

Network Coding for Content-Centric Networking / Named Data Networking: Considerations and Challenges

Сетевое кодирование для CCNx и NDN - соображения и проблемы

Аннотация

Этот документ описывает текущие результаты исследований сетевого кодирования (Network Coding или NC) для ориентированных на содержимое сетей (Content-Centric Networking или CCNx) и сетей именованных данных (Named Data Networking или NDN) и разъясняет технические вопросы и возможные проблемы при использовании NC в CCNx/NDN. Документ является результатом работы исследовательских групп Coding for Efficient Network Communications (NWCRG) и Information-Centric Networking (ICNRG).

Статус документа

Документ не относится к категории Internet Standards Track и публикуется лишь для информации.

Документ является результатом работы IRTF¹. IRTF публикует результаты относящихся к Internet исследований и разработок. Эти результаты могут оказаться не пригодными для реализации. Данный RFC представляет согласованное мнение исследовательской группы Coding for Efficient Network Communications в рамках IRTF. Документы, одобренные для публикации IRSG, не претендуют на статус Internet Standard (см. раздел 2 в RFC 7841).

Информацию о текущем статусе документа, ошибках и способах обратной связи можно найти по ссылке <https://www.rfc-editor.org/info/rfc9273>.

Авторские права

Авторские права (Copyright (c) 2022) 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.

Оглавление

1. Введение.....	2
2. Терминология.....	2
2.1. Определения, связанные с NC.....	2
2.2. Определения, связанные с CCNx/NDN.....	3
3. Основы CCNx/NDN.....	3
4. Основы NC.....	4
5. Преимущества NC и CCNx/NDN.....	4
6. Технические соображения.....	4
6.1. Именованное содержимое.....	4
6.1.1. Уникальные имена для пакетов NC.....	4
6.1.2. Неуникальные имена для пакетов NC.....	5
6.2. Транспорт.....	5
6.2.1. Область действия NC.....	5
6.2.2. Работа потребителя.....	5
6.2.3. Работа пересылающего.....	5
6.2.4. Работа издателя.....	6
6.2.5. Совместимость с прежними версиями.....	6
6.3. Кэширование в сети.....	6
6.4. Беспрепятственная мобильность потребителей.....	7
7. Проблемы.....	7
7.1. Адаптация свёрточного кодирования.....	7
7.2. Контроль скорости и перегрузки.....	7
7.3. Безопасность.....	7
7.4. Расширяемость маршрутизации.....	8
8. Взаимодействие с IANA.....	8
9. Вопросы безопасности.....	8
10. Литература.....	8
Благодарности.....	10
Адреса авторов.....	10

¹Internet Research Task Force - комиссия по исследовательским задачам Internet.

1. Введение

Информационно-ориентированные сети (Information-Centric Networking или ICN), в целом, ориентированные на содержимое сети (Content-Centric Networking или CCNx) [1] или сети именованных данных (Named Data Networking или NDN) [2], в частности, появились как новая парадигма коммуникаций, основанная на извлечении данных по их именам. Эта парадигма продвигает сведения о содержимом на сетевой уровень. Предполагается, что это позволит потребителям получать нужные сведения (содержимое) простым и эффективным способом из разнородных сетей, к которым они могут быть подключены. Архитектура CCNx/NDN представила новые идеи и стимулировала исследования в различных областях, таких как кэширование в сети, маршрутизация по именам, транспортировка по нескольким путям, защита содержимого. Одним из важных преимуществ запроса содержимого по именам устраняет необходимость организации соединения между клиентом и конкретным сервером, а содержимое может быть получено из нескольких источников.

Одновременно с этим в академических и промышленных кругах рос интерес к улучшению понимания фундаментальных аспектов сетевого кодирования (Network Coding или NC) для улучшения показателей пропускной способности, отказоустойчивости и снижения числа передач через соединённые сети и избыточных передач через широкополосные или многоточечные (point-to-multipoint) соединения. NC - это метод, обычно применяемый для кодирования пакетов с тем, чтобы восстановить потерянные исходные пакеты на стороне приёмника для быстрого получения желаемой информации полностью распределённым способом. Кроме того, с точки зрения безопасности, кодированием NC можно управлять с использованием практической схемы защиты, которая справляется с атаками загрязнения (pollution) [3] и может применяться для предотвращения получения злоумышленниками значимой информации [4] или защиты беспроводных видеопотоков при сохранении преимуществ NC [5] [6].

С точки зрения транспортного механизма NC можно разделить на две категории - когерентное NC и некогерентное NC [7] [8]. В когерентном NC исходный и целевой узел знают точную топологию сети и операции кодирования доступны на промежуточных узлах. Когда несколько потребителей пытаются получить одно и то же содержимое, например, видеотрансляцию в реальном времени, когерентное кодирование может обеспечить оптимальную пропускную способность за счёт передачи потока через созданные оптимальные деревья групповой передачи [9]. Однако для этого нужен полностью настраиваемый конкретный механизм маршрутизации и много вычислительных ресурсов для центральной координации. В случае некогерентного NC, где часто применяется случайное линейное кодирование (Random Linear Coding или RLC), не требуется знание топологии сети и операции кодирования на промежуточных узлах [10]. Некогерентное кодирование работает независимо и децентрализовано, поэтому обеспечивает большую гибкость.

NC объединяет пакеты с частями одного содержимого и может делать это в источнике и/или других узлах сети. Закодированные в сети пакеты не связаны с конкретным сервером, поскольку они могут объединяться внутри сети. Поскольку NC фокусируется на том, какую информацию следует кодировать в сетевом пакете, а не на конкретном хосте, где она была создана, это кодирование соответствует архитектуре уровня базовой сети CCNx/NDN. NC позволяет восстановить отсутствующие пакеты за счёт кодирования информации в нескольких пакетах. ICN взаимодействует с NC, поскольку позволяет кэшировать пакеты данных в сети. В типичной сети, использующей NC, отправитель может передать достаточно пакетов для получения исходных данных. ICN предоставляет возможность извлекать закодированные в сети пакеты напрямую из кэшей в сети, делая комбинацию ICN и NC особенно эффективной. Фактически кодирование NC уже реализовано для распространения информации (содержимого) [11] [12] [13]. Montpetit и др. Впервые предложили использовать методы NC для улучшения основных показателей производительности в ICN [14]. Хотя CCNx/NDN превосходно пользуется преимуществами NC (5. Преимущества NC и CCNx/NDN), нужно рассмотреть некоторые технические вопросы объединения NC и CCNx/NDN, связанные с уникальными свойствами CCNx/NDN (6. Технические соображения).

В этом документе рассматривается применение NC к архитектуре CCNx/NDN, приводятся технические соображения и описываются возможные проблемы для случая улучшения взаимодействий на основе CCNx/NDN с применением технологии NC. Следует подчеркнуть, что представление конкретных решений (например, методов оптимизации NC) для улучшения показателей производительности CCNx/NDN за счёт применения NC выходит за рамки документа.

Документ представляет совместные результаты и согласованное мнение исследовательских групп Coding for Efficient Network Communications (NWCRC) и Information-Centric Networking (ICNRG). Документ был прочитан и рецензирован всеми активными участниками этих групп. Он не является результатом IETF или стандартом.

2. Терминология

2.1. Определения, связанные с NC

В этом параграфе приведены связанные с NC термины, определённые в RFC 8406 [15] и предложенные NWCRC.

Source Packet - исходный пакет

Исходящий от источника пакет, содержащий один или несколько исходных символов (source symbol). Исходным символом считается единица данных, происходящих от источника, которая служит входными сведениями для операций кодирования. Например, пакет протокола RTP (Real-time Transport Protocol - транспортировка в реальном масштабе времени) в целом может представлять собой исходный символ. В иных ситуациях (например, для адресации пакетов переменного размера) один пакет RTP может вносить вклад в разные исходные символы.

Repair Packet - ремонтный пакет

Пакет, содержащий хотя бы один закодированный символ (его называют также ремонтным). Закодированный (ремонтный) символ - это единица данных, являющаяся результатом кодирования исходных символов или (в случае записи) исходных и/или закодированных символов. При наличии в ремонтном пакете одного ремонтного символа символ и пакет соответствуют друг другу.

Encoding versus Recoding versus Decoding - кодирование, запись, декодирование

Кодированием называется операция, принимающая исходные символы и дающая символы кодирования (encoding symbol) (исходные или закодированные). Запись - это операция, принимающая символы кодирования и дающая на выходе символы кодирования. Декодированием называется операция, принимающая символы кодирования и дающая на выходе исходные символы.

Далее определены термины, связанные с типами кодирования.

Random Linear Coding (RLC) - случайное линейное кодирование

Конкретная форма линейного кодирования с применением набора случайных коэффициентов кодирования. Линейное кодирование выполняет линейную комбинацию набора входных (т. е. исходных и кодированных) символов с использованием данного набора коэффициентов и создаёт ремонтный символ.

Block Coding - блочное кодирование

метод кодирования, где входные потоки сначала сегментируются в последовательности блоков. Кодирование и декодирование каждого блока выполняется независимо.

Sliding Window Coding (Convolutional Coding) - кодирование со скользящим окном (свёрточное кодирование)

базовый класс методов кодирования, основанных на скользящем окне кодирования, которое представляет собой набор исходных (кодированных в случае перекодирования) символов, служащих входными данными. Этот метод является альтернативой блочному кодированию.

Fixed or Elastic Sliding Window Coding - кодирование с фиксированным или эластичным скользящим окном

Метод кодирования, создающий кодированные символы «на лету» (обычно с помощью линейного кодирования) из набора исходных символов представленные в скользящем окне кодирования в данный момент. Скользящее окно может иметь фиксированный или переменный во времени размер (эластичное окно). Например, размер окна может зависеть от подтверждений получателей для конкретного исходного символа или исходного пакета (получен, декодирован, пригоден для декодирования).

Ниже приведены термины, относящиеся к низкоуровневым аспектам кодирования.

Rank of the Linear System (Degrees of Freedom) - ранг линейной системы (число степеней свободы)

На приёмной стороне это - число линейно независимых уравнений в линейной системе. Его называют также числом степеней свободы (Degrees of Freedom). Система может быть полноранговой (full rank), когда возможно декодирование, или «частичного ранга» (partial rank), когда возможно лишь частичное декодирование.

Generation (Block) - генерация (блок)

Для блочных кодов это - набор исходных символов входных потоков, которые логически группируются в блок перед выполнением кодирования.

Generation Size (Block Size) - размер генерации (блока)

Для блочных кодов это - число исходных символов, входящих в блок. Оно эквивалентно числу исходных пакетов, если в них содержится по одному исходному символу.

Coding Coefficient - коэффициент кодирования

Для линейного кодирования это - коэффициент в некоем конечном поле. Коэффициенты могут выбираться разными способами - например, случайно, из предопределённой таблицы или с использованием предопределённого алгоритма и затравки (seed).

Coding Vector - вектор кодирования

Набор коэффициентов кодирования, служащих для генерации кодированного символа путём линейного кодирования.

Finite Field - конечное поле

Для конечных полей, применяемых в линейных кодах, желательным свойством является то, что все элементы (кроме 0) обратимы для сложения и умножения (+ и *) и ни одна операция над элементом не может приводить к переполнению или опустошению (underflow). Примерами конечных полей служат первичные (prime) поля $\{0..p^{(m-1)}\}$, где p - простое число. Наиболее часто используются поля с $p=2$, называемые двоичными полями расширения (binary extension field) $\{0..2^{(m-1)}\}$, где m может принимать значение 1, 4, или 8 из практических соображений.

2.2. Определения, связанные с CCNx/NDN

Использованная в документе терминология, связанная с CCNx/NDN, определена в RFC 8793 [16], созданном ICNRP. Она согласуется с документами [17] [18].

3. Основы CCNx/NDN

Здесь кратко описаны основные концепции CCNx/NDN. В сети CCNx/NDN имеется два типа пакетов сетевого уровня - запросы (interest) и пакеты данных (определены в параграфе 3.4 [16]). Термин «объект содержимого» (content object), обозначающий блок данных содержимого, является синонимом пакета данных [16]. Потребитель ICN (определён в параграфе 3.2 [16]) запрашивает объект содержимого путём передачи запроса (interest) и именем данных.

После получения пересылающим элементом ICN (forwarder в соответствии с определением параграфа 3.2 в [16]) запроса interest, он выполняет ряд операций поиска. Сначала проверяется наличие искомого содержимого в кэш-хранилище, называемом хранилищем содержимого (Content Store или CS, как определено в параграфе 3.3 [16]). Если данные найдены, они отправляются запросившему и транзакция считается завершённой.

Если копии не найдено в кэше CS, выполняется поиск в таблице ожидающих запросов (Pending Interest Table или PIT, как определено в параграфе 3.3 [16]) для проверки наличия ожидающих запросов того же объекта содержимого. Если запроса не найдено, создаётся запись в таблице PIT с именем и интерфейсом, откуда получен запрос. Позднее это используется для отправки объекта содержимого, поскольку пакеты interest не включают поля источника, указывающего потребителя. Если запись для этого имени имеется в PIT, она обновляется включением входного интерфейса для этого запроса, а сам запрос отбрасывается.

После поиска в PIT выполняется поиск в таблице пересылки (Forwarding Information Base или FIB, как определено в параграфе 3.3 [16]) для выбора выходного интерфейса. FIB содержит список префиксов имён и соответствующих интерфейсов пересылки для отправки запроса в направлении узла пересылки (forwarder), владеющего копией запрошенных данных.

После извлечения копии данных она передаётся потребителям с использованием следов в записях PIT. Пересылающие узлы удаляют запись о состоянии из PIT при каждом выполнении запроса и могут сохранять данные в своих CS.

Пакеты данных содержат некие сведения для проверки целостности данных и подлинности источника, в частности, сведения о соответствии данных имени [19]. Это нужно потому, что аутентификация объекта очень важна для CCNx/NDN. Однако пересылающие узлы могут пропускать этот этап для повышения скорости обработки.

Важным аспектом CCNx/NDN является отсутствие сессии между потребителем и конкретным сервером. Фактически пересылающий узел или издатель (producer, как определено в параграфе 3.2 [16]), возвращающий объект содержимого, не знает о местоположении потребителя в сети, а тот не знает о местоположении узла, предоставившего содержимое. Теоретически это позволяет запросам interest следовать в сети по разным путям и даже по разным сетям.

4. Основы NC

Тогда как в традиционной сети IP пересылающий узел просто ретранслирует полученные пакеты, NC позволяет такому узлу комбинировать некоторые полученные пакеты данных в один или несколько пакетов для отправки. В этом разделе просто описываются базовые операции NC с упором на RLC в блочном режиме, который хорошо известен как основной метод кодирования.

Для простоты рассмотрим пример сквозного кодирования, где издатель и потребитель выполняют, соответственно, кодирование и декодирование элемента содержимого. Это сквозное кодирование в NC рассматривается как частный случай. Производитель делит содержимое на несколько блоков, называемых генерациями (generation). Кодирование и декодирование выполняется независимо для каждого блока (генерации). Предположим, что каждый блок содержит K исходных пакетов источника с одинаковым размером. Если размеры пакетов различаются, они дополняются нулями до большего. Для создания одного ремонтного пакета в некоем блоке издатель линейно комбинирует K исходных пакетов источника, выполняя сложение и умножение с использованием вектора кодирования, содержащего K коэффициентов, которые случайно выбираются из некоего конечного поля. Издатель может отвечать на запросы interest, отправляя исходные и ремонтные пакеты в поток содержимого (это называется систематическим кодированием), где ремонтные пакеты обычно применяются для восстановления потерянных исходных пакетов.

Ремонтные пакеты также можно применять для кодирования. Если пересылающие узлы знают каждый вектор кодирования и идентификатор генерации (далее generation ID) принятых ремонтных пакетов, они могут выполнить операцию кодирования (перекодирование), которая является наиболее отличительной чертой NC по сравнению с другими методами кодирования.

На стороне потребителя выполняется декодирование путём решения системы линейных уравнений, представленной векторами кодирования полученных исходных и ремонтных (возможно, лишь ремонтных) пакетов в рамках некоей генерации (блока). Для получения всех исходных пакетов потребителю нужно не меньше K линейно независимых пакетов. Иными словами, потребитель должен получить хотя бы K линейно независимых пакетов данных (их называют инновационными). Поскольку приём линейно зависимого пакета данных бесполезен для декодирования, перекодированию следует создавать и представлять инновационные пакеты. Одно из основных преимуществ RLC состоит в том, что даже при конечном поле малого размера (например, $q=2^8$), вероятность создания линейно зависимых пакетов пренебрежимо мала [9].

5. Преимущества NC и CCNx/NDN

Сочетание NC и CCNx/NDN может способствовать эффективному крупномасштабному распространению содержимого. По отдельности эти подходы обеспечивают похожие преимущества, такие как повышение производительности и отказоустойчивости. Различие состоит в том, что на транспортном уровне рассматривает поток содержимого как алгебраические данные, которые нужно комбинировать [7], а CCNx/NDN фокусируется на самом содержимом. Поскольку эти подходы дополняют друг друга, их сочетание было бы выгодно и естественно.

Коммуникации на основе имён в CCNx/NDN позволяют потребителям получать искомое содержимое без необходимости создавать и поддерживать сквозные каналы взаимодействия между узлами. Это облегчает использование сетевого кэширования, поиска и извлечения по нескольким путям, а также поддержку мобильности потребителей без необходимости обновлять сведения о местоположении и идентификаторе информации при переходах [1]. Кроме того, коммуникации на основе имён по своей сути поддерживают групповое взаимодействие, поскольку идентичные запросы interest объединяются пересылающими узлами.

NC может позволять транспортной системе CCNx/NDN эффективно распространять и кэшировать данные, связанные с извлечением по нескольким путям [14]. Использование извлечения по нескольким путям и кэширования в сети с помощью NC способствует не только более частому попаданию в кэш, но и повышению уровня анонимности каждого потребителя (данного потребителя может обслуживать набор возможных маршрутизаторов) [20]. Расширение осложняет злоумышленникам получение сведений о потребляемом другими содержимом, способствуя конфиденциальности кэшей. Были представлены другие варианты использования NC в CCNx/NDN, такие как распространение содержимого с кэшированием в сети [21] [22] [23], бесшовная мобильность потребителей [24] [25], малые задержки и потери при доставке видео-поток [26]. С учётом этого следует рассмотреть интеграцию NC в CCNx/NDN.

6. Технические соображения

В этом разделе приведены соображения по использованию CCNx/NDN с NC с точки зрения сетевой архитектуры и протоколов. Документ фокусируется на NC с блочным кодированием.

6.1. Именованное содержимое

Именованные объекты содержимого столь же важны в CCNx/NDN, как именованные хосты в современной сети Internet [19]. В этом параграфе представлены две возможные схемы именования.

6.1.1. Уникальные имена для пакетов NC

Каждый исходный и ремонтный пакет (далее, пакет NC) может иметь уникальное имя, как и каждый исходный объект содержимого в CCNx/NDN, а для работы PIT и CS обычно нужны уникальные имена, идентифицирующие пакеты NC. В качестве метода именования пакетов NC с учётом особенностей блочного кодирования можно использовать вектор кодирования и идентификатор генерации как часть имени объекта содержимого. Как и в [21], для идентификатора генерации g -id, размера генерации (блока) 4 и вектора кодирования (1,0,0,0) имя может иметь вид /CCNx.com/video-A/g-id/1000. Можно также применять некоторые другие идентификаторы, связанные со схемой кодирования. Например, может применяться идентификатор кодирования enc-id, задающий схему, как в /CCNx.com/video-A/enc-id/g-id/1000,

определённом в FEC Framework (FECFRAME) [27]. Эта схема именования проста и может поддерживать доставку пакетов NC с такими же операциями в PIT/CS как для объектов содержимого.

При использовании схемы именования содержимого, подобной описанной здесь, запрос interest для пакета NC может иметь полное имя, включающее идентификатор генерации и вектор кодирования (/CCNx.com/video-A/g-id/1000) или только префикс имени, включающий лишь generation ID (/CCNx.com/video-A/g-id). В первом случае пересылающими просто выполняется точное сопоставление с PIT (как в CCNx/NDN). Потребитель способен указать и получить инновационный пакет, требуемый для успешного декодирования. Это может перенести генерацию вектора кодирования от пересылающего узла к потребителю.

Во втором случае требуется частичное сопоставление имени у пересылающего. Поскольку interest лишь с префиксом имени соответствует любому пакету NC с таким же префиксом, потребитель может сразу же получить пакет NC от соседнего CS (из сетевого кэша), не зная заранее векторов кодирования в кэшированных пакетах NC. Когда пакеты NC при передаче изменяются путём перекодирования в сети, выполняемого пересылающими, потребитель также может получить изменённые пакеты NC. Однако в отличие от первого случая у него может не быть достаточной свободы (см. параграф 6.2.3. Работа пересылающего). Для решения этой проблемы может потребоваться новый тип TLV в сообщении interest для указания дополнительных сведений о кодировании, чтобы ограничить число получаемых пакетов NC. Например, это можно сделать путём задания векторов кодирования инновационных пакетов для потребителя (их называют матрицами декодирования) как в [14]. Это расширение может повлечь существенный рост размера пакетов interest, поэтому может оказаться полезным использование для векторов кодирования механизмов сжатия [28] [29]. Без таких сведений о кодировании, представляемых в interest, пересылающему придётся поддерживать записи о запросах interest, которые были выполнены ранее (см. 6.2.3. Работа пересылающего).

6.1.2. Неуникальные имена для пакетов NC

Пакет NC может иметь имя, указывающее, что это пакет NC, и переносить сведения о кодировании в поле метаданных внутри содержимого (т. е. имя включает тип данных, исходный или ремонтный пакет). Это было бы невыгодно для приложений, которым не нужно понимать содержимое пакета (payload). Из-за того, что несколько пакетов NC могут иметь одно имя, для потребителя требуется механизм получения инновационных пакетов. Как указано в параграфе 6.3. Кэширование в сети, требуется также механизм для поддержки множества инновационных пакетов в CS. Кроме того, при шифровании содержимого возникают дополнительные расчётные издержки.

6.2. Транспорт

Основанная на вытягивании (pull) функция запросов-откликов CCNx/NDN является там фундаментальным транспортным подходом - для одного запроса interest извлекается не более одного пакета данных. Это значит, что пересылающий или издатель не могут внедрить незапрошенные пакеты данных по своей инициативе. Считается важным соблюдение этого правила, поскольку нарушение 1) приведёт к атакам с отказом в обслуживании (DoS), 2) делает непригодными имеющиеся подходы к управлению перегрузкой на основе правила и 3) снизит эффективность имеющихся подходов к мобильности потребителей. Таким образом, для применения NC в CCNx/NDN. Следует рассмотреть описанную ниже базовую операцию. Тем не менее, при нарушении правила указанные проблемы безопасности должны быть решены.

6.2.1. Область действия NC

Не решён вопрос относительно возможности пересылающего выполнить перекодирование в сети для полученных транзитных пакетов данных или операции NC можно выполнять только для данных, которые соответствуют interest. В последнем случае кодирование или перекодирование выполняется для генерации пакета на любом пересылающем узле, способном ответить на interest. Это возможно, когда каждый пакет NC имеет уникальное имя, а interest имеет полное имя. С другой стороны, если interest имеет частичное имя без сведений о векторе кодирования или несколько пакетов NC имеют одно имя, может возникнуть первый случай и перекодирование может происходить в любом месте сети, где возможно изменить полученный пакет NC и переслать его. Поскольку CCNx/NDN включает механизмы обеспечения целостности данных в процессе передачи, кодирование в сети будет вызывать сложности, связанные с обеспечением работы механизмов защиты целостности. Точно так же может быть полезно сетевое кэширование пакетов NC на пересылающих узлах, однако им потребуется тот или иной механизм для проверки пригодности пакетов NC (см. 9. Вопросы безопасности).

6.2.2. Работа потребителя

Чтобы воспользоваться преимуществами NC (возможно, связанными с кэшированием в сети), потребитель должен отправлять запросы interest, которые направляют пересылающего (или издателя) отвечать инновационными пакетами, когда они доступны. В случае, когда каждый пакет NC может иметь уникальное имя (как описано в параграфе 6.1. Именованное содержимое), отправка interest с указанием уникального имени в g-id и вектора кодирования для пакета NC даёт потребителю возможность получить инновационный пакет, если такой доступен у некоторых пересылающих.

Чтобы задать точное имя пакета NC для извлечения, потребитель должен знать действительную схему именования. С практической точки зрения желательно, чтобы приложение потребителя автоматически создавало верные компоненты имени независимо от каких-либо спецификаций приложения. С этой целью приложение-потребитель может получить и указать (сослаться) манифест [17], указывающий объекты содержимого, включая пакеты NC, или использовать какой-либо спецификатор схемы кодирования как часть имени при создании компонентов имени в interest для запроса инновационных пакетов.

И наоборот, потребитель без возможности декодирования (например, конкретный датчик) может пожелать получать лишь исходные пакеты. Как указано в параграфе 6.1. Именованное содержимое, поскольку пакет NC может иметь имя, явно отличающееся от исходных пакетов, возможна передача interest для извлечения лишь исходных пакетов.

6.2.3. Работа пересылающего

Если пересылающий постоянно отвечает на входящие interest неинновационными пакетами, потребители не могут декодировать и получить исходные пакеты. Эта проблема возникает, когда 1) входящие interest для пакетов NC не задают некоторые параметры кодирования, такие как векторы кодирования для применения, и 2) у пересылающего нет

достаточного числа линейно независимых пакетов NC (возможно в CS) для использования при перекодировании. В этом случае пересылающий должен определить, может ли он генерировать инновационные пакеты для пересылки на интерфейсы, с которых поступил запрос interest. Подход к решению этой задачи состоит в том, что пересылающий узел ведёт учёт interest для конкретных имён, generation ID и входных интерфейсов, чтобы записать число уже предоставленных степеней свободы [21]. Поскольку такая схема требует поддержки состояний (возможно, и таймеров) на узлах пересылки, должны учитываться вопросы расширяемости и практичности. Кроме того, потребуются некоторые транспортные механизмы для обнаружения и восстановления потерь в сети [25][26] на узлах пересылки, а также управляемые потребителем механизмы для быстрого восстановления потерь и достижения преимуществ NC. Если пересылающий не может вернуть соответствующий инновационный пакет из локального хранилища содержимого или создать «на лету» перекодированный пакет, являющийся инновационным, важно, чтобы пересылающий не просто возвращал неинновационный пакет, а выполнял поиск пересылки в своей базе FIB и пересылал запрос interest в направлении издателя или восходящего пересылающего узла, способного обеспечить инновационный пакет. В этом контексте для эффективного и быстрого поиска инновационного пакета следует рассматривать подходящую настройку FIB и эффективную стратегию пересылки interest.

В другом возможном случае, когда запросы interest принимаются лишь для исходных пакетов, пересылающий узел может попытаться декодировать и получить все исходные пакеты и сохранить их (при доступности полной ёмкости кэша), позволяя тем самым ускоренные отклики на последующие interest. Поскольку кодирование или перекодирование связано с вычислительными издержками, пересылающий узел должен определить, как отвечать на полученные запросы interest в соответствии с вариантом применения (например, в чувствительном или устойчивом к задержкам приложении) и своими условиями, такими как доступность пространства в кэше и вычислительные возможности.

6.2.4. Работа издателя

Перед выполнением NC для конкретного содержимого в CCNx/NDN издатель должен обеспечить (отвечает за) разделение всего содержимого на более мелкие объекты для предотвращения фрагментации пакетов, которая может вызывать избыточную обработку и снижать пропускную способность. Размер содержимого должен соответствовать допустимому размеру пакетов, чтобы избежать фрагментирования в сети CCNx/NDN. Издатель выполняет операции кодирования для набора небольших объектов содержимого и именования для пакетов NC.

Если производитель берет на себя роль ведущего при определении применяемых векторов кодирования при генерации пакетов NC, имеется 3 общих стратегии именования и создания пакетов NC, указанные ниже.

1. Потребители сами понимают детали соглашений об именовании для пакетов NC и, таким образом, могут передавать соответствующие запросы interest в направлении издателя для получения пакетов NC, для которых параметры кодирования уже заданы издателем.
2. Издатель определяет векторы кодирования и генерирует пакеты NC после получения запросов interest, задающих пакеты, которые потребитель хочет получить.
3. Схема именования для задания векторов кодирования и соответствующих пакетов NC явно представлена через манифест (например, FLIC [30]), который может быть получен потребителем и использован для выбора среди доступных векторов кодирования и соответствующих пакетов, чтобы передать соответствующие interest.

В первом случае, хотя потребители не могут гибко задавать вектор кодирования для генерации пакетов NC, которые хочется получить, задержка получения пакета NC будет меньше, чем в двух других случаях. Во втором случае возникает расплата в виде задержки на дополнительные операции NC, выполняемые после получения interest. В третьем случае пакеты NC для включения в манифест, должны заранее рассчитываться издателем (поскольку манифест указывает пакеты NC хэш-значениями, а не именами), но издатель может выбрать, что включать в манифест, и создать несколько манифестов с разными вариантами кодирования заранее или по запросу, если нужно.

Общим преимуществом двух первых подходов к сквозному кодированию является то, что при добавлении издателем подписей к пакетам NC становится возможной проверка пригодности данных на всем протяжении (как в случае работы CCNx/NDN без NC). Третий подход с использованием манифеста компенсирует дополнительную задержку, вносимую необходимостью извлечения манифеста тем, что подписывается лишь манифест, а не отдельные пакеты NC.

6.2.5. Совместимость с прежними версиями

Операции NC следует применять в дополнение к обычному поведению ICN, наряду с ним. Поэтому не поддерживающим NC узлам следует обеспечивать корректную обработку не только при пересылке пакетов NC, но и при их кодировании. Модели NC следует быть совместимой с обычной схемой для обеспечения совместимости с прежними версиями и плавного перехода от одной схемы к другой.

6.3. Кэширование в сети

Кэширование является полезным методом повышения пропускной способности и снижения задержки в разных приложениях. Сетевое кэширование в CCNx/NDN, по сути, обеспечивает поддержку на сетевом уровне и является выгодным за счёт вовлечения NC для обеспечения эффективной групповой передачи [31], извлечения данных по нескольким путям [21] [24] и быстрого восстановления потерь [26]. Однако требуется рассмотреть несколько вопросов.

Как правило ёмкость CS ограничена и политика кэширования влияет на производительность потребителя [32] [33] [34]. Поэтому для пересылающих очень важно определить, какие объекты содержимого следует кэшировать, а какие - отбрасывать. Поскольку чувствительным к задержкам приложениям часто не требуется долговременное кэширование в сети из-за ограничений, связанных с реальным масштабом времени, пересылающие должны знать о необходимости кэширования полученных объектов содержимого для сохранения объёма кэша. В CCNx это можно сделать установкой рекомендуемого времени кэширования (Recommended Cache Time или RCT) в необязательном заголовке пакета данных на стороне издателя. RCT служит рекомендацией для CS-кэша при определении срока хранения кэшированного объекта. При RCT = 0 пересылающий считает кэширование бесполезным, а при больших значениях RCT кэшировать объекты на длительное время.

Одним из важных аспектов кэширования в сети является возможность пересылающих кэшировать пакеты NC в своих CS. Они могут кэшировать пакеты NC, не имея возможности проверить пригодность объектов содержимого, поэтому

для кэширования пакетов NC нужен механизм проверки пакетов NC (см. 9. Вопросы безопасности). Когда пакеты NC имеют одно имя, нужен также какой-либо механизм их идентификации.

6.4. Беспрепятственная мобильность потребителей

Важным свойством CCNx/NDN является отсутствие сессий, позволяющее потребителю и пересылающим передавать множество запросов interest параллельно в направлении разных копий содержимого с одновременным асинхронным использованием нескольких интерфейсов. За счёт извлечения данных по нескольким путям потребитель может получить содержимое из нескольких копий, распространяемое с использованием суммарной производительности нескольких интерфейсов. Для связывания нескольких копий потребитель может воспользоваться тем или иным механизмом адаптации потокового видео [24] или контроль перегрузок для получения содержимого [35].

NC добавляет в CCNx уровень надёжности распределенным и асинхронным способом, поскольку в NC обеспечивается механизм, гарантирующий для нескольких запросов interest, переданных параллельно разным копиям содержимого, получение инновационных пакетов даже в случаях потери пакетов на некоторых путях к этим копиям. Это относится к перемещению (мобильности) потребителя [24], когда тот может получить дополнительные степени свободы с любым инновационным пакетом, если в процессе перемещения имеется хотя бы один доступный интерфейс. Стратегия пересылки interest у потребителя (возможно, и у пересылающего) будет нужна потребителю для получения инновационных пакетов с сохранением беспрепятственной мобильности.

7. Проблемы

В этом разделе представлены некоторые проблемы и направления исследования, связанные с применением NC в CCNx/NDN.

7.1. Адаптация свёрточного кодирования

К настоящему времени предложено несколько подходов к блочному кодированию, тем не менее, в CCNx/NDN до сих пор нет достаточного рассмотрения и применения свёрточного кодирования (например, со скользящим или эластичным окном). Свёрточное кодирование часто подходит для случаев, когда нужна частично или полностью гарантированная доставка непрерывного потока данных, особенно в реальном масштабе времени. Как и в [36], на основе сквозного кодирования для непрерывного потока содержимого было бы выгодно применять кодирование со скользящим окном в CCNx/NDN. В этом случае издатель должен соответствующим образом установить параметры кодирования и сообщить сведения потребителям, а те должны передавать запросы interest, дополнительные информацией о состоянии приёма и декодирования данных. Поскольку CCNx/NDN применяет поэтапное (hop-by-hop) состояние пересылки, было бы целесообразно обсудить и исследовать применение свёрточного кодирования в стиле hop-by-hop и возможные преимущества. В частности, для случаев, когда сетевое перекодирование может происходить в узлах пересылки, потребуется управление окном кодирования и CS, для чего следует соответствующие возможности и практичность.

7.2. Контроль скорости и перегрузки

Добавление избыточности за счёт ремонтных пакетов может привести к дополнительному насыщению сети и отрицательно повлиять на общую пропускную способность. В частности, для случаев, когда более желательно беспристрастное распределение общей полосы, каждый поток содержимого должен адаптироваться к условиям в сети для справедливого распределения доступной полосы канала. Таким образом, все потоки содержимого должны совместно реализовать контроль перегрузок для регулирования потребляемой полосы [37]. Это требует эффективного подхода к развёртыванию (например, поддерживаемый пересылающими контроль перегрузки может обеспечить преимущества при частичном развёртывании).

Как описано в параграфе 6.4. Беспрепятственная мобильность потребителей, NC может способствовать беспрепятственной мобильности потребителей за счёт получения инновационных пакетов путём извлечения данных по нескольким путям без дубликатов, а предотвращение дубликатов способствует контролю перегрузок. Разработка эффективного механизма управления скоростью и контроля перегрузок для обеспечения мобильности потребителей при одновременном повышении общей пропускной способности и снижении задержек за счёт полного использования операций NC может оказаться сложной задачей.

7.3. Безопасность

Хотя CCNx/NDN создаёт новые проблемы безопасности на сетевом уровне, отличающемся от традиционных сетей IP, такие как отравление кэша, атаки с загрязнением и DoS-атаки с использованием пакетов interest, некоторые подходы в обеспечению безопасности уже представлены [19] [38]. Применение NC в CCNx/NDN связано с двумя аспектами безопасности, которые требуется учитывать.

Первый вопрос связан с записью на пересылающих узлах. Требуется тот или иной механизм защиты целостности пакетов NC, недавно созданных перекодированием в сети, чтобы потребители и другие пересылающие получали действительные пакеты NC. Для этого имеется несколько подходов, рассмотренных в разделе 9. Вопросы безопасности, но возможны и более эффективные методы с меньшей сложностью и вычислительными затратами.

Другой вопрос связан с вредоносными запросами атакующих и внедрением пакетов NC, которые могут усиливать некоторые атаки. Поскольку пакеты NC мало применяются в общем случае, они могут быть нацелены на загрязнение кэша менее частыми запросами, которое будет препятствовать кэшированию на основе популярности запросов. С такими атаками необходимо бороться для поддержки эффективности кэширования в сети. За счёт внедрения недействительных пакетов NC с целью заполнения CS на узлах пересылки влияние атак с отравлением кэша будет зависеть от степени контроля целостности пакетов NC. В предположении, что каждый пакет NC имеет действительную подпись, прямой подход будет заключаться в проверке подписи пересылающими узлами «на лету» с сохранением и пересылкой только проверенных пакетов NC. Однако проверка подписи пересылающими узлами со скоростью линии может оказаться невозможной, поэтому следует рассмотреть механизмы распределения и снижения такой нагрузки для сохранения преимуществ кэширования в сети, таких как сокращение задержек и нагрузки на сеть.

7.4. Расширяемость маршрутизации

В CCNx/NDN протокол маршрутизации по именам без процесса распознавания (resolution) упрощает процесс маршрутизации и сокращает общую задержку. В маршрутизации IP рост размера таблиц маршрутизации стал проблемой. Таким образом, необходимо применять схему иерархического именования, чтобы улучшить расширяемость маршрутизации за счёт агрегирования маршрутных данных.

Для реализации преимуществ NC потребителям нужно эффективно получать инновационные пакеты с применением механизмов CCNx/NDN для извлечения по нескольким путям. Это требует наличия эффективного механизма маршрутизации для подбирающего заполнения таблиц FIB, а также эффективной пересылки запросов interest. Такая координация может препятствовать расширяемости маршрутизации. Будет сложно обеспечить эффективную и расширяемую маршрутизацию запросов interest для пакетов NC с одновременным упрощением процесса маршрутизации.

8. Взаимодействие с IANA

Этот документ не требует действий IANA.

9. Вопросы безопасности

Перекодирование в сети является отличительной чертой NC. Запрашиваться и применяться (возможно, сохраняться) должны лишь действительные пакеты NC, созданные перекодированием в сети. Для решения этой задачи имеется несколько подходов. Во-первых, это проверка подписей с возможностью использования нескольких подписей. Это позволяет иметь свой уникальный ключ подписи не только исходному издателю содержимого, но и некоторым пересылающим, ответственным за перекодирование в сети. Каждый пересылающий узел группы подписывает созданный недавно пакет NC, чтобы другие узлы могли проверить данные по этой подписи. CS может проверять подписи пакетов NC перед их сохранением, чтобы не кэшировать недействительные пакеты. Во-вторых, потребители могут вносить ограничения для правил сопоставления, используя только имя запрошенных данных. Неоднозначность запросов interest можно устранить, задавая имя и идентификатор ключа (дайджест открытого ключа издателя) для сопоставления с запрошенными данными. Такое ограничение KeyId встроено в CCNx [17]. Пересылаются и хранятся в CS лишь пакеты, соответствующие interest и KeyId, что снижает возможности отравления кэша. Кроме того, в CCNx имеется правило, которому подчиняются CS, для предотвращения усиления непригодных данных. Если для interest задано ограничение KeyId, CS недопустимо отвечать на запрос, пока нет уверенности в корректности подписи соответствующего объекта содержимого. Если CS не может проверить подпись, запрос может считаться отсутствующим в кэше с пересылкой его восходящим (вышестоящим) узлам пересылки. Третьим подходом является поддержка цепочек сертификатов (возможно, без удостоверяющего центра) с использованием того или иного механизма (например, [39]) организации доверенного пути доставки. Этот подход использует механизм поэтапной аутентификации, в котором выполняется сбор сертификатов, объединённый с пересылкой, для создания цепочки сертификатов, обеспечивающей доверие к пути доставки.

В зависимости от принятой стратегии кэширования, такой как правила замены кэша, пересылающим следует соблюдать осторожность при записи и сохранении пакетов NC в CS, поскольку они могут быть частью атаки с загрязнением кэша. Для смягчения таких атак пересылающим узлам следует проверять частоту запросов содержимого, чтобы обнаруживать атаки и иметь возможность ограничивать число запросов, игнорируя некоторые повторяющиеся запросы. Затем они могут принять решение о применении или изменении механизма замены кэша.

Издатели и пересылающие должны уделять пристальное внимание DoS-атакам, нацеленным на создание высокой нагрузки операций NC с использованием запросов interest для пакетов NC. Для смягчения таких атак пересылающие узлы могут ограничивать скорость. Например, они могут отслеживать рост PIT для пакетов NC по содержимому с целью обнаружения атак и ограничения при необходимости частоты прибытия запросов interest. Если приложение NC хочет защитить запрос interest, считающийся активатором NC, для предотвращения таких атак, ему следует рассмотреть применение шифрования и явного протокола.

10. Литература

- [1] Jacobson, V., Smetters, D., Thornton, J., Plass, M., Briggs, N., and R. Braynard, "Networking Named Content", Proc. CoNEXT, ACM, DOI 10.1145/1658939.1658941, December 2009, <<https://doi.org/10.1145/1658939.1658941>>.
- [2] Zhang, L., Afanasyev, A., Burke, J., Jacobson, V., Claffy, K., Crowley, P., Papadopoulos, C., Wang, L., and B. Zhang, "Named data networking", ACM SIGCOMM Computer Communication Review, vol. 44, no. 3, DOI 10.1145/2656877.2656887, July 2014, <<https://doi.org/10.1145/2656877.2656887>>.
- [3] Gkantsidis, C. and P. Rodriguez, "Cooperative Security for Network Coding File Distribution", Proc. Infocom, IEEE, DOI 10.1109/INFCOM.2006.233, April 2006, <<https://doi.org/10.1109/INFCOM.2006.233>>.
- [4] Cai, N. and R. Yeung, "Secure network coding", Proc. International Symposium on Information Theory (ISIT), IEEE, DOI 10.1109/ISIT.2002.1023595, June 2002, <<https://doi.org/10.1109/ISIT.2002.1023595>>.
- [5] Lima, L., Gheorghiu, S., Barros, J., Medard, M., and A. Toledo, "Secure Network Coding for Multi-Resolution Wireless Video Streaming", IEEE Journal of Selected Area (JSAC), vol. 28, no. 3, DOI 10.1109/JSAC.2010.100409, April 2010, <<https://doi.org/10.1109/JSAC.2010.100409>>.
- [6] Vilea, J., Lima, L., and J. Barros, "Lightweight security for network coding", Proc. ICC, IEEE, DOI 10.48550/arXiv.0807.0610, May 2008, <<https://doi.org/10.48550/arXiv.0807.0610>>.
- [7] Koetter, R. and M. Medard, "An Algebraic Approach to Network Coding", IEEE/ACM Transactions on Networking, vol. 11, no. 5, DOI 10.1109/TNET.2003.818197, October 2003, <<https://doi.org/10.1109/TNET.2003.818197>>.
- [8] Vyetenko, S., Ho, T., Effros, M., Klierer, J., and E. Erez, "Rate regions for coherent and noncoherent multisource network error correction", Proc. International Symposium on Information Theory (ISIT), IEEE, DOI 10.1109/ISIT.2009.5206077, June 2009, <<https://doi.org/10.1109/ISIT.2009.5206077>>.

- [9] Wu, Y., Chou, P., and K. Jain, "A comparison of network coding and tree packing", Proc. International Symposium on Information Theory (ISIT), IEEE, DOI 10.1109/ISIT.2004.1365182, June 2004, <<https://doi.org/10.1109/ISIT.2004.1365182>>.
- [10] Ho, T., Medard, M., Koetter, R., Karger, D., Effros, M., Shi, J., and B. Leong, "A Random Linear Network Coding Approach to Multicast", IEEE Trans. on Information Theory, vol. 52, no. 10, DOI 10.1109/TIT.2006.881746, October 2006, <<https://doi.org/10.1109/TIT.2006.881746>>.
- [11] Dimarkis, A., Godfrey, P., Wu, Y., Wainwright, M., and K. Ramchandran, "Network Coding for Distributed Storage Systems", IEEE Trans. Information Theory, vol. 56, no.9, DOI 10.1109/TIT.2010.2054295, September 2010, <<https://doi.org/10.1109/TIT.2010.2054295>>.
- [12] Gkantsidis, C. and P. Rodriguez, "Network coding for large scale content distribution", Proc. Infocom, IEEE, DOI 10.1109/INFCOM.2005.1498511, March 2005, <<https://doi.org/10.1109/INFCOM.2005.1498511>>.
- [13] Seferoglu, H. and A. Markopoulou, "Opportunistic Network Coding for Video Streaming over Wireless", Proc. Packet Video Workshop (PV), IEEE, DOI 10.1109/PACKET.2007.4397041, November 2007, <<https://doi.org/10.1109/PACKET.2007.4397041>>.
- [14] Montpetit, M., Westphal, C., and D. Trossen, "Network Coding Meets Information-Centric Networking: An Architectural Case for Information Dispersion Through Native Network Coding", Proc. Workshop on Emerging Name-Oriented Mobile Networking Design (NoM), ACM, DOI 10.1145/2248361.2248370, June 2012, <<https://doi.org/10.1145/2248361.2248370>>.
- [15] Adamson, B., Adjih, C., Bilbao, J., Firoiu, V., Fitzek, F., Ghanem, S., Lochin, E., Masucci, A., Montpetit, M.-J., Pedersen, M., Peralta, G., Roca, V., Ed., Saxena, P., and S. Sivakumar, "Taxonomy of Coding Techniques for Efficient Network Communications", RFC 8406, DOI 10.17487/RFC8406, June 2018, <<https://www.rfc-editor.org/info/rfc8406>>.
- [16] Wissingh, B., Wood, C., Afanasyev, A., Zhang, L., Oran, D., and C. Tschudin, "Information-Centric Networking (ICN): Content-Centric Networking (CCNx) and Named Data Networking (NDN) Terminology", RFC 8793, DOI 10.17487/RFC8793, June 2020, <<https://www.rfc-editor.org/info/rfc8793>>.
- [17] Mosko, M., Solis, I., and C. Wood, "Content-Centric Networking (CCNx) Semantics", RFC 8569, DOI 10.17487/RFC8569, July 2019, <<https://www.rfc-editor.org/info/rfc8569>>.
- [18] Mosko, M., Solis, I., and C. Wood, "Content-Centric Networking (CCNx) Messages in TLV Format", RFC 8609, DOI 10.17487/RFC8609, July 2019, <<https://www.rfc-editor.org/info/rfc8609>>.
- [19] Kutscher, D., Ed., Eum, S., Pentikousis, K., Psaras, I., Corujo, D., Saucez, D., Schmidt, T., and M. Waehlich, "Information-Centric Networking (ICN) Research Challenges", RFC 7927, DOI 10.17487/RFC7927, July 2016, <<https://www.rfc-editor.org/info/rfc7927>>.
- [20] Wu, Q., Li, Z., Tyson, G., Uhlig, S., Kaafar, M., and G. Xie, "Privacy-Aware Multipath Video Caching for Content-Centric Networks", IEEE Journal on Selected Areas in Communications (JSAC), vol. 38, no. 8, DOI 10.1109/JSAC.2016.2577321, June 2016, <<https://doi.org/10.1109/JSAC.2016.2577321>>.
- [21] Saltarin, J., Boursoulatze, E., Thomos, N., and T. Braun, "NetCodCCN: a network coding approach for content-centric networks", Proc. Infocom, IEEE, DOI 10.1109/INFCOM.2016.7524382, April 2016, <<https://doi.org/10.1109/INFCOM.2016.7524382>>.
- [22] Wang, J., Ren, J., Lu, K., Wang, J., Liu, S., and C. Westphal, "An Optimal Cache Management Framework for Information-Centric Networks with Network Coding", Proc. Networking Conference, IFIP/IEEE, DOI 10.1109/IFIPNetworking.2014.6857127, June 2014, <<https://doi.org/10.1109/IFIPNetworking.2014.6857127>>.
- [23] Wang, J., Ren, J., Lu, K., Wang, J., Liu, S., and C. Westphal, "A Minimum Cost Cache Management Framework for Information-Centric Networks with Network Coding", Computer Networks, Elsevier, DOI 10.1016/j.comnet.2016.08.004, August 2016, <<https://doi.org/10.1016/j.comnet.2016.08.004>>.
- [24] Ramakrishnan, A., Westphal, C., and J. Saltarin, "Adaptive Video Streaming over CCN with Network Coding for Seamless Mobility", Proc. International Symposium on Multimedia (ISM), IEEE, DOI 10.1109/ISM.2016.0054, December 2016, <<https://doi.org/10.1109/ISM.2016.0054>>.
- [25] Carofiglio, G., Muscariello, L., Papalini, M., Rozhnova, N., and X. Zeng, "Leveraging ICN In-network Control for Loss Detection and Recovery in Wireless Mobile networks", Proc. of the 3rd ACM Conference on Information-Centric Networking, DOI 10.1145/2984356.2984361, September 2016, <<https://doi.org/10.1145/2984356.2984361>>.
- [26] Matsuzono, K., Asaeda, H., and T. Turletti, "Low Latency Low Loss Streaming using In-Network Coding and Caching", Proc. Infocom, IEEE, DOI 10.1109/INFCOM.2017.8057026, May 2017, <<https://doi.org/10.1109/INFCOM.2017.8057026>>.
- [27] Watson, M., Begen, A., and V. Roca, "Forward Error Correction (FEC) Framework", RFC 6363, DOI 10.17487/RFC6363, October 2011, <<https://www.rfc-editor.org/info/rfc6363>>.
- [28] Thomos, N. and P. Frossard, "Toward One Symbol Network Coding Vectors", IEEE Communications Letters, vol. 16, no. 11, DOI 10.1109/LCOMM.2012.092812.121661, November 2012, <<https://doi.org/10.1109/LCOMM.2012.092812.121661>>.
- [29] Lucani, D., Pedersen, M., Ruano, D., Sørensen, C., Heide, J., Fitzek, F., and O. Geil, "Fulcrum Network Codes: A Code for Fluid Allocation of Complexity", DOI 10.48550/arXiv.1404.6620, April 2014, <<https://doi.org/10.48550/arXiv.1404.6620>>.
- [30] Tschudin, C., Wood, C. A., Mosko, M., and D. Oran, "File-Like ICN Collections (FLIC)", Work in Progress, Internet-Draft, draft-irtf-icnrg-flic-03, 7 November 2021, <<https://datatracker.ietf.org/doc/html/draft-irtf-icnrg-flic-03>>.

- [31] Maddah-Ali, M. and U. Niesen, "Coding for Caching: Fundamental Limits and Practical Challenges", IEEE Communications Magazine, vol. 54, no. 8, DOI 10.1109/MCOM.2016.7537173, August 2016, <<https://doi.org/10.1109/MCOM.2016.7537173>>.
- [32] Perino, D. and M. Varvello, "A reality check for content centric networking", Proc. SIGCOMM Workshop on Information-centric networking (ICN '11), ACM, DOI 10.1145/2018584.2018596, August 2011, <<https://doi.org/10.1145/2018584.2018596>>.
- [33] Podlipnig, S. and L. Böszörményi, "A Survey of Web Cache Replacement Strategies", Proc. ACM Computing Surveys, vol. 35, no. 4, DOI 10.1145/954339.954341, December 2003, <<https://doi.org/10.1145/954339.954341>>.
- [34] Rossini, G. and D. Rossi, "Evaluating CCN multi-path interest forwarding strategies", Elsevier Computer Communications, vol. 36, no. 7, DOI 10.1016/j.comcom.2013.01.008, April 2013, <<https://doi.org/10.1016/j.comcom.2013.01.008>>.
- [35] Mahdian, M., Arianfar, S., Gibson, J., and D. Oran, "MIRCC: Multipath-aware ICN Rate-based Congestion Control", Proc. Conference on Information-Centric Networking (ICN), ACM, DOI 10.1145/2984356.2984365, September 2016, <<https://doi.org/10.1145/2984356.2984365>>.
- [36] Tournoux, P., Lochin, E., Lacan, J., Bouabdallah, A., and V. Roca, "On-the-Fly Erasure Coding for Real-Time Video Applications", IEEE Transactions on Multimedia, vol. 13, no. 4, DOI 10.1109/TMM.2011.2126564, August 2011, <<https://doi.org/10.1109/TMM.2011.2126564>>.
- [37] Kuhn, N., Lochin, E., Michel, F., and M. Welzl, "Forward Erasure Correction (FEC) Coding and Congestion Control in Transport", RFC 9265, DOI 10.17487/RFC9265, July 2022, <<https://www.rfc-editor.org/info/rfc9265>>.
- [38] Pentikousis, K., Ed., Ohlman, B., Davies, E., Spirou, S., and G. Boggia, "Information-Centric Networking: Evaluation and Security Considerations", RFC 7945, DOI 10.17487/RFC7945, September 2016, <<https://www.rfc-editor.org/info/rfc7945>>.
- [39] Li, R., Asaeda, H., and J. Wu, "DCAuth: Data-Centric Authentication for Secure In-Network Big-Data Retrieval", IEEE Trans. on Network Science and Engineering, vol. 7, no. 1, DOI 10.1109/TNSE.2018.2872049, September 2018, <<https://doi.org/10.1109/TNSE.2018.2872049>>.

Благодарности

Авторы признательны членам ICNRC и NWCRG, особенно Marie-Jose Montpetit, David Oran, Vincent Roca, Thierry Turletti за их значимые комментарии и предложения.

Адреса авторов

Kazuhisa Matsuzono

National Institute of Information and Communications
Technology
4-2-1 Nukui-Kitamachi, Tokyo
184-8795
Japan
Email: matsuzono@nict.go.jp

Hitoshi Asaeda

National Institute of Information and Communications
Technology

4-2-1 Nukui-Kitamachi, Tokyo
184-8795
Japan
Email: asaeda@nict.go.jp

Cedric Westphal

Huawei
2330 Central Expressway
Santa Clara, California 95050
United States of America
Email: cedric.westphal@futurewei.com

Перевод на русский язык

Николай Малых

nmalykh@protokols.ru