

Internet Engineering Task Force (IETF)
Request for Comments: 10031
Category: Standards Track
ISSN: 2070-1721

R. Housley
Vigil Security
C. Bonnell
TurboLight Solutions
J. Mandel
AKAYLA
T. Okubo
Penguin Securities
M. StJohns
NthPermutation Security LLC
August 2026

Media Access Control (MAC) Addresses in X.509 Certificates

MAC-адреса в сертификатах X.509

Аннотация

Этот документ задаёт GeneralName.otherName для включения в расширения X.509 Subject Alternative Name¹ (SAN) и Issuer Alternative Name² (IAN) с целью передачи адресов IEEE Media Access Control³ (MAC). Новая форма имени позволяет связать идентификатор интерфейса канального уровня (L2) с сертификатом открытого ключа. Кроме того, документ задаёт кодирование и обработку ограничений этой формы в расширении имён X.509 (Name Constraints extension или NCE).

Статус документа

Документ относится к категории Internet Standards Track.

Документ является результатом работы IETF⁴ и представляет согласованный взгляд сообщества IETF. Документ прошёл открытое обсуждение и был одобрен для публикации IESG⁵. Дополнительную информацию о стандартах Internet можно найти в разделе 2 в RFC 7841.

Информацию о текущем статусе документа, ошибках и способах обратной связи можно найти по ссылке <https://www.rfc-editor.org/info/rfc10031>.

Авторские права

Авторские права (Copyright (c) 2026) 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 пересмотренной лицензией BSD, как указано в параграфе 4.е документа IETF Trust Legal Provisions, без каких-либо гарантий (как указано в Revised BSD License).

Оглавление

1. Введение.....	2
2. Уровни требований.....	2
3. MACAddress otherName.....	2
3.1. Кодирование MACAddress как Alternative Name.....	2
3.2. Кодирование ограничений MACAddress.....	2
3.3. Правила генерации и проверки.....	2
3.4. Обработка пути для NCE.....	3
3.4.1. Правила сопоставления.....	3
3.4.2. Процедура проверки пути OtherName.MACAddress.....	3
3.4.2.1. Инициализация.....	4
3.4.2.2. Операция пересечения.....	4
3.4.2.3. Операция объединения.....	4
4. Вопросы безопасности.....	5
4.1. Вопросы приватности.....	5
5. Взаимодействие с IANA.....	5
6. Модуль ASN.1.....	5
7. Примеры MAC Address otherName.....	6
7.1. Идентификатор EUI-48.....	6
7.2. Идентификатор EUI-64.....	6
7.3. Ограничение EUI-48 для глобальных индивидуальных адресов.....	6
8. Литература.....	6

¹Дополнительное имя субъекта.

²Дополнительное имя эмитента.

³Управление доступом к среде передачи.

⁴Internet Engineering Task Force - комиссия по решению инженерных задач Internet.

⁵Internet Engineering Steering Group - комиссия по инженерным разработкам Internet.

8.1. Нормативные документы.....	6
8.2. Дополнительная литература.....	6
Благодарности.....	6
Адреса авторов.....	6

1. Введение

При использовании сертификатов X.509 для идентификации устройств по MAC-адресам нужен стандартный способ кодирования этих адресов в расширении SAN, заданном в [RFC5280]. Этот документ определяет новую форму otherName - MACAddress - для передачи 48-битового адреса IEEE 802 MAC (EUI-48) или 64-битового расширенного идентификатора (EUI-64) в строке октетов (OCTET STRING) [X680]. Кроме того, эта форма имени позволяет передавать ограничения для значений EUI-48 или EUI-64 при включении в расширение NCE, заданное в параграфе 4.2.1.10 [RFC5280]. Новая форма имени позволяет выполнять аутентификацию на уровне L2 и упрощает безопасное предоставление услуг IoT¹, в частности, в автомобильных сетях.

Хотя эта конструкция позволяет передавать адреса EUI-48 или EUI-64 в расширении IAN, причин для этого, вероятно, немного или нет совсем.

2. Уровни требований

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

3. MACAddress otherName

В этом документе термины otherName, OtherName и GeneralName.otherName обозначают поле GeneralName.otherName, включаемое в SAN или IAN. Новая форма имени указывается идентификатором объекта (OBJECT IDENTIFIER или OID) id-on-MACAddress (1.3.6.1.5.5.7.8.12) и объявлена ниже с использованием синтаксиса объявления классов OTHER-NAME. Форма имени имеет варианты для передачи EUI-48 как OCTET STRING из 6 октетов и EUI-64 как OCTET STRING из 8 октетов. Ограничения для значений EUI-48 и EUI-64 передаются как OCTET STRING размером вдвое больше размера соответствующего идентификатора. Первые N октетов (N - число октетов адреса) указывает битовый шаблон, которому должен соответствовать идентификатор, вторые N октетов - маску битов для сопоставления.

В последующих параграфах описано кодирование значений EUI-48 и EUI-64, а также соответствующих ограничений.

3.1. Кодирование MACAddress как Alternative Name

Когда такая форма имени включается в расширение SAN или IAN как OtherName, она состоит в точности из 6 или 8 октетов. Значения кодируются от старшего октета к младшему (big-endian или left-to-right). В сертификатах не допускается текстовое представление, поскольку удобочитаемые формы, такие как 00-24-98-7B-19-02 или 0024.987B.1902 применяются лишь в интерфейсах управления. Когда устройство имеет 48-битовый идентификатор MAC, удостоверяющий центр (Certification Authority или CA) **должен** кодировать его в OCTET STRING с 6 октетами как значение MACAddress. Когда заводской идентификатор устройства имеет форму EUI-64 или нет канонического 48-битового адреса, УЦ **должен** кодировать идентификатор в OCTET STRING из 8 октетов как значение MACAddress.

Например, адрес 00-24-98-7B-19-02 кодируется как OCTET STRING '0024987B1902'Н.

3.2. Кодирование ограничений MACAddress

При включении формы имени в NCE применяется OCTET STRING размером вдвое больше размера OCTET STRING идентификатора, для которого задаётся ограничение. OCTET STRING состоит из двух элементов.

1. Первые N октетов (N имеет значение 6 для ограничений EUI-48 и 8 - для ограничений EUI-64) содержат шаблон для сопоставления, указывающий биты, которые должен включать адрес после наложения маски.
2. Вторые N октетов задают битовую маску для ограничения. Установленный в маске бит указывает, что соответствующий бит должен совпадать с битом шаблона (первые N октетов) в той же позиции.

Например, ограничение, задающее имена в рамках уникального идентификатора организации (Organizationally Unique Identifier или OUI) 00-00-5e для адреса EUI-48, будет иметь шаблон '00005E000000'Н и маску 'FFFFFFFF000000'Н, а его представлением будет OCTET STRING '00005E000000FFFFFFFF000000'Н.

Биты шаблона и маски кодируются, начиная со старшего (big-endian или left-to-right).

Если в маске бит не установлен, УЦ **недопустимо** устанавливать соответствующий бит в шаблоне. Это правило гарантирует использование канонического представления для данного шаблона и маски.

В соответствии с параграфом 4.2.1.10 [RFC5280], NCE является действительным и «**должен** применяться только в сертификате УЦ».

3.3. Правила генерации и проверки

УЦ **должен** убедиться, что значения MACAddress otherName в выпускаемых сертификатах принадлежат (или будут принадлежать) устройству-субъекту в течение срока действия сертификата. Один и тот же MAC-адрес **недопустимо** включать в сертификаты разных устройств, если они не имеют общего интерфейса L2.

Проверяющей стороне, которая сопоставляет MAC-адрес с сертификатом, **нужно** выполнять побайтовое сравнение с OCTET STRING. Подстановочные шаблоны (wildcard) не поддерживаются.

¹Internet of Things - Internet вещей.

Самоподписанные сертификаты с MACAddress otherName **должны** включать адрес одного из физических портов устройства.

3.4. Обработка пути для NCE

MACAddress otherName следует общим правилам для ограничений otherName из параграфа 4.2.1.10 в [RFC5280]. NCE **может** вносить разрешённые (permittedSubtrees) и исключённые (excludedSubtrees) поддеревья в OtherName типа id-on-MACAddress.

В псевдокоде ниже сокращение mask применяется для битовой строки маски из ограничения (вторые N битов ограничения, где N имеет значение 6 для EUI-48 и 8 для EUI-64), value указывает битовую строку из первых N битов ограничения. Декларация constraint указывает пару значение-маска ограничения OtherName.MACAddress с полями mask, value и length, где length указывает число байтов полного представления ограничения (12 или 16 в зависимости от типа). Декларация name представляет имя OtherName.MACAddress с полями value и length, где length имеет значение 6 или 8, представляющее размер закодированного имени.

3.4.1. Правила сопоставления

Ниже приведён алгоритм сравнения имени с заданными ограничениями (для использующего сертификат приложения).

1. Если имя содержит 6 октетов (EUI-48), а ограничение - 16 (EUI-64), имя не соответствует ограничению.
2. Если имя содержит 8 октетов (EUI-64), а ограничение - 12 (EUI-48), имя не соответствует ограничению.
3. Извлекается шаблон из старших (big-endian) N октетов ограничения, где N имеет значение 6 для EUI-48 и 8 для EUI-64.
4. Извлекается маска из младших (big-endian) N октетов ограничения, где N имеет значение 6 для EUI-48 и 8 для EUI-64.
5. Выполняется побитовая операция XOR для строки битов шаблона (п. 3) и имени.
6. Выполняется побитовая операция AND для строки битов, полученной в п. 5, и маски.
7. Если п. 6 даёт строку битов, содержащую только нули, имя соответствует ограничению, в ином случае - не соответствует.

Алгоритм можно выразить приведённым ниже псевдокодом

```
// Возвращает true, если name n соответствует constraint c
boolean nameMatchesConstraint (name n, constraint c) {
    return ((2 * n.length) == c.length &&
           ((c.value ^ n.value) &
            c.mask) == 0) ;
}
```

Например, ограничению '000000000000 030000000000'Н будет соответствовать любой адрес universal/unicast EUI-48, такой как 00-00-5e-00-50-34. Ограничению '00005E000000 FFFFFFFF000000'Н будет соответствовать любой адрес universal/unicast с OUI 00-00-5E, например 00-00-5e-00-50-34. Отметим, что OUI 00-00-5E контролирует IANA (параграф 1.3 в [RFC9542]).

Реализации не обязаны применять этот алгоритм, но они **должны** давать для тех же входных данных идентичный результат.

3.4.2. Процедура проверки пути OtherName.MACAddress

В этом параграфе описана процедура проверки пути (Path Validation Processing) для ограничений OtherName.MACAddress. Важно отметить, что можно создавать иерархии NCE для OtherName.MACAddress, которые запретят все имена, даже если это не предусмотрено. Предположим, например, что NCE уровня 1 содержит только permitted_subtrees лишь для OtherName.MACAddress global/unicast EUI-48, а NCE уровня 2 - только permitted_subtrees для любого адреса (т. е. исходный набор ограничений). Это приведёт к пустому набору permitted_subtrees, поскольку ограничение «any address» не входит в global/unicast. Работающий пример оставим читателям.

Ниже представлена вспомогательная функция для определения вхождения набора адресов, соответствующих ограничению MACAddress, в набор адресов, соответствующих другому ограничению. Например, набор для приведённого ниже ограничения child (используется IANA DOI¹) полностью содержится в наборе parent

```
constraint parent = '000000000000 000000000000'Н
constraint child = '00005E000000 FCFFFF000000'Н
```

child является подмножеством parent, поскольку 1) относится к тому же размеру (EUI-48), 2) child.mask AND parent.mask совпадает с маской parent и 3) биты child.value с маской parent.mask совпадают с битами parent.value с той же маской. Отметим, что маска child разрешает любую комбинацию адресов local/universal и unicast/multicast с OUI 00-00-5e.

Если сравнить ограничение child2 = '00005E005000 FFFFFFFF00'Н с child, то child2 будет подмножеством child. Они используют один идентификатор OUI, но child2 ограничивает соответствие адресами universal/unicast за счёт установки битов маски '030000000000'Н, а также сокращения диапазона адресов с 00-00-5E-00-50-00 до 00-00-5E-00-50-FF (диапазон примера для 00-00-5E OUI).

```
// child и parent - ограничения OtherName.MACAddress
// Возвращает true, если все адреса, соответствующие child,
// соответствуют и parent, иначе - false. Применяется при расчете
// пересекающихся наборов для ограничений OtherName.MACAddress.
boolean childIsSubsetOfParent (constraint c, constraint p) {
    return (
        // если размеры совпадают
        c.length == p.length &&
```

¹Digital Object Identifier - идентификатор цифрового объекта. Прим. перев.

```
// и в маске parent не установлены биты, которые не установлены
// и в маске child (например, можно добавить биты маски к
// текущему набору, но нельзя удалить их)
(c.mask & p.mask) == p.mask &&
// и в значении child установлены все биты, которые установлены
// (и действуют) в значении parent (например, нельзя изменить
// значения действующих битов из-за более сильного ограничения)
(c.value & p.mask) == (p.value & p.mask)
};
}
```

3.4.2.1. Инициализация

В соответствии с пп. (h) и (i) параграфа 6.1.1 в [RFC5280] нужно задать значения NCE OtherName.MACAddress как для initial-permitted-subtrees, так и для initial-excluded-subtrees. Для initial-permitted-subtree первым ограничением является восприятие всех EUI-48 MACAddress, а вторым восприятие всех адресов EUI-64 MACAddress:

```
initial-permitted-subtrees{} += { 00000000000000000000000000000000H,
                                   00000000000000000000000000000000H }
initial-excluded-subtrees{} += { };
```

3.4.2.2. Операция пересечения

См. п. (g) (1) в параграфе 6.1.4 [RFC5280]. При перемещении вниз по дереву от корня набор permitted_subtrees может сохраняться или сокращаться. На каждом уровне очищается набор permitted_subtrees для каждого ограничения NCE OtherName.MACAddress.permitted_subtree в сертификате и проверяется наличие на предыдущем уровне ограничения permitted_subtree, которое включает это новое ограничение или совпадает с ним. При наличии такого ограничения оно добавляется в набор permitted_subtrees текущего уровня. Процедура повторяется при перемещении по дереву для всех оставшихся сертификатов УЦ. Операция пересечения текущих permitted_subtrees для OtherName.MACAddress с каждым сертификатом на пути показана ниже.

```
// Эта логика подходит для типов OtherName MACAddress и IPAddress
// Инициализация
permitted_subtrees{} (0) = initial-permitted-subtrees;

// для каждого сертификата в пути выполняются операции {
set constraint prevSubtrees{} =
{ установка OtherName.MACAddress.permitted_subtrees
из переменной permitted_subtree{} (i-1) };
constraint tempPermittedSubtrees {} = {};
constraint tempRequestedSubtrees {} =
{ установка OtherName.MACAddress.permitted_subtrees из
расширения NCE в текущем сертификате };

// rst => одно из запрошенных субдеревьев (из сертификата)
// pst -> одно из текущих разрешённых субдеревьев
foreach ( constraint rst в tempRequestedSubtrees) {
    foreach ( constraint pst в prevSubtrees) {
        if (childIsSubsetOfParent (rst, pst)) {
            tempPermittedSubtrees += rst;
            break;
        }
    }
}

permitted_subtrees{} (i) = tempPermittedSubtrees;
// } для каждого сертификата УЦ в пути
```

3.4.2.3. Операция объединения

См. п. (g) (2) в параграфе 6.1.4 [RFC5280]. В отличие от permitted_subtrees (пересечение NCE на каждом уровне), excluded_subtrees - это объединение всех ограничений. Начиная с пустого excluded_subtrees, каждый уровень добавляет в этот набор все ограничения из сертификатов УЦ, которые ещё не установлены или не входят в установленные ограничения. Процедура объединения excluded_subtrees с OtherName.MACAddress excluded_subtrees из каждого сертификата в пути показана ниже.

```
// Инициализация
excluded_subtrees{} (0) = initial-excluded-subtrees;

// для каждого сертификата в пути выполняются операции {
// процедура объединения начинается того, что было исключено
// на предыдущем уровне и пытается добавить новые ограничения.
tempExcludedSubtrees {} =
{ установка OtherName.MACAddress.excluded_subtrees из
excluded_subtrees (i-1) };
tempRequestedSubtrees {} =
{ установка OtherName.MACAddress.excluded_subtrees из
текущего сертификата };

// порядок цикла отличается от операции пересечения.
foreach (constraint rExcl в tempRequestedSubtrees) {
    boolean matches = false;
    foreach (constraint est в tempExcludedSubtrees) {
        // если текущие ограничения excluded охватывают
        // запрошенное дерево, его не нужно добавлять
        // в excluded_subtrees.
        if (childIsSubsetOfParent (rExcl, est)) {
            matches = true;
        }
    }
}
```

```

        break;
    }
}
if (!matches) {
    tempExcludedSubtrees += rExcl;
}
}
// } для каждого сертификата УЦ в пути
excluded_subtrees{} (i) = tempExcludedSubtrees;

```

4. Вопросы безопасности

Привязка MAC-адреса к сертификату является столь же строгой, как процесс проверки (validation) УЦ. Удостоверяющие центры **должны** убедиться, что абонент (subscriber) правомочно контролирует заявленный MAC-адрес или владеет им. В процессе проверки **должна** учитываться возможность подмены MAC-адресов.

Некоторые системы динамически назначают MAC-адреса или обобществляют их. Такая практика может подорвать уникальность и учитываемость, которые являются целью для этой формы имен.

В отличие от адресов IP, MAC-адреса обычно не маршрутизируются через границы L3 (сетевой уровень). Проверяющим сторонам **недопустимо** полагаться на уникальность адресов за пределами их локальной сети, если нет достоверных сведений о стабильности адресов за пределами сети.

Раздел «Вопросы безопасности» из [RFC5280] применим и к данной спецификации.

4.1. Вопросы приватности

MAC-адрес может однозначно указывать физическое устройство и, тем самым, его пользователя. Сертификаты со встроенными неизменяемыми MAC-адресами упощают долгосрочное отслеживание устройств. В системах с использованием MACAddress **следует** рассмотреть вопрос ротации адресов, используя краткосрочные сертификаты или случайные MAC-адреса (когда это возможно).

5. Взаимодействие с IANA

Агентство IANA выделило показанное в таблице 1 значение в реестре SMI Security for PKIX Module Identifier (1.3.6.1.5.5.7.0)

Таблица 1.

Десятичный код	Описание	Документ
126	id-mod-mac-address-other-name-2025	RFC 10031

Агентство IANA выделило показанное в таблице 2 значение в реестре SMI Security for PKIX Other Name Forms (1.3.6.1.5.5.7.8)

Таблица 2.

Десятичный код	Описание	Документ
12	id-on-MACAddress	RFC 10031

6. Модуль ASN.1

Ниже представлен модуль ASN.1 для MAC-адреса в соответствии с соглашениями [RFC5912].

```

<CODE BEGINS>
MACAddressOtherName-2025
{ iso(1) identified-organization(3) dod(6) internet(1)
  security(5) mechanisms(5) pkix(7) id-mod(0)
  id-mod-mac-address-other-name-2025(126) }

DEFINITIONS IMPLICIT TAGS ::=
BEGIN

IMPORTS
  OTHER-NAME FROM PKIX1Implicit-2009
  { iso(1) identified-organization(3) dod(6) internet(1)
    security(5) mechanisms(5) pkix(7) id-mod(0)
    id-mod-pkix1-implicit-02(59) }

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

-- id-pkix 8 – ребро для otherName
id-on OBJECT IDENTIFIER ::= { id-pkix 8 }

-- OID для этой формы имени
id-on-MACAddress OBJECT IDENTIFIER ::= { id-on 12 }

-- Содержимое поля otherName
MACAddressOtherNames OTHER-NAME ::= { on-MACAddress, ... }

on-MACAddress OTHER-NAME ::= {
  MACAddress IDENTIFIED BY id-on-MACAddress }

MACAddress ::= OCTET STRING (SIZE (6 | 8 | 12 | 16))

END
<CODE ENDS>

```


7. Примеры MAC Address otherName

7.1. Идентификатор EUI-48

Ниже приведено понятное человеку представление расширения Subject Alternative Name из сертификата, содержащего одно имя MACAddress otherName со значением 00-24-98-7B-19-02.

```
SEQUENCE {
  otherName [0] {
    OBJECT IDENTIFIER id-on-MACAddress
    [0] OCTET STRING '0024987B1902'
  }
}
```

7.2. Идентификатор EUI-64

Пример EUI-64 (AC-DE-48-00-11-22-33-44): [0] OCTET STRING 'ACDE480011223344'H

7.3. Ограничение EUI-48 для глобальных индивидуальных адресов

Первый октет MAC-адреса содержит два бита флагов. В нумерации IEEE бит 0 является младшим битом октета, поскольку он передаётся первым.

- Флаг «индивидуальный-групповой» (Individual - I/Group - G) - (бит 0 или маска 0x01): 0 = unicast, 1 = multicast. Групповые префиксы не являются OUI.
- Флаг «глобальный-локальный» (Universal - U/Local - L) - (бит 1 или маска 0x02): 0 = глобальный (выданный IEEE), 1 = локальный.

Эти флаги позволяют реализации исключить групповые и локальные адреса, но не могут служить доказательством того, что 24-битовое значение является зарегистрированным в IEEE OUI. 36-битовые идентификаторы компаний (Company ID или CID) имеют такие же первые 24 бита и организации **могут** задействовать псевдо-OUI. УЦ **должны** включать только адреса, которые абонент правомерно контролирует (зарегистрированные OUI или CID). Перед выдачей сертификата с MACAddress или ограничением имени по разрешённому набору адресов УЦ **должен** проверить контроль (владение) для адреса, например, обратившись к реестру IEEE [IEEE802.1] или документации производителя.

Приведённый ниже пример ограничивает значения EUI-48 глобальными индивидуальными адресами, а локально назначенные и групповые адреса не будут соответствовать ограничению.

```
[0] OCTET STRING '000000000000 030000000000'H
```

8. Литература

8.1. Нормативные документы

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, [RFC 2119](https://www.rfc-editor.org/info/rfc2119), DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC5280] Cooper, D., Santesson, S., Farrell, S., Boeyen, S., Housley, R., and W. Polk, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 5280, DOI 10.17487/RFC5280, May 2008, <<https://www.rfc-editor.org/info/rfc5280>>.
- [RFC5912] Hoffman, P. and J. Schaad, "New ASN.1 Modules for the Public Key Infrastructure Using X.509 (PKIX)", RFC 5912, DOI 10.17487/RFC5912, June 2010, <<https://www.rfc-editor.org/info/rfc5912>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, [RFC 8174](https://www.rfc-editor.org/info/rfc8174), DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.
- [X680] ITU-T, "Information Technology -- Abstract Syntax Notation One (ASN.1): Specification of basic notation", ITU-T Recommendation X.680, ISO/IEC 8824-1:2021, February 2021, <<https://www.itu.int/rec/T-REC-X.680>>.

8.2. Дополнительная литература

- [IEEE802.1] IEEE Standards Association, "Guidelines for Use of Extended Unique Identifier (EUI), Organizationally Unique Identifier (OUI), and Company ID (CID)", 3 August 2017, <<https://standards.ieee.org/wp-content/uploads/import/documents/tutorials/eui.pdf>>.
- [RFC9542] Eastlake 3rd, D., Abley, J., and Y. Li, "IANA Considerations and IETF Protocol and Documentation Usage for IEEE 802 Parameters", BCP 141, [RFC 9542](https://www.rfc-editor.org/info/rfc9542), DOI 10.17487/RFC9542, April 2024, <<https://www.rfc-editor.org/info/rfc9542>>.

Благодарности

Спасибо участникам почтовой конференции рабочей группы LAMPS за их полезные отклики и комментарии. Авторы выражают искреннюю признательность Bob Beck, David von Oheimb, Deb Cooley, Francois Rousseau, Jacqueline McCall, John Preuß Mattsson, Mahesh Jethanandani, Mohamed Boucadair, Murray Kucherawy, Sean Turner, Tim Hollebeek за их рецензии и предложения, которые существенно повысили качество этого документа.

Адреса авторов

Russ Housley
Vigil Security, LLC
Email: housley@vigilsec.com

Corey Bonnell
TurboLight Solutions, LLC

Email: corey.bonnell@turbolightsolutions.com

Joe Mandel

AKAYLA, Inc.

Email: joe@akayla.com

Tomofumi Okubo

Penguin Securities Pte. Ltd.

Email: tomofumi.okubo+ietf@gmail.com

Michael StJohns

NthPermutation Security LLC

Email: msj@nthpermutation.com

Перевод на русский язык

Николай Малых

nmalykh@protokols.ru