Internet Engineering Task Force (IETF) T. Mizrahi
Request for Comments: 7821 Marvell
Category: Experimental March 2016
ISSN: 2070-1721

UDP Checksum Complement in the Network Time Protocol (NTP)

Дополнение контрольной суммы UDP в NTP

Аннотация

Протокол сетевого времени (Network Time Protocol или NTP) позволяет синхронизировать клиентов с сервером времени с помощью протокольных сообщений с временными метками. Для повышения точности временных меток в некоторых реализациях применяются аппаратные средства создания меток времени, встраивающие точное время передачи в каждый исходящий пакет NTP. Поскольку эти пакеты доставляются по протоколу UDP, поле Checksum обновляется с учётом вставки временной метки. Этот документ предлагает использовать 2 последних октета каждого пакета как дополнение контрольной суммы (Checksum Complement), что позволяет средствам записи временных меток возможность отразить изменение контрольной суммы в этих 2 октетах, а не в поле UDP Checksum. Задаваемое этим документом поведение полностью совместимо с имеющимися реализациями NTP.

Статус документа

Документ не относится к категории Internet Standards Track и публикуется лишь для проверки, экспериментальной реализации и оценки.

Документ содержит экспериментальный протокол для сообщества Internet и является результатом работы IETF[1] и представляет согласованный взгляд сообщества IETF. Документ прошёл открытое обсуждение и был одобрен для публикации IESG[2]. Дополнительную информацию о стандартах Internet можно найти в разделе 2 в RFC 5741.

Информацию о текущем статусе документа, ошибках и способах обратной связи можно найти по ссылке http://www.rfc-editor.org/info/rfc7821.

Оглавление

1. Введение

Протокол Network Time Protocol [NTPv4] позволяет клиентам синхронизировать свои часы с сервером точного времени путём обмена пакетами NTP. Рост требований к точности синхронизации часов служит мотивом для повышения точности меток времени.

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

1.1. Промежуточные элементы

В этом документе термин «промежуточный элемент» (intermediate entity) означает элемент на пути между отправителем и получателем пакета NTP, меняющим этот пакет.

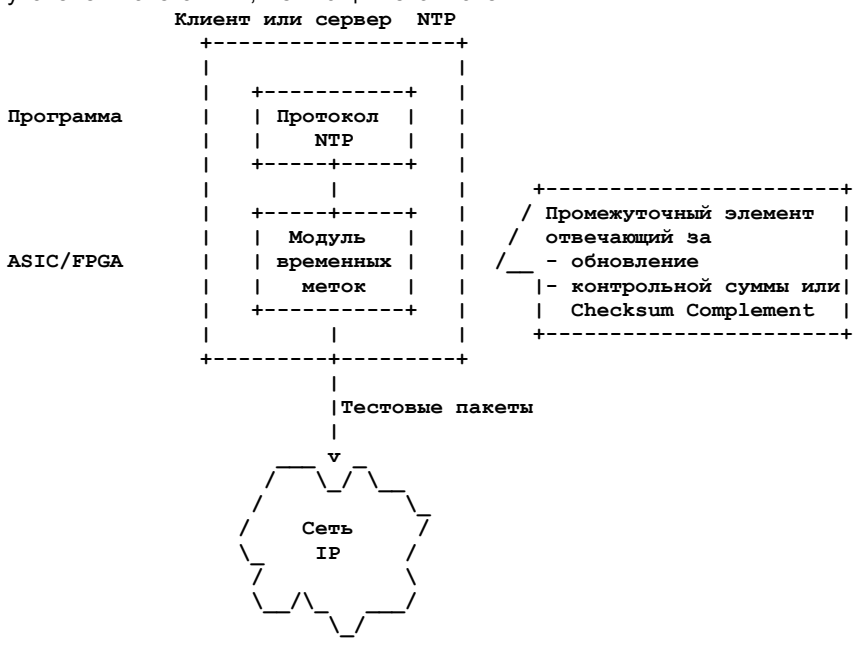

Рисунок 1. Точная установка временных меток в NTP.

Для повышения точности меток времени реализация может использовать аппаратный модуль временных меток (timestamping engine), как показано на рисунке 1. В таких случаях пакеты NTP передаются и принимаются на программном уровне, а аппаратный модуль меток меняет каждый исходящий пакет NTP, встраивая точное время передачи в поле пакета <Transmit Timestamp>.

Точность синхронизации часов через пакетную сеть сильно зависит от вариаций задержки в базовой сети и это сильно влияет на точность измерения времени. Для решения этой проблемы протокол точного времени (Precision Time Protocol или PTP) [IEEE1588] определяет «прозрачные часы» (Transparent Clock или TC) - коммутаторы и маршрутизаторы, которые повышают сквозную точность часов, обновляя поле корректировки Correction Field в пакете PTP путём добавления задержки, внесённой текущими часами TC. В NTP эквивалентных элементов в настоящее время нет, но в будущие версии NTP могут быть добавлены промежуточные узлы, меняющие пакеты NTP «на лету» с использованием поля Correction Field.

1.2. Обновление UDP Checksum

Когда данные UDP (payload) изменяются промежуточным узлом, таким как модуль меток времени, поле UDP Checksum должно соответственно обновляться. При использовании UDP по протоколу IPv4 [UDP] у промежуточного узла, не способного обновить значение UDP Checksum не остаётся вариантов кроме установки значения 0 в поле Checksum, заставляющего получателя игнорировать поле Checksum и, возможно, принимать повреждённые пакеты. UDP по протоколу IPv6, как указано в [IPv6], не разрешает значение 0 для контрольной суммы за исключением особых случаев [ZeroChecksum]. Как отмечено в [ZeroChecksum], использование нулевой контрольной суммы в общем случае не рекомендуется и его следует избегать, когда это возможно.

Поскольку промежуточный элемент меняет лишь конкретное поле пакета (Timestamp), обновление UDP Checksum можно выполнить инкрементно с использованием концепций, представленных в [Checksum].

Этот документ определяет Checksum Complement для [NTPv4]. Поле Checksum Complement является 2-октетным полем, комещаемым в конце данных UDP (payload). Это позволяет промежуточным элементам обновлять пакеты NTP и сохранять корректность UDP Checksum путём изменения 2 последних октетов в пакете вместо обновления поля UDP Checksum. Это реализуется путём добавления поля расширения NTP в конце пакета, где 2 последних октета используются как дополнение контрольной суммы (Checksum Complement).

Использование Checksum Complement позволяет в некоторых случаях упростить реализацию, поскольку при последовательной обработке данных пакета проще сначала обновить поле Timestamp, а затем - Checksum Complement вместо обновления временной метки и последующего обновления поля UDP Checksum в заголовке UDP. Отметим, что несмотря на невозможность аппаратной реализации модуля меток времени, обновляющего UDP Checksum, использование Checksum Complement может существенно упростить реализацию.

Отметим, что программный уровень и промежуточный элемент (Рисунок 1) являются модулями одних часов NTP. Это предполагает согласованность обоих модулей в части включения поля Checksum Complement при передаче пакетов NTP.

В [RFC7820] определён механизм Checksum Complement для протоколов односторонних активных измерений (One-Way Active Measurement Protocol или OWAMP) и двухсторонних активных измерений (Two-Way Active Measurement Protocol или TWAMP). Похожий механизм представлен в приложении E к [IEEE1588].

2. Используемые соглашения

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), **возможно** (MAY), **необязательно** (OPTIONAL) в данном документе интерпретируются в соответствии с [KEYWORDS].

2.2. Сокращения

MAC
Message Authentication Code - код проверки подлинности сообщения.

NTP
Network Time Protocol - протокол сетевого времени.

PTP
Precision Time Protocol - протокол точного времени.

UDP
User Datagram Protocol - протокол пользовательских дейтаграмм.

3. Применение UDP Checksum Complement в NTP

3.1. Обзор

UDP Checksum Complement представляет собой 2-октетное поле, добавляемое в конец данных UDP, с использованием поля расширения NTP. На рисунке 2 показан формат пакета NTP с расширением Checksum Complement.

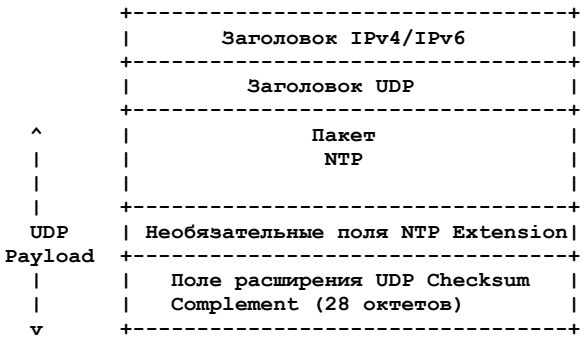

Рисунок 2. Checksum Complement в пакетах NTP.

Поле Checksum Complement служит для компенсации изменений, внесённых в пакет NTP промежуточными элементами, как описано в разделе 1. Пример использования Checksum Complement представлен в Приложении A.

3.2. Checksum Complement в пакетах NTP

Пакеты NTP передаются по протоколу UDP с использованием IPv4 или IPv6. Этот документ применим к обоим вариантам использования NTP.

Пакеты NTP могут включать поля расширения, как определено в [NTPv4]. Checksum Complement в пакетах NTP размещается в выделенном поле расширения NTP, как показано на рисунке 3.

Если пакет NTP включает более одного поля расширения, Checksum Complement всегда является последним полем. Таким образом, поле Checksum Complement - это 2 последних октета данных UDP, расположенные со смещением (UDP Length - 2) от начала заголовка UDP. Отметим, что Checksum Complement не применяется в аутентифицированных пакетах NTP (см. параграф 3.4).

3.2.1. Использование Checksum Complement

Как указано в разделе 1, промежуточный элемент, который обновляет временную метку в пакете NTP, может использовать Checksum Complement для сохранения корректности поля UDP Checksum. В частности. При обновлении временной метки это вызывает изменение значения UDP Checksum. Таким образом, промежуточный элемент задаёт новое значение Checksum Complement для компенсации этого изменения, сохраняя корректность текущего поля UDP Checksum. Пример использования Checksum Complement представлен в Приложении A.

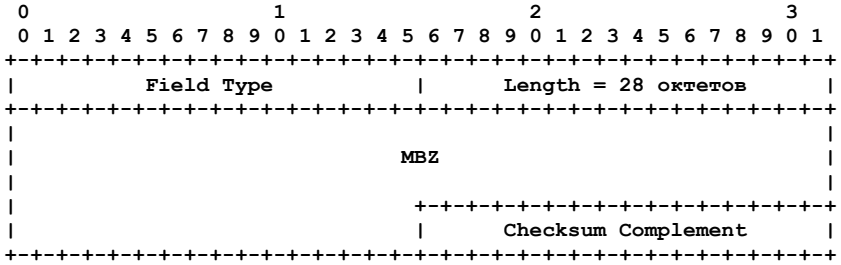

Рисунок 3. Поле расширения NTP Checksum Complement.

Field Type
Для указания расширения Checksum Complement применяется выделенное значение Field Type (раздел 5).

Length
Размер поля расширения Checksum Complement составляет 28 октетов. Это гарантирует корректный анали принятых пакетов получателем независимо от наличия в них кода MAC. Дополнительные детали относительно размера поля расширения при отсутствии MAC приведены в [RFC7822].

MBZ

Это поле расширения включает 22 октета MBZ (MUST be zero - должно быть 0). В этом поле отправитель **должен** устанавливать 0, а получатель **должен** игнорировать поле. Поле MBZ служит для заполнения поля расширения до 28 октетов.

Checksum Complement

расширение Checksum Complement включает поле Checksum Complement, размещающееся в 2 последних октетах.

3.2.2. Передача NTP с Checksum Complement

Передатчик пакета NTP **может** включать поле расширения Checksum Complement.

3.2.3. Обновление NTP с Checksum Complement

Промежуточный элемент, получающий и изменяющий пакет NTP с расширением Checksum Complement **может** использовать поле Checksum Complement для сохранения корректности значения UDP Checksum.

3.2.4. Приём NTP с Checksum Complement

Этот документ не задаёт дополнительных требований к приёму пакетов NTP.

Уровень UDP на приёмной стороне проверяет значение UDP Checksum в полученных пакетах NTP, а уровню NTP **следует** игнорировать поле расширения Checksum Complement.

3.3. Совместимость с имеющимися реализациями

Поведение, заданное в этом документе, не вносит дополнительных требований к поведению при получении тестовых пакетов NTP сверх заданных в [RFC7822]. Отметим, что в соответствии с [RFC7822] хосту, принявшему сообщение NTP с неизвестным полем расширения, **следует** игнорировать это поле и **можно** отбросить пакет, если правила требуют этого. Таким образом, передатчики и промежуточные элементы, поддерживающие Checksum Complement, могут взаимодействовать с узлами, не поддерживающими Checksum Complement, если эти получатели игнорируют неизвестные поля расширения. Замечено, что имеющиеся реализации, отбрасывающие пакеты с неизвестными полями расширения, не совместимы с передатчиками, использующими Checksum Complement.

Следует отметить, что при использовании аппаратных модулей временных меток они будут скорей всего применяться на обеих сторонах и, таким образом, оба хоста, участвующие в протоколе, будут поддерживать описанную здесь функциональность. Если модуль аппаратных меток использует лишь одна сторона, поле Checksum Complement может применяться лишь в том случае, когда известно, что другая сторона может воспринимать Checksum Complement.

3.4. Checksum Complement и проверка подлинности

Checksum Complement недопустимо применять при включённой проверке подлинности. Поле Checksum Complement полезно в режиме без аутентификации, позволяя промежуточным элементам последовательно обрабатывать пакет без записи и пересылки.

При использовании аутентификации промежуточный узел, изменяющий пакет NTP, должен пересчитать код MAC. В этом случае невозможно обновить поле Checksum Complement, поскольку это изменение требует повторного расчёта MAC и вносит циклическую зависимость между MAC и Checksum Complement. Т. е. при обновлении MAC необходимо обновить поле UDP Checksum, что делает поле Checksum Complement ненужным при использовании аутентификации.

4. Вопросы безопасности

Этот документ описывает использование расширения Checksum Complement для обеспечения корректности UDP Checksum. Вопросы безопасности протоколов передачи времени в целом рассмотрены в [SecTime], а вопросы безопасности NTP - в [NTPv4].

Целью этого расширения является упрощение реализации модулей точных временных меток, как показано на рисунке 1. Расширение предназначено для внутреннего использования клиентами или серверами NTP и не рассчитано на промежуточные коммутаторы или маршрутизаторы между клиентом и сервером. В отличие от PTP [IEEE1588], протокол NTP не требует промежуточных коммутаторов или маршрутизаторов для изменения содержимого пакетов NTP, поэтому такие изменение следует рассматривать как вредоносную MITM-атаку (man-in-the-middle).

Важно подчеркнуть, что описанная здесь схема не повышает уровень уязвимости протоколов к MITM-атакам. Злоумышленник MITM, злонамеренно изменяющий пакет и Checksum Complement в нем, логически эквивалентен злоумышленнику MITM, который меняет пакет и его поле UDP Checksum.

Описанная в документе концепция предназначена лишь для применения в режиме unauthenticated. Как указано в параграфе 3.4, при использовании криптографических механизмов защиты Checksum Complement не упрощает реализации со сравнению с традиционным расчётом контрольных сумм, поэтому Checksum Complement не применяется в этом случае.

5. Взаимодействие с IANA

Агентство IANA выделило в реестре NTP Extension Field Types новое значение

```
0x2005 Checksum Complement
```

5. Литература

5.1. Нормативные документы

[Checksum] Rijsinghani, A., Ed., "Computation of the Internet Checksum via Incremental Update", RFC 1624, DOI 10.17487/RFC1624, May 1994, <http://www.rfc-editor.org/info/rfc1624>.

[IPv6] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, DOI 10.17487/RFC2460, December 1998, <http://www.rfc-editor.org/info/rfc2460>.

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

[NTPv4] Mills, D., Martin, J., Ed., Burbank, J., and W. Kasch, "Network Time Protocol Version 4: Protocol and Algorithms Specification", RFC 5905, DOI 10.17487/RFC5905, June 2010, <http://www.rfc-editor.org/info/rfc5905>.

[RFC7822] Mizrahi, T. and D. Mayer, "Network Time Protocol Version 4 (NTPv4) Extension Fields", RFC 7822, DOI 10.17487/RFC7822, March 2016, <http://www.rfc-editor.org/info/rfc7822>.

[UDP] Postel, J., "User Datagram Protocol", STD 6, RFC 768, DOI 10.17487/RFC768, August 1980, <http://www.rfc-editor.org/info/rfc768>.

5.2. Дополнительная литература

[IEEE1588] IEEE, "IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems", IEEE Std 1588-2008, DOI 10.1109/IEEESTD.2008.4579760, July 2008.

[RFC7820] Mizrahi, T., "UDP Checksum Complement in the One-Way Active Measurement Protocol (OWAMP) and Two-Way Active Measurement Protocol (TWAMP)", RFC 7820, DOI 10.17487/RFC7820, March 2016, <http://www.rfc-editor.org/info/rfc7820>.

[SecTime] Mizrahi, T., "Security Requirements of Time Protocols in Packet Switched Networks", RFC 7384, DOI 10.17487/RFC7384, October 2014, <http://www.rfc-editor.org/info/rfc7384>.

[ZeroChecksum] Fairhurst, G. and M. Westerlund, "Applicability Statement for the Use of IPv6 UDP Datagrams with Zero Checksums", RFC 6936, DOI 10.17487/RFC6936, April 2013, <http://www.rfc-editor.org/info/rfc6936>.

Приложение A. Пример использования Checksum Complement

Рассмотрим пакет NTP, переданный от клиента серверу NTP.

Программный уровень клиента (Рисунок 1) генерирует пакет NTP с Origin Timestamp T и UDP Checksum U. Значение U учитывает заголовок и данные UDP, а также псевдозаголовок. Таким образом,

```
U = Const + checksum(T)                    (1)
```

где Const - контрольная сумма всех охватываемых полей, за исключением T.

Напомним, что программа отправителя выдаёт тестовый пакет с полем Checksum Complement, которое представляет собой просто 2 последних октета заполнения. В этом примере предполагается, что отправитель заполнил эти октеты нулями.

Модуль временных меток отправителя обновляет поле Timestamp точным знасением, меняя T на T', а также обновляет поле Checksum Complement, помещая вместо 0 значение C, так что

```
checksum(C) = checksum(T) - checksum(T')    (2)
```

Когда модуль временных меток отправителя передаёт пакет, значение контрольной суммы U остаётся прежним

```
U = Const + checksum(T) = Const + checksum(T) + checksum(T') -
    checksum(T') = Const + checksum(T') + checksum(C)    (3)
```

Таким образом, после обновления метки времени модулем меток значение U в пакете остаётся корректным.

Когда тестовый пакет приходит к получателю, тот выполняет обычный расчёт UDP Checksum и получает значение U. Поскольку Checksum Complement является частью заполнение, значение checksum(C) «прозрачно» включается в расчёт в соответствии с уравнением (3) без специальных действий сервера.

Благодарности

Автор признателен Danny Mayer, Miroslav Lichvar, Call Kyzivat, Suresh Krishnan, Brian Haberman за рецензии и полезные комментарии.

Адрес автора

Tal Mizrahi
Marvell
6 Hamada St.
Yokneam, 20692
Israel
Email: talmi@marvell.com

Перевод на русский язык

Николай Малых

nmalykh@protokols.ru