при превышении этого предела.

```

<allocation-ranges>
  <allocation-range>
    <id>1</id>
    <description>prefix-pool-example</description>
    <network-prefix>2001:db8::/32</network-prefix>
    <prefix-pools>
      <valid-lifetime>54000</valid-lifetime>
      <renew-time>7200</renew-time>
      <rebind-time>32400</rebind-time>
      <preferred-lifetime>43200</preferred-lifetime>
      <prefix-pool>

```

```

<pool-id>0</pool-id>
<option-set-id>1</option-set-id>
<pool-prefix>2001:db8:1::/48</pool-prefix>
<client-prefix-length>56</client-prefix-length>
<max-pd-space-utilization>80</max-pd-space-utilization>
</prefix-pool>
</prefix-pools>
</allocation-range>
</allocation-ranges>

```

Рисунок 6. Пример XML для конфигурации сервера с делегированием префиксов.

Следующий фрагмент конфигурации показывает набор опций, который может возвращаться клиентам в зависимости от содержимого полученного запроса DHCP. Идентификатор 1 для набора опций будет указываться в других местах конфигурации (например, в конфигурации пула адресов) как опции, доступные запрашивающим клиентам.

Пример показывает, как определения опций можно расширить с помощью дополнения (augmentat). В этом случае OPTION_SIP_SERVER_D (21) SIP Servers Domain-Name List из примера модуля в приложении В добавляется к набору опций сервера.

```

<option-sets>
  <option-set>
    <option-set-id>1</option-set-id>
    <description>Example DHCP option set</description>
    <vendor-specific-information-options>
      <vendor-specific-information-option>
        <enterprise-number>32473</enterprise-number>
        <vendor-option-data>
          <sub-option-code>01</sub-option-code>
          <sub-option-data>1234abcd</sub-option-data>
        </vendor-option-data>
        <vendor-option-data>
          <sub-option-code>02</sub-option-code>
          <sub-option-data>abcd1234</sub-option-data>
        </vendor-option-data>
      </vendor-specific-information-option>
    </vendor-specific-information-options>
    <sol-max-rt-option>
      <sol-max-rt-value>3600</sol-max-rt-value>
    </sol-max-rt-option>
    <sip-server-domain-name-list-option
      xmlns="https://example.com/ns/example-dhcpv6-opt-sip-serv">
      <sip-server>
        <sip-serv-id>0</sip-serv-id>
        <sip-serv-domain-name>sip1.example.org</sip-serv-domain-name>
      </sip-server>
      <sip-server>
        <sip-serv-id>1</sip-serv-id>
        <sip-serv-domain-name>sip2.example.org</sip-serv-domain-name>
      </sip-server>
    </sip-server-domain-name-list-option>
  </option-set>
</option-sets>

```

Рисунок 7. Пример XML для конфигурации сервера набором опций.

A.2. Пример конфигурации ретранслятора DHCPv6

Следующий пример показывает базовую конфигурацию для одного интерфейса ретранслятора DHCP и его взаимодействие с модулем ietf-interfaces. Показаны два документа XML, один для ietf-interfaces, другой для ietf-dhcpv6-relay, задающие:

- конфигурацию с применением модуля ietf-interfaces, применяемую для настройки интерфейса ретранслятора;
- включение функций ретранслятора DHCP глобально и для соответствующего интерфейса;
- интерфейс с настройкой ретранслятора через interface-ref со ссылкой на модуль ietf-interfaces;
- два целевых адресов для пересылки входящих сообщений DHCP;
- значение link-address для передачи пересылаемых сообщений;
- значение Interface ID Option (OPTION_INTERFACE_ID (18)) для включения в пересылаемые сообщения.

```

<interfaces xmlns="urn:ietf:params:xml:ns:yang:ietf-interfaces"
  xmlns:ianaift="urn:ietf:params:xml:ns:yang:iana-ift">
  <interface>
    <name>eth0</name>
    <type>ianaift:ethernetCsmacd</type>
    <description>DHCPv6 Relay Interface</description>
    <enabled>true</enabled>
  </interface>
</interfaces>

<dhcpv6-relay xmlns="urn:ietf:params:xml:ns:yang:ietf-dhcpv6-relay">
  <enabled>true</enabled>
  <relay-if>
    <if-name>eth0</if-name>

```

```

<enabled>true</enabled>
<destination-address>2001:db8:2::1</destination-address>
<destination-address>2001:db8:2::2</destination-address>
<link-address>2001:db8:3::1</link-address>
<relay-options>
  <interface-id-option>
    <interface-id>EXAMPLEINTERFACEID01</interface-id>
  </interface-id-option>
</relay-options>
</relay-if>
</dhcpv6-relay>

```

Рисунок 8. Пример XML для базовой конфигурации ретранслятора.

А.3. Пример конфигурации клиента DHCPv6

Следующий пример показывает базовую конфигурацию клиента DHCP и его взаимодействие с модулем ietf-interfaces. Показаны два документа XML, один для ietf-interfaces, другой для ietf-dhcpv6-client, задающие:

- конфигурацию с применением модуля ietf-interfaces, применяемую для настройки интерфейса клиента;
- включение функций клиента DHCP глобально и для соответствующего интерфейса;
- интерфейс с настройкой клиента через interface-ref со ссылкой на модуль ietf-interfaces;
- DUID для интерфейса с DHCPv6;
- список кодов опций, которые клиент будет запрашивать в своих Option Request Option (OPTION_ORO (6));
- 1 экземпляр Vendor-specific Information Option (OPTION_VENDOR_OPTS (17)) с 1 элементом данных субопции;
- запрос постоянного адреса IPv6 (IA_NA) с идентификатором интерфейса ассоциации 1;
- запрос делегирования префикса IPv6 (IA_PD) с идентификатором интерфейса ассоциации 2.

```

<interfaces xmlns="urn:ietf:params:xml:ns:yang:ietf-interfaces"
  xmlns:ianaift="urn:ietf:params:xml:ns:yang:iana-if-type">
  <interface>
    <name>eth0</name>
    <type>ianaift:ethernetCsmacd</type>
    <description>DHCPv6 Relay Interface</description>
    <enabled>true</enabled>
  </interface>
</interfaces>

<dhcpv6-client
  xmlns="urn:ietf:params:xml:ns:yang:ietf-dhcpv6-client">
  <enabled>true</enabled>
  <client-if>
    <if-name>eth0</if-name>
    <enabled>true</enabled>
    <interface-duid>000200090CC084D303000913</interface-duid>
    <client-configured-options>
      <option-request-option>
        <oro-option>17</oro-option>
        <oro-option>23</oro-option>
        <oro-option>24</oro-option>
        <oro-option>82</oro-option>
      </option-request-option>
      <vendor-specific-information-options>
        <vendor-specific-information-option>
          <enterprise-number>32473</enterprise-number>
          <vendor-option-data>
            <sub-option-code>1</sub-option-code>
            <sub-option-data>abcd1234</sub-option-data>
          </vendor-option-data>
        </vendor-specific-information-option>
      </vendor-specific-information-options>
    </client-configured-options>
    <ia-na>
      <ia-id>1</ia-id>
    </ia-na>
    <ia-pd>
      <ia-id>2</ia-id>
    </ia-pd>
  </client-if>
</dhcpv6-client>

```

Рисунок 9. Пример XML для базовой конфигурации клиента.

Приложение В. Пример добавления определений опций DHCPv6

В этом приложении представлен пример расширения определений опций DHCPv6. Предполагается добавление новых опций будущих документах.

Пример задаёт модуль YANG для OPTION_SIP_SERVER_D (21) и OPTION_SIP_SERVER_D (22) из [RFC3319]. Пример конфигурации XML, показывающий взаимодействие с другими модулями, представлен на рисунке 7.

- В имени модуля применено сокращённое имя документа, где задан формат опций DHCP.

- Для определения каждой опции применяется своя группировка.
- Имена опций произведены из зарегистрированных IANA имён путём добавления суффикса -option.
- В полях description указано соответствующее имя и код опции.
- В reference указан номер и название RFC с определением опции DHCPv6.
- Остальные поля соответствуют полям опции DHCP с сохранением порядка из определения. По возможности для полей опции DHCP применяется подходящий тип YANG.
- Для полей, которые могут иметь несколько значений, применяются узлы list или leaf-list.

В группировках для определений опций применяются операторы augment для добавления в соответствующие модули элементов DHCP (server, relay, client).

```

<CODE BEGINS>
module example-dhcpv6-opt-sip-serv {
  yang-version 1.1;
  namespace "https://example.com/ns/"
    + "example-dhcpv6-opt-sip-serv";
  prefix sip-srv;

  import ietf-inet-types {
    prefix inet;
  }
  import ietf-dhcpv6-server {
    prefix dhcp6-srv;
  }

  organization
    "IETF Dynamic Host Configuration (DHC) Working Group";
  contact
    "WG Web: <https://datatracker.ietf.org/wg/dhc/>
    WG List: <mailto:dhcwq@ietf.org>
    Author: Yong Cui <yong@csnet1.cs.tsinghua.edu.cn>
    Author: Linhui Sun <lh.sunlinh@gmail.com>
    Editor: Ian Farrer <ian.farrer@telekom.de>
    Author: Sladjana Zeichlin <sladjana.zechlin@telekom.de>
    Author: Zihao He <hezihao9512@gmail.com>
    Author: Michal Nowikowski <godfryd@isc.org>";
  description
    "Этот модуль YANG содержит опции DHCPv6 из RFC 8415, которые
    могут применяться серверами DHCPv6.

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

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

    Эта версия модуля YANG является частью RFC 9243, где правовые
    аспекты приведены более полно.";

  revision 2022-06-20 {
    description
      "Исходный выпуск.";
    reference
      "RFC 9243: A YANG Data Model for DHCPv6 Configuration";
  }

  /*
  * Группировки
  */

  grouping sip-server-domain-name-list-option-group {
    description
      "OPTION_SIP_SERVER_D (21) SIP Servers Domain-Name List.";
    reference
      "RFC 3319: Dynamic Host Configuration Protocol
      (DHCPv6) Options for Session Initiation Protocol (SIP)
      Servers";
    container sip-server-domain-name-list-option {
      description
        "OPTION_SIP_SERVER_D (21) SIP Servers Domain Name List
        Option.";
      list sip-server {
        key "sip-serv-id";
        description
          "Информация о сервере SIP.";
        leaf sip-serv-id {
          type uint8;
          description
            "Идентификатор списка серверов SIP.";
        }
      }
    }
  }
}

```

```

    }
    leaf sip-serv-domain-name {
        type inet:domain-name;
        description
            "Доменное имя сервера SIP.";
    }
}
}

grouping sip-server-address-list-option-group {
    description
        "OPTION_SIP_SERVER_A (22) SIP Servers IPv6 Address List.";
    reference
        "RFC 3319: Dynamic Host Configuration Protocol
        (DHCPv6) Options for Session Initiation Protocol (SIP)
        Servers";
    container sip-server-address-list-option {
        description
            "OPTION_SIP_SERVER_A (22) SIP Servers IPv6 Address List
            Option.";
        list sip-server {
            key "sip-serv-id";
            description
                "Информация о сервере SIP.";
            leaf sip-serv-id {
                type uint8;
                description
                    "Идентификатор записи списка серверов SIP.";
            }
            leaf sip-serv-addr {
                type inet:ipv6-address;
                description
                    "Адрес IPv6 сервера SIP.";
            }
        }
    }
}

/*
 * Дополнения
 */

augment "/dhcp6-srv:dhcpv6-server/dhcp6-srv:option-sets/"
    + "dhcp6-srv:option-set" {
    description
        "Дополняет группировки определений опций для модуля server.";
    uses sip-server-domain-name-list-option-group;
    uses sip-server-address-list-option-group;
}

<CODE ENDS>

```

Корректное место добавления (augment) новых определений опций будет меняться в зависимости от правил использования конкретной опции. Например, для опций, добавляемых в модуль `ietf-dhcpv6-server`, это будет в большинстве случаев `/dhcp6-srv:dhcp6-srv/dhcp6-srv:option-sets/dhcp6-srv:option-set`, чтобы задавать опции в наборах. Однако имеются опции, применимые лишь в определённых вариантах развёртывания, и в таких случаях может оказаться более логичным добавление в группу, подходящую для опции. Одним из таких примеров является опция `OPTION_PD_EXCLUDE` (67), которая применима лишь вместе с делегированным префиксом, содержащим конкретный префикс. В этом случае для дополнения может лучше подойти `/dhcp6-srv:dhcp6-srv/dhcp6-srv:allocation-ranges/dhcp6-srv:allocation-range/dhcp6-srv:prefix-pools/dhcp6-srv:prefix-pool`.

Приложение С. Пример фирменного модуля конфигурации сервера

В этом приложении показано расширение заданного здесь модуля YANG для сервера фирменными узлами конфигурации, например, для настройки доступа к базе данных об аренде.

В примере определены дополнительные атрибуты сервере, такие как имя и описание. Хранилище записей об аренде задано в контейнере `lease-storage`, поддерживающем три варианта - память (`memfile`), MySQL, PostgreSQL. Для каждого из вариантов представлены параметры конфигурации.

Для простоты в примере предполагается, что сервер DHCPv6 совмещён с сервером MySQL или PostgreSQL и может безопасно обслуживать трафик на локальном хосте без криптографической защиты. В реальной среде эти серверы будут вероятно разнесены и потребуется применять TLS для защиты соединений между сервером баз данных и сервером DHCPv6. Модуль YANG для настройки TLS задан в [GROUPINGS-TLS].

В конце модуля имеется оператор `augment`, добавляющий фирменную конфигурацию из `dhcpv6-server-config:config` в точку монтирования `/dhcpv6-server:config/dhcpv6-server:vendor-config`.

```

<CODE BEGINS>
module example-dhcpv6-server-conf {
    yang-version 1.1;
    namespace "https://example.com/ns/"
        + "example-dhcpv6-server-conf";
    prefix dhcp6-srv-conf;

```

```

import ietf-inet-types {
  prefix inet;
}
import ietf-interfaces {
  prefix if;
}
import ietf-dhcpv6-server {
  prefix dhc6-srv;
}

organization
  "IETF Dynamic Host Configuration (DHC) Working Group";
contact
  "WG Web: <https://datatracker.ietf.org/wg/dhc/>
  WG List: <mailto:dhcwg@ietf.org>
  Author: Yong Cui <yong@csnet1.cs.tsinghua.edu.cn>
  Author: Linhui Sun <lh.sunlinh@gmail.com>
  Editor: Ian Farrer <ian.farrer@telekom.de>
  Author: Sladjana Zeichlin <sladjana.zechlin@telekom.de>
  Author: Zihao He <hezihao9512@gmail.com>
  Author: Michal Nowikowski <godfryd@isc.org>";
description
  "Этот модуль YANG задаёт компоненты для настройки и управления
  зависящей от производителя или развёртывания функциональностью
  сервера DHCPv6. Такие функции будут сильно различаться между
  реализациями, поэтому модуль служить лишь примером.

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

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

  Эта версия модуля YANG является частью RFC 9243, где правовые
  аспекты приведены более полно.";

revision 2022-06-20 {
  description
    "Исходный выпуск.";
  reference
    "RFC 9243: A YANG Data Model for DHCPv6 Configuration";
}

/*
 * Группировки
 */

grouping config {
  description
    "Параметры, требуемые для настройки сервера DHCPv6.";
  container serv-attributes {
    description
      "Базовые атрибуты, требуемые для работы сервера DHCPv6.";
    leaf name {
      type string;
      description
        "Имя сервера DHCPv6.";
    }
    leaf description {
      type string;
      description
        "Описание сервера DHCPv6.";
    }
    leaf ipv6-listen-port {
      type uint16;
      default "547";
      description
        "Порт UDP, прослушиваемый сервером.";
    }
    choice listening-interfaces {
      default "all-interfaces";
      description
        "Интерфейсы или адреса для приёма входящих сообщений.";
      case all-interfaces {
        container all-interfaces {
          presence "true";
          description
            "Настройка прослушивания входящих сообщений на всех
            адресах IPv6 (unicast и multicast) всех интерфейсов.";
        }
      }
    }
    case interface-list {
      leaf-list interfaces {

```

```

    type if:interface-ref;
    description
      "Список интерфейсов, на которых сервер будет слушать
      входящие сообщения. Приниматься будут сообщения
      по любому действительному адресу IPv6.";
  }
}
case address-list {
  leaf-list address-list {
    type inet:ipv6-address;
    description
      "Список адресов IPv6, на которых сервер будет слушать
      входящие сообщения DHCPv6.";
  }
}
leaf-list interfaces-config {
  type if:interface-ref;
  default "if:interfaces/if:interface/if:name";
  description
    "Список интерфейсов, которые серверу следует слушать.";
}
container lease-storage {
  description
    "Способ сохранения сведений об аренде.";
  choice storage-type {
    description
      "Тип хранилища для сведений о аренде.";
    case memfile {
      description
        "Конфигурация для хранения записей в файле CSV.";
      leaf memfile-name {
        type string;
        description
          "Указывает абсолютный путь к файлу хранилища. Формат
          записей следует семантике операционной системы.";
      }
      leaf memfile-lfc-interval {
        type uint64;
        description
          "Интервал очистки сервером файла записей (секунды).";
      }
    }
  }
}
case mysql {
  leaf mysql-name {
    type string;
    description
      "Имя базы данных MySQL на локальном хосте.";
  }
  leaf mysql-username {
    type string;
    description
      "Имя пользователя для доступа к базе данных.";
  }
  leaf mysql-password {
    type string;
    description
      "Пароль пользователя для доступа к базе данных.";
  }
  leaf mysql-port {
    type inet:port-number;
    default "3306";
    description
      "Если база данных размещена в другой системе, можно
      указать номер порта.";
  }
  leaf mysql-lfc-interval {
    type uint64;
    description
      "Интервал очистки сервером базы записей (секунды).";
  }
  leaf mysql-connect-timeout {
    type uint64;
    description
      "Тайм-аут для соединения с базой данных. Для
      удаленной базы тайм-аут можно увеличить.";
  }
}
case postgresql {
  leaf postgresql-name {
    type string;
    description
      "Имя базы данных PostgreSQL на локальном хосте.";
  }
  leaf postgresql-username {
    type string;

```


"WG Web: <<https://datatracker.ietf.org/wg/dhc/>>
 WG List: <<mailto:dhcwg@ietf.org>>
 Author: Yong Cui <yong@csnet1.cs.tsinghua.edu.cn>
 Author: Linhui Sun <lh.sunlinh@gmail.com>
 Editor: Ian Farrer <ian.farrer@telekom.de>
 Author: Sladjana Zeichlin <sladjana.zechlin@telekom.de>
 Author: Zihao He <hezihao9512@gmail.com>
 Author: Michal Nowikowski <godfryd@isc.org>";

description

"Этот модуль YANG задаёт компоненты конфигурации селекторов классов клиентов для сервера DHCPv6. Эта функциональность существенно различается в реализациях и модуль служит лишь примером.

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

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

Эта версия модуля YANG является частью RFC 9243, где правовые аспекты приведены более полно.";

```
revision 2022-06-20 {
  description
    "Исходный выпуск.";
  reference
    "RFC 9243: A YANG Data Model for DHCPv6 Configuration";
}

/*
 * Группировки
 */
grouping client-class-id {
  description
    "Определения классификации сообщений клиентов для
    аутентификации и назначения.";
  leaf client-class-name {
    type string;
    mandatory true;
    description
      "Уникальный идентификатор записей классов клиентов.";
  }
  choice id-type {
    mandatory true;
    description
      "Определения типов идентификаторов класса клиентов.";
    case client-id-id {
      leaf client-id {
        type string;
        mandatory true;
        description
          "Строка идентификатора клиента.";
      }
    }
    description
      "Выбор класса клиента по строковым идентификаторам.";
  }
  case received-interface-id {
    description
      "Выбор класса клиента по входному интерфейсу сообщения
      DHCPv6.";
    leaf received-interface {
      type if:interface-ref;
      description
        "Ссылка на запись для входного интерфейса сообщения
        DHCPv6.";
    }
  }
}
case packet-source-address-id {
  description
    "Выбор класса клиента по адресу отправителя сообщения
    DHCPv6.";
  leaf packet-source-address {
    type inet:ipv6-address;
    mandatory true;
    description
      "Адрес отправителя в сообщении DHCPv6.";
  }
}
case packet-destination-address-id {
  description
    "Выбор класса клиента по адресу получателя сообщения
    DHCPv6.";
```

```
leaf packet-destination-address {
    type inet:ipv6-address;
    mandatory true;
    description
        "Адрес получателя в сообщении DHCPv6.";
}
}
case relay-link-address-id {
    description
        "Выбор класса клиента по префиксу в поле канального адреса
        в заголовке сообщения агента ретрансляции.";
    leaf relay-link-address {
        type inet:ipv6-prefix;
        mandatory true;
        description
            "Префикс в поле link-address заголовка сообщения агента.";
    }
}
case relay-peer-address-id {
    description
        "Выбор класса клиента по полю peer-address в заголовке
        сообщения агента ретрансляции.";
    leaf relay-peer-address {
        type inet:ipv6-prefix;
        mandatory true;
        description
            "Префикс в поле peer-address заголовка сообщения агента";
    }
}
case relay-interface-id {
    description
        "Выбор класса клиента по полученному экземпляру
        OPTION_INTERFACE_ID (18).";
    leaf relay-interface {
        type string;
        description
            "Необработываемое (ораque) значение произвольного размера
            от агента ретрансляции для указания интерфейса агента.";
    }
}
case user-class-option-id {
    description
        "Выбор класса клиента по значению OPTION_USER_CLASS (15) и
        полю user-class-data в этой опции.";
    leaf user-class-data {
        type string;
        mandatory true;
        description
            "Значение User Class для сопоставления.";
    }
}
case vendor-class-present-id {
    description
        "Выбор класса клиента по наличию OPTION_VENDOR_CLASS (16)
        в полученном сообщении.";
    leaf vendor-class-present {
        type boolean;
        mandatory true;
        description
            "OPTION_VENDOR_CLASS (16) присутствует в сообщении.";
    }
}
case vendor-class-option-enterprise-number-id {
    description
        "Выбор класса клиента по значению поля enterprise-number
        в OPTION_VENDOR_CLASS (16).";
    leaf vendor-class-option-enterprise-number {
        type uint32;
        mandatory true;
        description
            "Значение поля enterprise-number.";
    }
}
case vendor-class-option-data {
    description
        "Выбор класса клиента по значению поля данных в записи
        vendor-class-data для сопоставления с полем
        enterprise-number в OPTION_VENDOR_CLASS (16).";
    container vendor-class-option-data {
        description
            "Контейнер данных опции Vendor-class.";
        leaf enterprise-number {
            type uint32;
            description
                "Enterprise Number производителя от IANA.";
        }
    }
}
```



```
}  
<CODE ENDS>
```

Благодарности

Авторы признательны Qi Sun, Lishan Li, Hao Wang, Tomek Mrugalski, Marcin Siodelski, Bernie Volz, Ted Lemon, Bing Liu, Tom Petch, Acee Lindem, Benjamin Kaduk, Kris Lambrechts, Paul Dumitru за важные комментарии и вклад в работу.

Участники работы

Перечисленные ниже люди являются соавторами этого документа.

Yong Cui

Tsinghua University
Beijing,
100084
China
Email: cuiyong@tsinghua.edu.cn

Linhui Sun

Tsinghua University
Beijing,
100084
China
Email: lh.sunlinh@gmail.com

Sladjana Zechlin

Deutsche Telekom AG
CTO-IPT, Landgrabenweg 151

53227, Bonn
Germany
Email: sladjana.zechlin@telekom.de

Zihao He

Tsinghua University
Beijing,
100084
China
Email: hezihao9512@gmail.com

Michal Nowikowski

Internet Systems Consortium
Gdansk
Poland
Email: godfryd@isc.org

Адрес автора

Ian Farrer (editor)
Deutsche Telekom AG
S&TI, Landgrabenweg 151
53227 Bonn
Germany
Email: ian.farrer@telekom.de

Перевод на русский язык

Николай Малых
nmalykh@protokols.ru