МЕЖДУНАРОДНЫЙ СОЮЗ ЭЛЕКТРОСВЯЗИ

МСЭ-Т X.1206

СЕКТОР СТАНДАРТИЗАЦИИ
ЭЛЕКТРОСВЯЗИ МСЭ

(04/2008)

СЕРИЯ X: СЕТИ ПЕРЕДАЧИ ДАННЫХ, ВЗАИМОСВЯЗЬ ОТКРЫТЫХ СИСТЕМ И БЕЗОПАСНОСТЬ

Безопасность электросвязи

Независимая от производителя структура автоматического сообщения связанной с безопасностью информации и распространения обновлений

Рекомендация МСЭ-Т X.1206

РЕКОМЕНДАЦИИ МСЭ-Т СЕРИИ X

СЕТИ ПЕРЕДАЧИ ДАННЫХ, ВЗАИМОСВЯЗЬ ОТКРЫТЫХ СИСТЕМ И БЕЗОПАСНОСТЬ

СЕТИ ПЕРЕДАЧИ ДАННЫХ ОБЩЕГО ПОЛЬЗОВАНИЯ	
Службы и услуги	X.1–X.19
Интерфейсы	X.20–X.49
Передача, сигнализация и коммутация	X.50–X.89
Сетевые аспекты	X.90–X.149
Техническое обслуживание	X.150–X.179
Административные предписания	X.180–X.199
ВЗАИМОСВЯЗЬ ОТКРЫТЫХ СИСТЕМ	
Модель и обозначение	X.200–X.209
Определения служб	X.210–X.219
Спецификации протоколов с установлением соединений	X.220–X.229
Спецификации протоколов без установления соединений	X.230–X.239
Проформы PICS	X.240–X.259
Идентификация протоколов	X.260–X.269
Протоколы обеспечения безопасности	X.270–X.279
Управляемые объекты уровня	X.280–X.289
Испытание на соответствие	X.290–X.299
ВЗАИМОДЕЙСТВИЕ МЕЖДУ СЕТЯМИ	
Общие положения	X.300–X.349
Спутниковые системы передачи данных	X.350–X.369
Сети, основанные на протоколе Интернет	X.370–X.379
СИСТЕМЫ ОБРАБОТКИ СООБЩЕНИЙ	X.400–X.499
СПРАВОЧНИК	X.500–X.599
ОРГАНИЗАЦИЯ СЕТИ ВОС И СИСТЕМНЫЕ АСПЕКТЫ	
Организация сети	X.600–X.629
Эффективность	X.630–X.639
Качество обслуживания	X.640–X.649
Наименование, адресация и регистрация	X.650–X.679
Абстрактно-синтаксическая нотация 1 (ASN.1)	X.680–X.699
УПРАВЛЕНИЕ В ВОС	
Структура и архитектура управления системами	X.700–X.709
Служба и протокол связи для общего управления	X.710–X.719
Структура управляющей информации	X.720–X.729
Функции общего управления и функции ODMA	X.730–X.799
БЕЗОПАСНОСТЬ	X.800–X.849
ПРИЛОЖЕНИЯ ВОС	
Фиксация, параллельность и восстановление	X.850–X.859
Обработка транзакций	X.860–X.879
Удаленные операции	X.880–X.889
Общие приложения ASN.1	X.890–X.899
ОТКРЫТАЯ РАСПРЕДЕЛЕННАЯ ОБРАБОТКА	X.900–X.999
БЕЗОПАСНОСТЬ ЭЛЕКТРОСВЯЗИ	**X.1000–X.1999**

Для получения более подробной информации просьба обращаться к перечню Рекомендаций МСЭ-Т.

Рекомендация МСЭ-Т X.1206

Независимая от производителя структура автоматического сообщения связанной с безопасностью информации и распространения обновлений

Резюме

В Рекомендации МСЭ-Т X.1206 предусматривается структура для автоматического сообщения связанной с безопасностью информации и распространения обновлений. Ключевой особенностью структуры является то, что она не зависит от производителя. После того как ресурс становится зарегистрированным, обновления информации об уязвимости, корректировки или обновления, касающиеся этого ресурса, могут автоматически стать доступными для пользователей или непосредственно для приложений.

Источник

Рекомендация МСЭ-Т X.1206 утверждена 18 апреля 2008 года 17-й Исследовательской комиссией МСЭ-Т (2005–2008 гг.) в соответствии с процедурой, изложенной в Резолюции 1 ВАСЭ.

ПРЕДИСЛОВИЕ

Международный союз электросвязи (МСЭ) является специализированным учреждением Организации Объединенных Наций в области электросвязи. Сектор стандартизации электросвязи МСЭ (МСЭ-Т) – постоянный орган МСЭ. МСЭ-Т отвечает за изучение технических, эксплуатационных и тарифных вопросов и за выпуск Рекомендаций по ним с целью стандартизации электросвязи на всемирной основе.

На Всемирной ассамблее по стандартизации электросвязи (ВАСЭ), которая проводится каждые четыре года, определяются темы для изучения Исследовательскими комиссиями МСЭ-Т, которые, в свою очередь, вырабатывают Рекомендации по этим темам.

Утверждение рекомендаций МСЭ-Т осуществляется в соответствии с процедурой, изложенной в Резолюции 1 ВАСЭ.

В некоторых областях информационных технологий, которые входят в компетенцию МСЭ-Т, необходимые стандарты разрабатываются на основе сотрудничества с ИСО и МЭК.

ПРИМЕЧАНИЕ

В настоящей Рекомендации термин "администрация" используется для краткости и обозначает как администрацию электросвязи, так и признанную эксплуатационную организацию.

Соблюдение положений данной Рекомендации носит добровольный характер. Однако в Рекомендации могут содержаться определенные обязательные положения (например, для обеспечения возможности взаимодействия или применимости), и соблюдение положений данной Рекомендации достигается в случае выполнения всех этих обязательных положений. Для выражения необходимости выполнения требований используется синтаксис долженствования и соответствующие слова (такие, как "должен" и т. п.), а также их отрицательные эквиваленты. Использование этих слов не предполагает, что соблюдение положений данной Рекомендации является обязательным для какой-либо из сторон.

ПРАВА ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ

МСЭ обращает внимание на вероятность того, что практическое применение или реализация этой Рекомендации может включать использование заявленного права интеллектуальной собственности. МСЭ не занимает какую бы то ни было позицию относительно подтверждения, обоснованности или применимости заявленных прав интеллектуальной собственности, независимо от того, отстаиваются ли они членами МСЭ или другими сторонами вне процесса подготовки Рекомендации.

На момент утверждения настоящей Рекомендации МСЭ не получил извещения об интеллектуальной собственности, защищенной патентами, которые могут потребоваться для выполнения этой Рекомендации. Однако те, кто будет применять Рекомендацию, должны иметь в виду, что это может не отражать самую последнюю информацию, и поэтому им настоятельно рекомендуется обращаться к патентной базе данных БСЭ по адресу: http://www.itu.int/ITU-T/ipr/.

СОДЕРЖАНИЕ

Рекомендация МСЭ-Т X.1206

Независимая от производителя структура автоматического сообщения связанной с безопасностью информации и распространения обновлений

1 Сфера применения

Настоящая Рекомендация предусматривает структуру двунаправленного потока автоматического сообщения и распространения информации об уязвимости, а также распространения обновлений версий и/или корректировок. Кроме того, настоящая Рекомендация позволяет системным администраторам узнать о состоянии любого ресурса в пределах их ответственности.

В пунктах 6 и 7 описаны проблемы сопровождения ресурса в аспекте идентификации ресурса, а также распространения информации и управления системами/сетями.

В пункте 8 приводится обзор независимой от производителя структуры, в которой описаны образец системы, обеспеченной такой структурой, действия структуры и примерная последовательность обменов в рамках структуры. В пункте 8 также рассматриваются вопросы безопасности, которые следует учитывать в независимой от производителя структуре.

Пункт 9 описывает функциональные характеристики и особенности данной Рекомендации.

В пункте 10 приводятся определения состава данных в компонентах данной Рекомендации.

В пункте 11 содержится схема XML, определение и описание которой приводятся в пункте 10.

Настоящая Рекомендация предусматривает структуру, которой может пользоваться любой производитель с целью сообщения и получения информации об уязвимости и распространения необходимых корректировок/обновлений для охватываемых ресурсов, и определяет формат информации, которая должна использоваться в компонентах, составляющих данную структуру, и между ними.

Настоящая Рекомендация не определяет протоколы, используемые в связи между компонентами, поскольку многие протоколы поддерживаются без специального учета.

Необходимо установить некоторые общие правила и обязательства для работы на основе независимой от производителя структуры; однако обсуждение вопроса об установлении и функционировании возможных ролей и вытекающих из них обязательств не входит в сферу применения настоящей Рекомендации.

2 Справочные документы

Не имеется.

3 Определения

3.1 Термины, получившие определения в настоящей Рекомендации

В настоящей Рекомендации определены следующие термины:

3.1.1 Агент: Осуществление настоящей Рекомендации с целью поддержки установленного ресурса на данном устройстве, обеспечения функций сервера или функций локального сервера.

3.1.2 Ресурс: Устройство, отдельно идентифицируемый предмет оборудования, приложение, операционная система или экземпляр исполняемой программы.

3.1.3 Клиент: Устройство, которое запрашивает услуги у другого устройства.

3.1.4 Устройство: Система, действующая как клиент, сервер, или как и то, и другое одновременно, либо как локальный сервер.

3.1.5 Группа: Несколько устройств, работающий как единый блок.

3.1.6 **Локальный сервер**: Клиент, действующий узел сервера для дополнительных клиентов в нисходящем направлении.

3.1.7 **Сообщение**: Запрос конкретного действия, которое следует выполнить, например, общие действия, такие как "Register" ("Зарегистрировать") ресурс как ресурс данной версии и/или встроенных компонентов данных версий, "Request" ("Запросить") существующие или будущие имеющиеся обновления, корректировки или информацию об уязвимости и пр. сообщения, расширяющие применимость настоящей Рекомендации, могут быть определены за рамками данной Рекомендации.

3.1.8 **Информация сообщения**: Информация, сопровождающая данное сообщение. Из почти бесконечного числа возможных вариантов настоящая Рекомендация определяет конкретные примеры, к которым относятся информация о версии определения данных, информация об уязвимости, касающаяся данных версий, а также обновления или корректировки для конкретных версий.

3.1.9 **Набор сообщений**: Комбинация и ассоциация универсально-однозначного идентификатора, сообщения и связанного с сообщением определения данных сообщения, которые описаны в схеме XML на основе и с расширением описанного в настоящей Рекомендации *Message_Core*.

3.1.10 **Патч**: Широко распространяемое средство для устранения уязвимости конкретного продукта в отношении безопасности. Метод обновления файла, который заменяет только изменяющиеся части, а не весь файл.

3.1.11 **Сервер**: Устройство, используемое для запросов в отношении обслуживания, поступающих с других устройств.

3.1.12 **Уязвимость**: Любой недостаток, административный процесс или действие или физическая подверженность воздействию, которые делают компьютер или сеть компьютеров восприимчивыми к угрозе.

4 Сокращения

В настоящей Рекомендации употребляются следующие сокращения и акронимы:

API	Application Programming Interface	Интерфейс прикладного программирования
GUID	Globally Unique IDentifier	Глобально уникальный идентификатор
HTTP	HyperText Transfer Protocol	Транспортный протокол передачи гипертекста
ISIRT	Information Security Incident Response Team	Группа реагирования на нарушения компьютерной защиты
ISP	Internet Service Provider	Поставщик услуг интернета
OS	Operating System	Операционная система
POAS	Platform/Operating System/Application/ Service	Платформа/Операционная система/Приложение/ Услуга
URI	Uniform Resource Identifier	Унифицированный идентификатор ресурса

5 Условные обозначения

Отсутствуют.

6 Введение

По мере того как люди пользуются компьютерами у себя дома и на работе, а доля тех, кто имеет хоть какую-то специальную подготовку в этой области, не говоря уже об аспектах безопасности, оказывается все меньше, быстро приближается тот момент, когда обеспечение безопасности станет практически невозможным и лицам, отвечающим за поддержание безопасности на уровне системы, будет все труднее узнать о состоянии систем, за которые они отвечают и которые они обслуживают, пока не случится какое-либо нарушение или авария, но тогда уже будет слишком поздно.

Такая ситуация вызвана в первую очередь тем фактом, что слишком большое число различных компьютеров находятся в различных состояниях обслуживания и обновления. При этом там, где затронуты вопросы безопасности, управление системой становится все меньше процессом предотвращения аварий, и все больше процессом управления авариями и восстановления после них.

Хотя ряд приложений и даже операционных систем (OS) имеют свои собственные механизмы обновления, все они страдают некоторыми общими для них недостатками. Одной из таких проблем является то, что все механизмы обновления сначала должны быть запущены, и лишь после этого они могут выполнить свою работу, если пользователь уведомлен о наличии обновления, при этом предполагается, что пользователь хотя бы включил уведомления.

Хотя, возможно, самым худшим является то, что до сих пор системные администраторы остаются не в курсе всех этих проблем, так что без установки своих собственных систем мониторинга на каждом компьютере, находящемся в их ведении, они не могут иметь представление об общем уровне безопасности внутри сетей и систем, за которые они несут ответственность.

Еще одно соображение состоит в том, что при обновлении программного обеспечения самыми последними версиями, часто случается так, что решением проблемы являются не сами по себе обновления, а усовершенствованные методы пользования, для которых бесполезны какие-либо обновления, кроме информации, получаемой конечным пользователем. Несмотря на то, что различные приложения и OS имеют механизмы обновления, ни одно из них не располагает унифицированным методом информирования пользователей о последних наиболее удачных практических способах обеспечения постоянной безопасности при пользовании.

Большое значение имеют также методы, применяемые при распространении обновлений. В настоящее время все различные механизмы обновления направляют новые версии по специально выделенным каналам – по одному для каждого сеанса обновления. Но если бы обновления или другая важная информация предоставлялись различным центрам распределения, например ПУИ или корпоративным сетям и тогда пакеты распределялись бы внутри сетей надежным и безопасным способом, пропускная способность, требуемая для распространения, могла бы быть эффективно уменьшена вдвое или, по меньшей мере, в значительной степени, как в случае модулей-посредников транспортного протокола передачи гипертекста (HTTP).

Еще одна проблема, которая в большинстве случаев решается неэффективно, связана с ситуацией, когда пользователи сталкиваются с трудностями пользования данным ресурсом или его действий, и им не к кому обратиться с вопросом или за информацией о решении возникшей проблемы. Даже если имеется какой-либо метод, позволяющий пользователям связаться с системными администраторами или другим обслуживающим персоналом, использование такой системы часто заканчивается так же, как игра "20 вопросов", то есть пользователь и обслуживающий сотрудник должны обращаться туда-сюда за запросом информации, в ответе на который обычно содержится больше вопросов.

Даже опытному в этих делах сотруднику службы поддержки бывает трудно добиться четкого представления о том, из чего именно состоят многие ресурсы. Из-за модульных архитектур программного обеспечения и различных элементов программ, фактически состоящих из модулей от различных производителей, которые используют свои собственные системы управления версиями, даже если имеется обновление или соответствующая часть информации о данном продукте, человеку может быть трудно узнать, к чему и кому они применяются, и поэтому часто менее информированные люди в целом игнорируют значительную часть механизмов, обеспечивающих безопасность их систем.

В итоге пользователи остаются неосведомленными, а те, кто отвечает за безопасность всей системы, по сути ничего не знают об этом.

7 Текущая ситуация в области обеспечения информации об уязвимости

В настоящее время информация об уязвимости предоставляется многими производителями и связанными с безопасностью организациями, такими как различные Группы реагирования на нарушения компьютерной защиты (ISIRT), которые стремятся посвятить пользователей в вопросы безопасности, а также предоставить им обновления и корректировки, если возникнет необходимость. Однако бывает так, что конечные пользователи не пользуются ни информацией, ни обновлениями, ни корректировками или, что даже не знают, применяется ли к ним все предоставленное.

Существуют различные причины для возникновения такой ситуации, но, прежде всего, требуется понять, почему конечные пользователи могут столкнуться с трудностями в использовании представленной им информации.

Обычно конечный пользователь пользуется информацией версии в отношении ресурсов, с тем чтобы определить, включают ли они систему, программное обеспечение или компоненты, которые затронуты недавно обнаруженной уязвимостью. Однако при пользовании информацией версии в целях выявления затронутых систем, программного обеспечения или компонентов возникают некоторые проблемы.

- Неоднозначность записи версии

Правила записи версии у разных производителей отличаются друг от друга. По причине таких различий конечный пользователь может интерпретировать информацию версии не так, как намеревался представить ее производитель. Не понимая конкретной системы версии производителя, конечные пользователи могут не узнать о том, что к ним применяется данное обновление, корректировка или информация.

В качестве примера можно привести ситуацию, когда существуют три версии системы – "v1", "v1.1" и "v1.2", которые описаны в информации об уязвимости как цель, и информация об уязвимости описана следующим образом:

"Затронутые системы

* XXXXv1 затронутая система".

Может иметь два различных смысла:

1) v1 – это только v1.0.
2) v1 означает v1.x, то есть все подверсии версии 1.

Попытки стандартизировать записи версии не дают практического результата, поэтому необходимо прочитать и понять конкретные правила производителя относительно записи версии, но было бы нереалистично надеяться на то, что конечные пользователи сделают это.

- Версия данного продукта не может всегда использоваться как критерий для определения того, что на самом деле может быть уязвимым.

У некоторых производителей версия продуктов, которые обычно известны пользователям, может оказаться не пригодной для определения того, имеет ли продукт уязвимые элементы (Рисунок 1).

В большинстве продуктов, ориентированных на компоненты, наличие уязвимости можно установить с помощью версии каждого компонента, из которого состоит продукт, а не версии самого продукта.

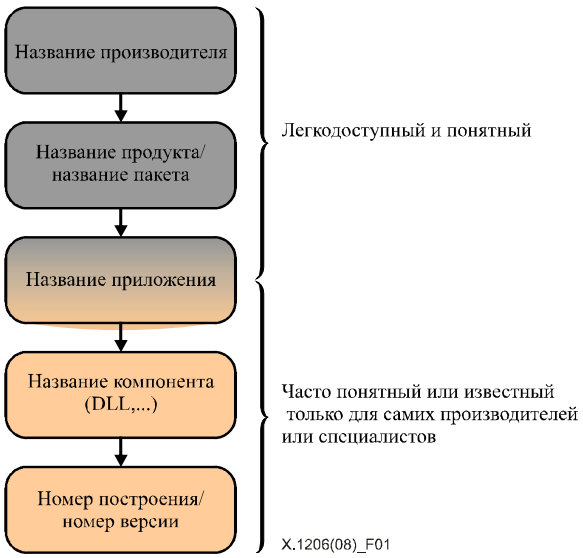


Рисунок 1 – Иерархическая структура "версии"

- Продукты, которые содержат в качестве компонентов продукты другого производителя

Значительная часть информации об уязвимости обычно относится к данному продукту. Конечный пользователь может использовать конкретный указанный продукт не напрямую, а через использование другого приложения, которое загружает или иным образом применяет услугу данного уязвимого продукта. На самом деле при наличии открыто публикуемого Интерфейса прикладного программирования (API) производитель продуктов, используемых в других приложениях, может иметь слабое представление или не знать вовсе о том, где используются его продукты или в каких других приложениях они могут быть загружены или применены иным образом.

- Системные администраторы не знают о состоянии ресурсов, находящихся в их ведении

Поскольку системные администраторы зависят от пользователей, обеспечивающих свои собственные системы, и без личной, самостоятельной проверки статуса каждой системы или использования самоотчетности невозможно знать статус систем, за которые они в конченом счете несут ответственность. Помимо механизма самоотчетности, которая часто бывает неточной и неполной по многим причинам, уже указанным в отношении контроля версий или приложений по контролю за системой заказчика, которые требуют разработки и сопровождения, лица, отвечающие за обеспечение безопасности корпоративных сетей или сетей поставщика услуг интернета (ПУИ), располагают немногими механизмами.

8 Обзор независимой от производителя структуры

В этом разделе содержится обзор независимой от производителя структуры распространения информации об уязвимости, обновлений и корректировок.

При внедрении структуры системы распространения информации об уязвимости, обновлений и корректировок, пример которых приводится на Рисунке 2, могут быть конструированы с использованием простой схемы абонирования. Каждый ресурс, устройство или локальный сервер может зарегистрироваться на любом и всех имеющихся серверах для получения запрашиваемой (вытягивание) или предлагаемой (выталкивание) информации об уязвимости, обновлений и/или корректировок.

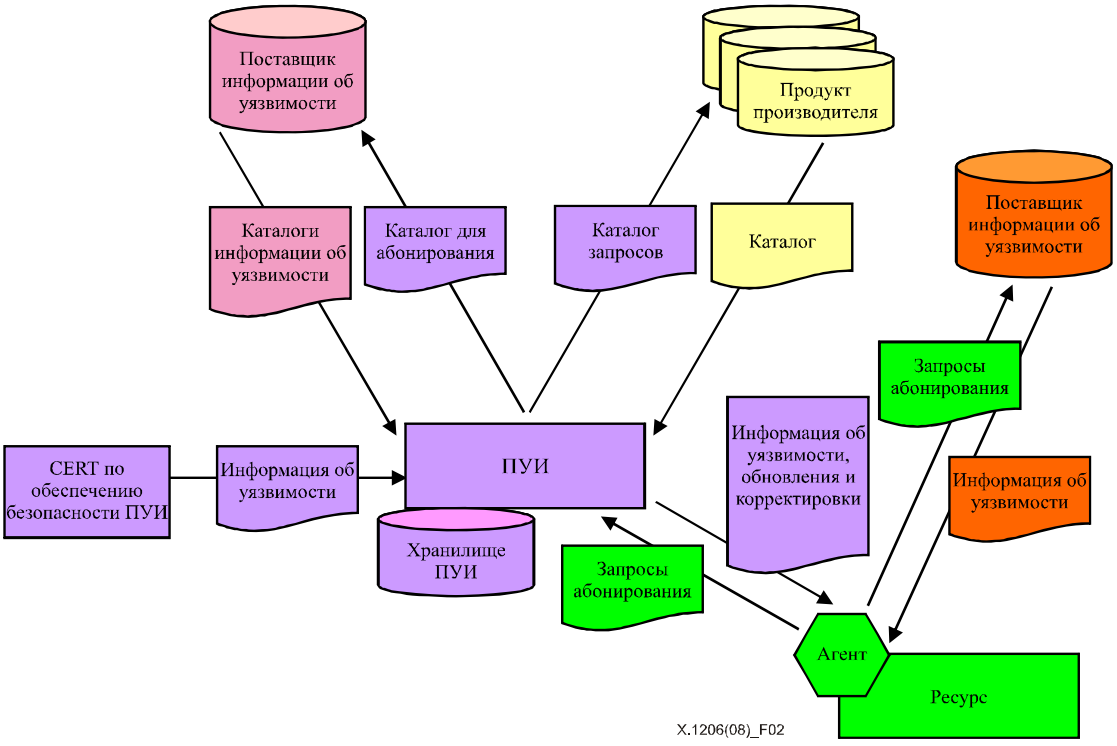


Рисунок 2 – Примерная архитектура приложения

8.1 Множественные источники информации об уязвимости, обновлений и корректировок

Применяя такую схему сообщений, любой орган может делать запросы в отношении информации об уязвимости, обновлений и/или корректировок для любого другого органа и/или предоставлять их. В этом конкретном примере ПУИ действует как коллективный пункт, выступающий в качестве источника для всех своих абонентов. Кроме того, отдельные абоненты могут зарегистрироваться у независимого третьего лица, являющегося источником обновлений информации, анализа компьютерной программы или даже обновлений или корректировок для программы. Разумеется, любому приложению, пытающемуся применить данное обновление или корректировку, необходимо принять это обновление или корректировку как санкционированную, но способы достижения этого, кроме как использования поддержанного включения подписей, в настоящей Рекомендации не рассматриваются.

8.2 Примерное действие приложения

8.2.1 Процесс направления запросов и доставки информации и обновлений

Ресурс (например, текстовой редактор) устанавливается, и во время установки направляется уведомление установленному прежде Агенту. На основе информации, предоставленной агенту во время установки ресурса, агент направляет ПУИ пользователя запрос на абонирование в отношении любой и всякой информации об уязвимости, обновлений и/или корректировок.

До или после получения запроса на абонирование от агента пользователя ПУИ связывается с различными источниками информации об уязвимости, обновлений и корректировок на основе информации, предоставленной агентом в отношении источника или производителя нового установленного приложения.

Кроме того, агент пользователя имеет возможность, в данном примере, отдельно запросить другие источники информации об уязвимости и представить пользователю любую полученную информацию для ознакомления. Осуществить эту последнюю процедуру можно через веб-сайт, рекламирующий услуги источника информации об уязвимости, и через канал загрузки, или посредство отправки сообщения, как определено в настоящей Рекомендации, на основе которого сможет действовать впоследствии агент.

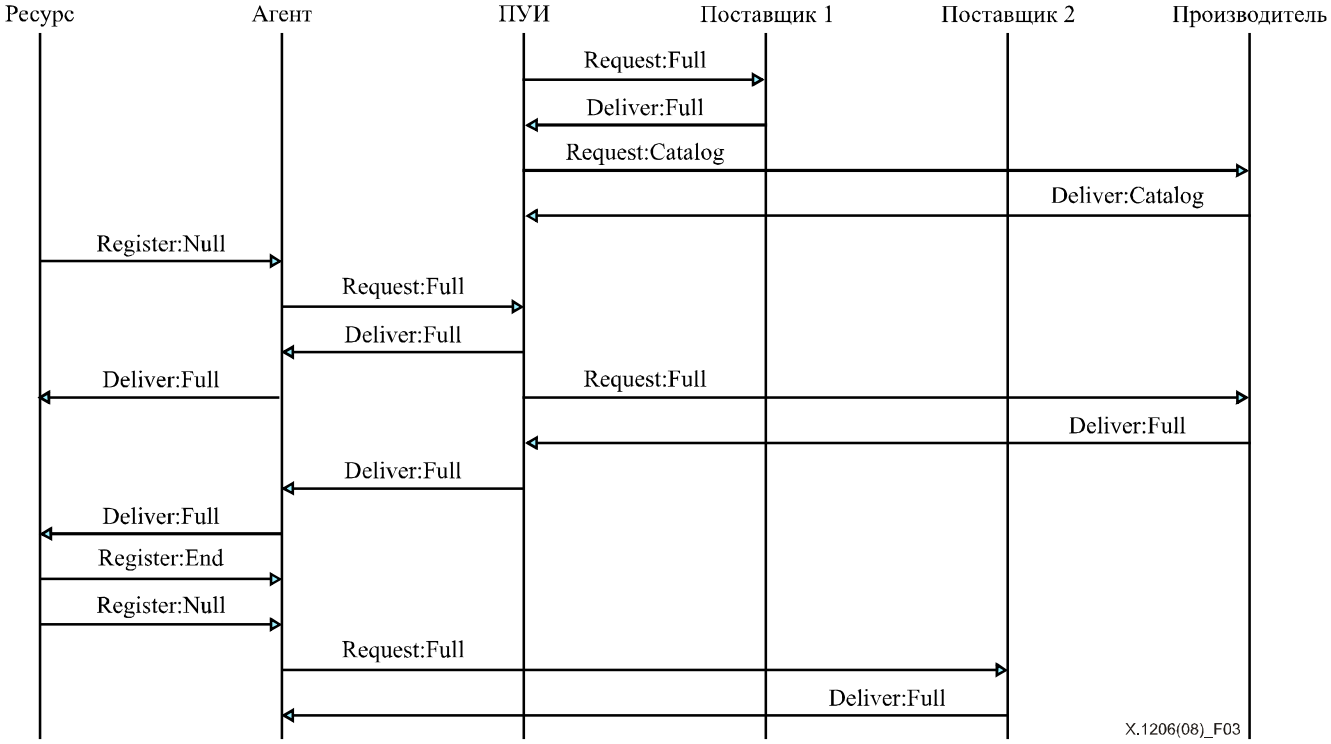


Рисунок 3 – Примерные сообщения приложения

8.2.2 Сообщение в отношении запросов и доставки информации и обновлений

ПУИ, который вполне может быть администратором корпоративной или частной сети, делает запросы у различных поставщиков информации об уязвимости, обновлений и корректировок. ПУИ может запросить либо полное содержание всего что имеется, с тем чтобы действовать в качестве локального архива с самого начала операции, либо может просто запросить список (Каталог) имеющейся информации об уязвимости, обновлений и корректировок, из которого впоследствии он может запросить отдельные элементы в полном объеме. Здесь не показано, что ответом на запрос в отношении полного содержания может быть "Major Message not Supported" ("Основное сообщение не поддерживается") или "Minor Message not Supported" ("Второстепенное сообщение не поддерживается"), и в таком случае запрашивающая служба будет пытаться запросить вместо него версию списка (Каталога). Однако предпочтительнее, чтобы поставщики делали свои правила в отношении запросов общедоступными, с тем чтобы избежать необходимости в согласовании.

В данном конкретном примере ПУИ запрашивает у производителя только каталог, поскольку полный архив информации производителя об уязвимости, обновлениях и/или корректировках может быть весьма объемным. Но так как запрос каталога действует также как подписка на всю будущую информацию, обновления и корректировки, ПУИ может получить заверения, что все доставляемое им агентам, действующим от имени своих пользователей, будет самым новым. С другой стороны, поскольку поставщик информации об уязвимости, например поставщик 1, вероятно, располагает меньшим объемом данных в онлайновом режиме, может быть практичнее запросить все имеющееся.

В процессе установки ресурса локально установленному агенту может быть направлено сообщение *Request:Full* (Запрос:полный) (виды основного/второстепенного сообщения) с запросом всей имеющейся и будущей информации об уязвимости, обновлений и корректировок по мере их поступления для специфической версии ресурса или любого из его встроенных компонентов.

Затем агент направляет сообщение *Request:Full* в адрес ПУИ от имени нового установленного ресурса.

Поскольку в этом примере ПУИ уже переносит информацию об уязвимости от поставщика 1 в свой архив, он может доставить ее мгновенно или ждать, пока не будет получен ответ от производителя и любая и вся информация не будет объединена и затем доставлена.

После того как ПУИ установит из своего хранимого каталога производителя как источник применимой к ресурсу информации, ПУИ проверяет свой кеш, чтобы узнать, имеется ли информация на локальном уровне, и, если не имеется, делает запрос у производителя в отношении имеющейся информации в полном объеме, которую он затем доставляет агенту, при этом при желании помещая информацию в свой собственный архив.

Через интерфейс пользователя с агентом пользователь может поместить полезный источник информации об уязвимости самостоятельно и, возможно, с использованием заранее составленного запроса у поставщика, например, поставщик 2, которого пользователь мог загрузить, поручает агенту запросить либо каталог, либо фактически имеющуюся информацию об уязвимости, либо информацию, которая появится в будущем.

После повышения уровня данного ресурса или его компонента запросы на обновления предыдущей версии аннулируются с использованием *Register:End* (*Регистрация:конец*), а запросы в отношении применимых обновлений информации для новых версий могут направляться с использованием *Register:null* (*Регистрация: пусто*). Аналогичным образом, если ресурс снимается с обслуживания, будущие обновления информации могут быть аннулированы также с использованием *Register:End*.

8.3 Аспекты безопасности и секретности

В рамках настоящей Рекомендации информация пользователя не передается на сторону производителя. Однако необходимо рассмотреть вопрос об аутентификации и проверке целостности доставленной информации, в связи с чем предлагаются следующие методы:

- Источники должны сами себя идентифицировать с использованием подписей в сообщениях.
- Агенты должны проверять аутентичность и целостность доставленных сообщений.
- Обновления не должны иметь индивидуализированной основы (не персонализированы), так что в сеть никакой специфической информации о системе конечного пользователя не доставляется.
- Для загрузки или установки на оборудовании конечного пользователя требуется подтверждение конечного пользователя.

9 Рекомендуемая архитектура

9.1 Уровень памяти сообщения

Память сообщения обеспечивает связь, независимую от POAS (Платформа/Операционная система/Приложение/Услуга), между приложениями соответствующего сервера и клиента, построенными согласно требованиям, зависящим от POAS. Уровень памяти сообщения абстрагирует до реализации, согласующейся с конкретным сообщением, сложности, связанные с определением способа выполнения данной операции, в результате чего администратору необходимо лишь сосредоточиться на том, какие операции следует выполнять.

9.2 Уровень сообщения/Приложения

Уровень сообщения/приложения состоит из двух главных элементов. Первый элемент – это независимые процессы или функции POAS, которые обычно выполняются во время операций ресурса или устройства, например, запроса информации об уязвимости, обновлений или корректировок. Эти операции относятся к более высокому уровню и применяются к системам и функциям, независимо от того, как реализуется данная система. Например, все системы нуждаются в наличии касающихся их обновлений или периодической корректировки либо информации об уязвимости, и пользователи должны знать об этом.

Проблема заключается в том, что практически каждая отличная от другой система выполняет эти, точно такие же функции высокого уровня абсолютно разными способами на уровне реализации. В предлагаемой Рекомендацией архитектуре, в которой независимые от POAS требования противоречат реализациям более низкого уровня, и создается первичный интерфейс.

В целом идентификатор предназначается для операций/процессов высокого уровня, которые должны быть выполнены в системах независимо от вида системы. Затем системы различных видов используют эти идентификаторы для сообщения указаний друг другу. По получении того или иного указания получающая система просто передает это указание вместе с любыми связанными с ним данными специальному обрабатывающему устройству системы, предназначенному для обработки этого конкретного указания.

Во многих отношениях это не отличается от обработки и представления потока сжатого видеоизображения или считывания файла, такого как настоящий документ, на множестве различных платформ. Каждый зависимый от платформы видеоплеер или устройство считывания текстовых файлов отвечает за обработку полученных данных на основе соответствующих стандартов.

9.3 Универсальность

Универсальность систем, реализуемых на основе применения настоящей Рекомендации, поддерживается через архитектуру идентификации топографического сервера/клиента, которая поддерживает клиента одного сервера, являющегося сервером для других клиентов ниже. На самом деле поддерживается любое топографическое группирование серверов и клиентов, что можно описать узлами на древовидном графике. Кроме того, способность комплектовать сообщения для доставки на основе принадлежности к конкретному клиенту или группе позволяет любому серверу на любом уровне выше данного клиента связываться и взаимодействовать с этим клиентом.

- Клиенты и серверы идентифицируются/присваивают Глобально уникальный идентификатор (GUID) длиной 128 битов.
- GUID может быть заранее присвоен или повторно присвоен клиентам, или то и другое одновременно, при регистрации на данном сервере.
- В зависимости от топографических правил клиенту могут быть присвоены или использованы им другие GUID для взаимодействия с различными серверами.
- Если клиент сам является сервером для других клиентов (локальный сервер), то локальный сервер может "Затенять" настоящий GUID сервера, которому он отчитывается, и наоборот.
- Клиенты могут отчитываться/быть назначены нескольким серверам, каждый из которых отвечает за предоставление различных наборов услуг.

9.4 Расширяемость

Новые платформы, приложения, функции и услуги могут быть добавлены просто посредством создания схемы на основе применения и расширения схемы памяти сообщения, предусмотренной в настоящей Рекомендации.

Затем из определенной схемы сообщения могут быть созданы модули для любой желаемой платформы с гарантией возможности полного взаимодействия.

На основе сообщений новой схемы сообщений и структур данных сообщения могут быть легко и оперативно созданы интерфейсы пользователя, в соответствии с потребностями и пожеланиями.

Общие сообщения могут быть "повторно использованы" для новых комплектов сообщений посредством импортирования либо существующего определения сообщения, либо структуры данных сообщения, либо того и другого, в зависимости от потребностей.

9.5 Независимость от платформы

- Рекомендация представляет собой "Нормативный" интерфейс между всеми участвующими органами.
- Модули, реализующие Рекомендацию, созданы на зависимой от POAS основе с целью реализации сообщений, функций и услуг по мере необходимости.
- Любое участвующее устройство может выпускать любое обозначенное сообщение или запрос любому другому участвующему устройству, независимому от платформы, на которой реализуется каждое из этих устройств. В отдельных случаях содержащиеся данные сообщения могут или вероятно будут содержать специфический для платформы контент, но во всех случаях Рекомендация предназначена для устранения необходимости в знании всех платформ. Рекомендация поддерживает только одно устройство для передачи команды другому устройству, что делать, а не как делать.

9.6 Связь между клиентом и сервером

9.6.1 Открытый протокол

- На основе XML.

9.6.2 Режимы связи между сервером и клиентом

- Выталкивание от сервера
  - Сервер направляет уведомления и обновления по мере того, как они стали доступны клиенту, в отношении которых зарегистрировался клиент.
- Вытягивание в сервер
  - Сервер направляет клиенту запросы в отношении операций, уведомлений и обновлений.
- Выталкивание от клиента
  - Клиент направляет серверу уведомления и обновления
- Вытягивание клиенту
  - Клиент направляет серверу запросы в отношении операций, уведомления и обновлений.

10 Компоненты структуры

Настоящая Рекомендация формулируется с использованием XML и содержит информацию об архитектуре передачи сообщений, обеспечивающей адресацию источников и пунктов назначения сообщений, перенос сообщений, а также соответствующих данных сообщения.

В данном разделе предусмотрены определения формата данных для главных компонентов: Контейнер сообщения, сообщение о регистрации и сообщение о версии.

10.1 Контейнер сообщения

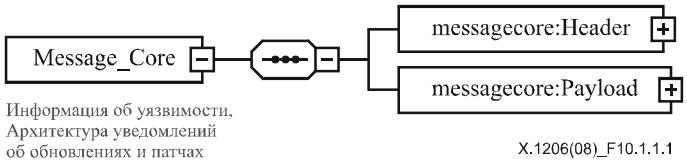
Контейнер сообщения играет роль как конверта для переноса, маршрутизации и доставки сообщений и ответов, так и абстрактной базы для создания других сообщений и ответов посредством расширения и вывода.

10.1.1 Message_Core

Все сообщения, используемые в архитектуре данной Рекомендации, определяются посредством импорта, расширения и вывода из *Message_Core*.

Корневым элементом структуры сообщения является Message_Core.

10.1.1.1 Синтаксис



10.1.1.2 Семантика

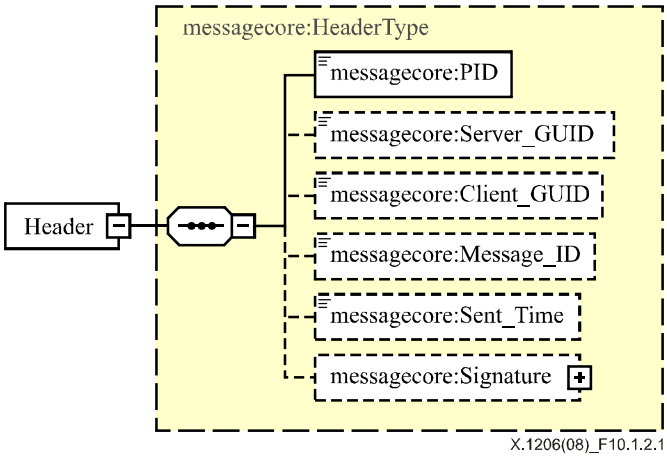
Header – Содержит общую информацию о маршрутизации и идентификации сообщения. [ОБЯЗАТЕЛЬНО]

Payload – Содержит **выбор** XML одного или нескольких групповых сообщений *Message_Core* или единственное сообщение. Для использования в ситуациях, когда сообщения могут быть помещены в кеш на данном узле (локальный сервер) или когда сервер, контролирующий/обслуживающий локальный север, может пожелать получить сообщения и данные сообщения от клиентов, контролируемых/обслуживаемых локальным сервером. [ОБЯЗАТЕЛЬНО]

10.1.2 Заголовок

Заголовок сообщения содержит ID протокола в целях расширяемости и возможности взаимодействия, адресацию информации, такую как источник(и) и адресат(ы) в отношении данного сообщения, поле для переноса ID сообщения и поле для переноса информации о времени отправления сообщения.

10.1.2.1 Синтаксис



10.1.2.2 Семантика

PID – ID протокола. Величина 128 битов, используемая для идентификации конкретной версии и/или расширения до Протокола сообщения или полной его замены, которая, как минимум, импортирует *Message_Core*, *Header* и *PID*. Для того, чтобы сигнал ответствовал данной Рекомендации, в приложениях может использоваться величина #h00000001. [ОБЯЗАТЕЛЬНО]

Server_GUID – Величина 128 битов, используемая клиентами для однозначной идентификации либо сервера, либо локального сервера, на который клиенты непосредственно отправляют отчет. [ФАКУЛЬТАТИВНО]

Client_GUID – Величина 128 битов, используемая серверами и локальными серверами для однозначной идентификации клиентов, отправляющих на них отчет. [ФАКУЛЬТАТИВНО]

Message_ID – Величина 128 битов, используемая для однозначной идентификации данного сообщения. Хотя метод выбора соответствующего *Message_ID* не является нормативным, предполагается, что равноправный объект данной системы будет использовать в последующих ответах такую же величину *Message_ID*. [ФАКУЛЬТАТИВНО]

Sent_Time – Указывает время, когда было отправлено данное сообщение. [ФАКУЛЬТАТИВНО]

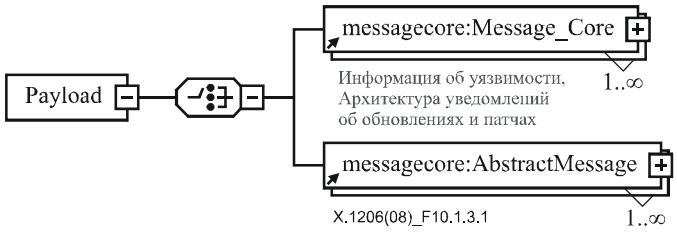
Signature – Упакованная подпись XML. [ФАКУЛЬТАТИВНО]

10.1.3 Информационное наполнение

Функциональные характеристики, связанные с содержанием данного *Информационного наполнения*, определяются описанием и ассоциацией *UUID* и/или *CID* с *Сообщением* и связанным с *Сообщением Message_Data*. *UUID/CID*, *Сообщение* и *Message_Data* являются главными элементами расширяемости в архитектуре, предусмотренной в настоящей Рекомендации, и именно они делают эту архитектуру независимой от платформ. Заданное сообщение и связанное с ним *Message_Data* называются Комплектом сообщения.

Описывая и ассоциируя *UUID* и/или *CID* с описанием схемы XML для комплекта сообщения, которая импортирует и расширяет схему *Message_Core*, и реализуя модуль для данного комплекта сообщения, который включает схему *Сообщения* и *Message_Data*, модули могут быть загружены в любое соответствующее Рекомендации приложение с гарантией возможности полного взаимодействия.

10.1.3.1 Синтаксис



X.1206(08)_F10.1.3.1

10.1.3.2 Семантика

Message_Core – Обеспечивает возможность группировать и доставлять множественные сообщения из множественных источников. В случае выбора. [ОБЯЗАТЕЛЬНО]

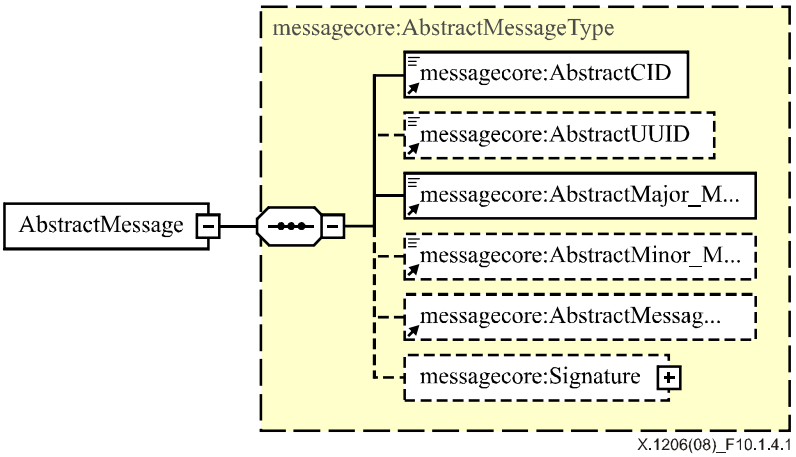
AbstractMessage – Соответствующее сообщение и идентификатор модуля, а также указание конкретного типа сообщения и данных конкретного сообщения. [ОБЯЗАТЕЛЬНО]

10.1.4 AbstractMessage

Структура данных используется для переноса информации с целью точной идентификации того, какое запрашивается сообщение и режим сообщения, но также и для переноса любых данных, необходимых для выполнения запрашиваемого сообщения.

Эта структура данных предназначена для расширения любым конкретным сообщением, описанным через нормативную схему, и по причине его реализации в базовой схеме как абстрактного класса она не может использоваться без расширяемого определения.

10.1.4.1 Синтаксис



10.1.4.2 Семантика

AbstractCID – Величина 128 битов, используемая для однозначной идентификации единственной данной функциональности. Данный реализуемый модуль, идентифицированный данным *UUID*, может реализовать поддержку для нескольких типов функциональности, идентифицированной посредством *CID*. И наоборот, несколько модулей могут реализовать одну и ту же функциональность, но для разных приложений, то есть модуля версии текстового процессора и модуля версии видеоредактора. [ОБЯЗАТЕЛЬНО]

AbstractUUID – Величина 128 битов, используемая для однозначной идентификации реализуемого модуля, который может обрабатывать сообщения, с которыми ассоциируется его ID. [ФАКУЛЬТАТИВНО]

*AbstractMajor_Messag*e – Цепочка, которая используется для переноса указаний либо с серверов клиентам, либо наоборот. Эти указания определяются в схемах, импортирующих и расширяющих схему *Message_Core*, и могут включать, но не только, "Register", "Request" и пр. [ОБЯЗАТЕЛЬНО]

AbstractMinor_Message – Дополнительная цепочка идентификатора сообщения для дальнейшего определения режима указанного сообщения. Такие указания описаны в схемах, импортирующих и расширяющих схему *Message_Core*, и могут включать, но не только, "Full", "Catalog" и пр. [ФАКУЛЬТАТИВНО]

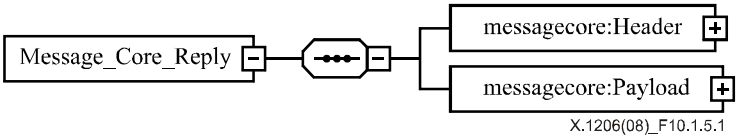
AbstractMessage_Data – Данные, требуемые для обработки запрошенного сообщения. [ФАКУЛЬТАТИВНО]

Signature – Упакованная подпись XML. [ФАКУЛЬТАТИВНО]

10.1.5 Message_Core_Reply

Структура данных *Message_Core_Reply* используется для переноса ответа в отношении статуса и любых дополнительных данных по желанию на любой объект, который инициирован в данном потоке связи.

10.1.5.1 Синтаксис



10.1.5.2 Семантика

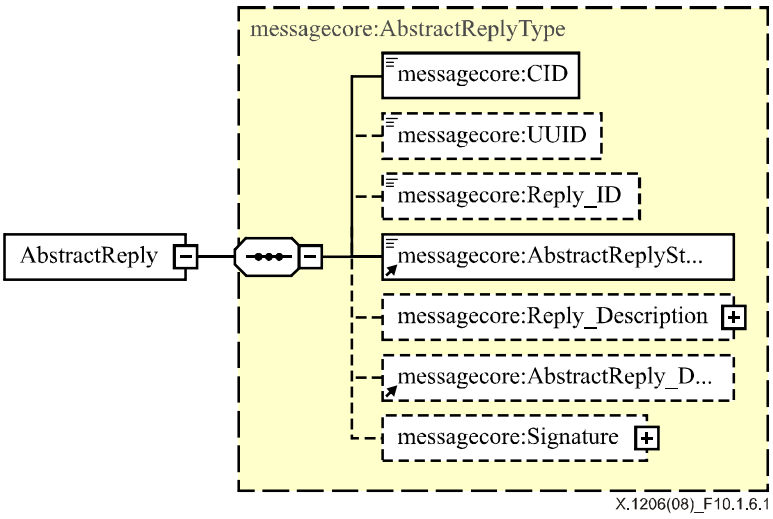
Header – Структура, данные и требования, идентичные *Message_Core:Header*.

Payload – Содержит выбор XML одного или нескольких групповых сообщений *Message_Core_Reply* или единственное сообщение *Message_Core_Reply*. Для использования в ситуациях, когда сообщения могут быть помещены в кеш на данном узле (локальный сервер) или когда сервер, контролирующий/обслуживающий локальный север, может пожелать получить сообщения и данные сообщения от клиентов, контролируемых/обслуживаемых локальным сервером. [ОБЯЗАТЕЛЬНО]

10.1.6 Abstract_Reply

Структура данных *Abstract_Reply* используется для переноса ответа в отношении статуса и любых дополнительных данных по желанию на любой объект, который инициировал данный поток связи.

10.1.6.1. Синтаксис



10.1.6.2 Семантика

CID – ID сообщения. Величина 128 битов, используемая для идентификации конкретной версии и/или расширения до протокола сообщения или полной его замены, которая, как минимум, импортирует *Message_Core*, *Header* и *PID*. Для того, чтобы сигнал ответствовал данной Рекомендации, в приложениях может использоваться величина #h00000001. [ОБЯЗАТЕЛЬНО]

UUID – Величина 128 битов, используемая для однозначной идентификации реализуемого модуля, который может обрабатывать ответы, с которыми ассоциируется его ID. [ФАКУЛЬТАТИВНО]

Reply_ID – Величина 128 битов, используемая для однозначной идентификации данного ответа. Хотя метод выбора соответствующего *Reply_ID* не является нормативным, предполагается, что равноправный объект данной системы будет использовать такую же величину, как и *Message_ID*, в полученных запросах, на которые дается данный ответ. [ФАКУЛЬТАТИВНО]

AbstractReplyString – Расширено сообщениями, импортирующими схему *Message_Core*, и содержит цепочки, указывающие статус предыдущего запроса, на которые дается ответ. Такими цепочками могут быть, но не только, "Failed", "Succeeded", "Message not supported" и пр. [ОБЯЗАТЕЛЬНО]

Reply_Description – Простой текст или HTML с изложением дополнительной информации относительно *AbstractReplyString*. [ФАКУЛЬТАТИВНО]

AbstractReply_Data – Любые требуемые данные, зависимые от реализации. [ФАКУЛЬТАТИВНО]

Signature – Упакованная подпись XML. [ФАКУЛЬТАТИВНО]

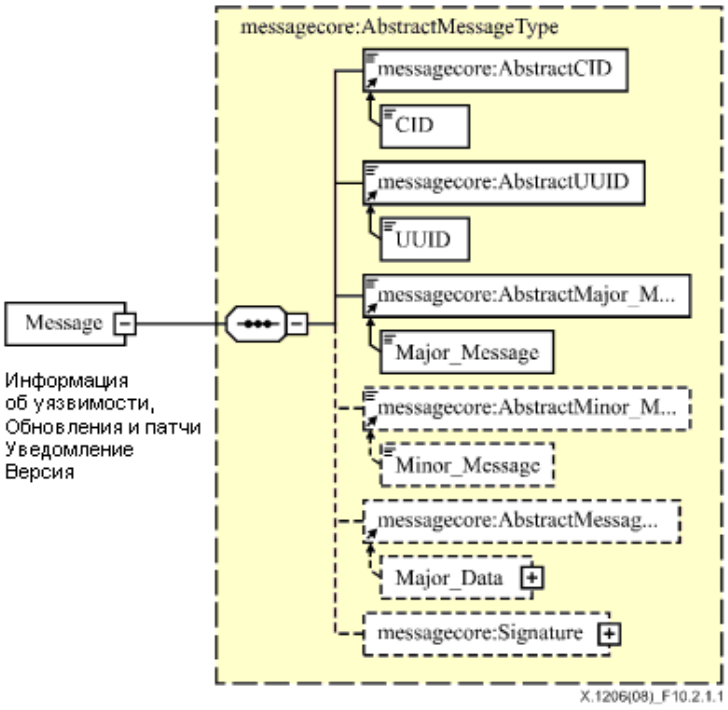
10.2 Сообщение версии

Сообщение версии используется между компонентами системы с целью запроса и предоставления информации об уязвимости, а также обновлений и/или корректировок.

10.2.1 Version:Message

Version:Message предназначено для прямой замены любого местоположения, где существует *AbstractMessage*, и его структура данных ограничена и расширена для однозначной поддержки процесса предоставления, запроса и доставки информации об уязвимости, обновлений и/или корректировок.

10.2.1.1 Синтаксис



10.2.1.2 Семантика

CID – Переопределяет и ограничивает базовый класс *Message_Core:AbstractCID* конкретной установленной величиной #h02. Однозначно идентифицирует это сообщение как принадлежащее к классу сообщений версии. [ОБЯЗАТЕЛЬНО]

UUID – Переопределяет и ограничивает базовый класс сообщения *Message_Core:AbstractUUID* и используется для однозначной идентификации реализуемого модуля, который может обрабатывать *сообщения*, с которыми ассоциируется его ID. [ОБЯЗАТЕЛЬНО]

Major_Message – Переопределяет и ограничивает базовый класс *Message_Core:AbstractMajor_Message* пронумерованным перечнем возможных значений "Register", "Request", "Deliver", "Analyse". [ОБЯЗАТЕЛЬНО]

Значения используются в следующем порядке:

- "Register" – Уведомить получающего клиента о том, что ресурс и его версия, как они описаны в Version:Message_Data, используются и что информация об уязвимости, обновления и корректировки должны быть доставлены по мере их поступления.
- "Request" – Запросить имеющуюся информация об уязвимости, обновления и корректировки для любых продуктов, которые в минимальной степени соответствуют требованиям, указанным в Version:Message_Data.
- "Deliver" – Информация об уязвимости, одно или несколько обновлений или корректировок включаются в Version:Message_Data.
- "Analyze"— Информация, содержащаяся в Version:Message_Data, запрашивается с целью проведения анализа. Причина запроса может быть известна заранее, или может быть дано объяснение в version:Info – version:Description или version:Info – version:Container – version:Description, в зависимости от того, что из них больше подходит к приложению.

Minor_Message – Переопределяет и ограничивает базовый класс *Message_Core:AbstractMinor_Message* пронумерованным перечнем возможных значений "Full", "Catalog" и "End". [ФАКУЛЬТАТИВНО]

Значения используются в следующем порядке:

- "Full" – Запрос всех имеющихся и применимых обновлений и/или корректировок.
- "Catalog" – Доставить список информации об уязвимости, обновлений и/или корректировок, которые имеются для продуктов, указанных в Version:Message_Data.
- "End" – Закончить абонирование в соответствии с информацией, содержащейся в Version:Message_Data.

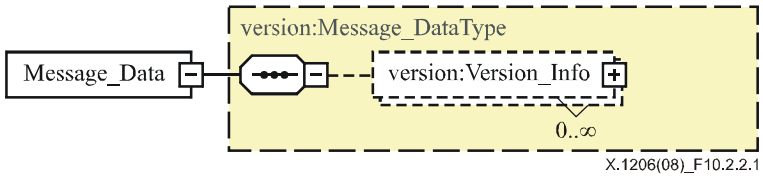
Message_Data – Переопределяет и ограничивает базовый класс *Message_Core:AbstractMajor_Message_Data* структурой конкретных данных, используемой для переноса идентификации версии продукта, а также информации об уязвимости, обновлений и/или корректировок. [ФАКУЛЬТАТИВНО]

Signature – Упакованная подпись XML. [ФАКУЛЬТАТИВНО]

10.2.2 Version:Message_Data

Version:Message_Data предназначено для прямой замены любого местоположения, где существует *AbstractMessage_Data*, и его структура данных ограничена и расширена для однозначной поддержки процесса переноса версии продукта, информации об уязвимости, обновлений и/или корректировок.

10.2.2.1 Синтаксис



X.1206(08)_F10.2.2.1

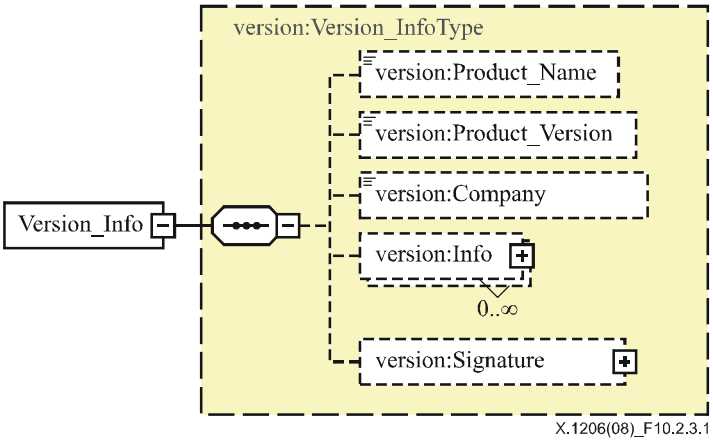
10.2.2.2 Семантика

Version_Info – Информация, используемая для идентификации, до требуемого уровня конкретности, продуктов и их версий, а также поддержки переноса информации об уязвимости, обновлений и/или корректировок.

10.2.3 Version_Info

Используется не только для идентификации продуктов, вместе с их производителем, версией и информацией, относящейся к версии, но и для переноса информации об уязвимости, обновлений и/или корректировок, необходимых для распространения между нуждающимися в них приложениями.

10.2.3.1 Синтаксис



10.2.3.2 Семантика

Product_Name – Название продукта высокого уровня, к которому относится информация идентифицированной версии, например, для данного дисковода DVD, который содержит конкретную версию флэш-системы, *Product_Name* будет указывать на название продукта дисковода DVD. [ФАКУЛЬТАТИВНО]

Product_Version – Версия продукта высокого уровня, к которому относится информация идентифицированной версии. Относительно предыдущего примера *Product_Version* будет указывать на конкретную версию дисковода DVD. [ФАКУЛЬТАТИВНО]

Company – Производитель, поставщик или автор идентифицированного продукта или ресурса. [ФАКУЛЬТАТИВНО]

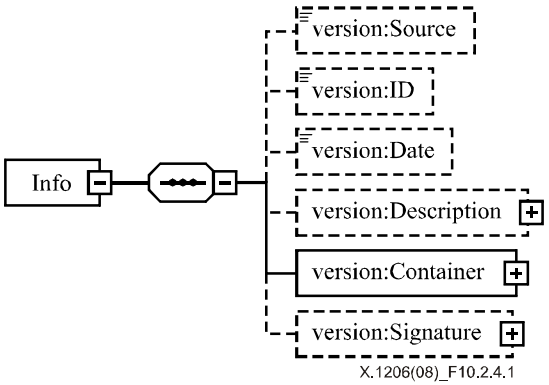
Info – Структура данных, используемая для переноса информации об уязвимости для продуктов или ресурсов либо их фактической реализации, как указано в предыдущих значениях полей. [ФАКУЛЬТАТИВНО]

Signature – Упакованная подпись XML. [ФАКУЛЬТАТИВНО]

ПРИМЕЧАНИЕ. – Информация, содержащаяся в Info, всегда подлежит использованию для идентификации данного ресурса с *Product_Name* и *Product_Version*, используемыми только ради удобства; однако если идентифицированный в Info ресурс является продуктом высокого уровня, *Product_Name* и *Product_Version*, как предполагается, подлежат использованию для четкого указания того, что для идентифицированного ресурса не существует никакого "родительского" продукта.

10.2.4 Info

10.2.4.1 Синтаксис



10.2.4.2 Семантика

Source – Текстовое описание, например название, производителя или поставщика информации об уязвимости, предусматривающее включенный код или информацию об уязвимости. [ФАКУЛЬТАТИВНО]

ID – Идентификатор, которым можно идентифицировать данный набор данных *Info*. Хотя метод присвоения не указан, предполагается, что присвоенные Идентификаторы должны быть только однозначными в рамках серии данного ресурса из данного *Источника* (*Source*). [ФАКУЛЬТАТИВНО]

Date – Обозначение даты XML относительно данных *Info*. [ФАКУЛЬТАТИВНО]

Description – Текстовое или HTML описание либо кода, содержащегося в факультативном варианте *Info_Container*, либо ссылки на его унифицированный идентификатор ресурса (URI) для конкретной версии ресурса, описанной в конкретном примере этой структуры данных. [ФАКУЛЬТАТИВНО]

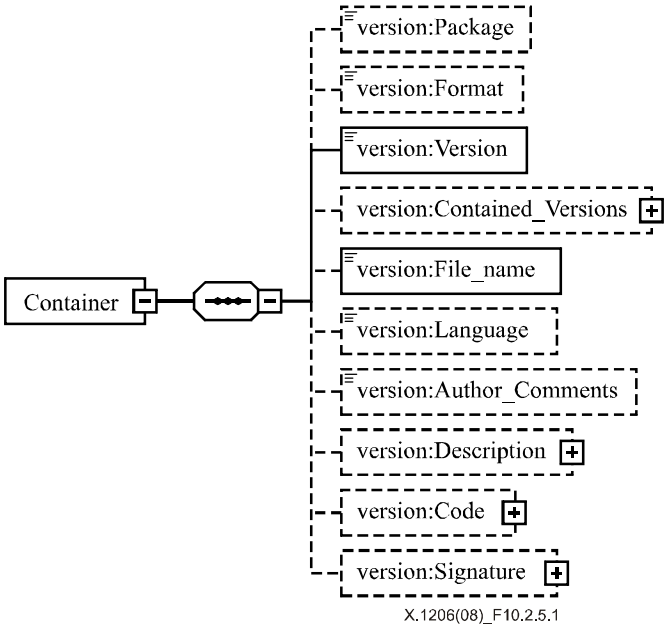
Container – Структура данных для переноса кода в форме обновления или корректировки. [ОБЯЗАТЕЛЬНО]

Signature – Упакованная подпись XML. [ФАКУЛЬТАТИВНО]

10.2.5 Container

Контейнер для информации об идентификации версии, а также информации об уязвимости, обновления или корректировки.

10.2.5.1 Синтаксис



10.2.5.2 Семантика

Package – Метод упаковки, используемой конкретной версией, и/или содержание *Кода* (*Code*). [ФАКУЛЬТАТИВНО]

Format – Формат конкретной версии. [ФАКУЛЬТАТИВНО]

Version – Версия, в соответствующих случаях либо включенная для запросов, либо используемая для идентификации содержания *Version_Control* и/или *Code*. [ФАКУЛЬТАТИВНО]

Contained_Version – Список, описанный с использованием структуры данных *Version_Info (см. выше)*, который содержит описание всех компонентов действующего ресурса или компонента. [ФАКУЛЬТАТИВНО]

File_Name – Название файла. [ФАКУЛЬТАТИВНО]

Language – В отношении ресурсов, для которых в разных странах имеются различные версии, в зависимости от языка, это поле используется для идентификации языка. [ФАКУЛЬТАТИВНО]

Author_Comments – Любое текстовое содержание, по желанию [ФАКУЛЬТАТИВНО]

Description – Текстовое или HTML описание конкретной версии, содержащееся в *Коде* (*Code*), или текстовое или HTML описание информации об уязвимости, относящейся к ресурсу,

идентифицированному с помощью *Version* и/или *File_Name*, используемого с помощью или в рамках *Version_Info: Product_Name* и *Version_Info: Product_Version*.

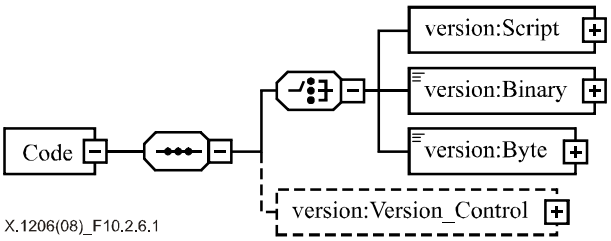
Code – Контейнер для выполняемого кода, а также информация об управлении версией.

Signature – Упакованная подпись XML. [ФАКУЛЬТАТИВНО]

10.2.6 Code

Контейнер для фактического кода или цепочек битов, например, как вывод памяти, или ссылки на них URL, а также информация, полезная для контролирующих версию корректировок.

10.2.6.1 Синтаксис



10.2.6.2 Семантика

Script – Код в зашифрованном виде. [ФАКУЛЬТАТИВНО ОБЯЗАТЕЛЬНО]

Binary – Код в скомпилированной двоичной форме. [ФАКУЛЬТАТИВНО ОБЯЗАТЕЛЬНО]

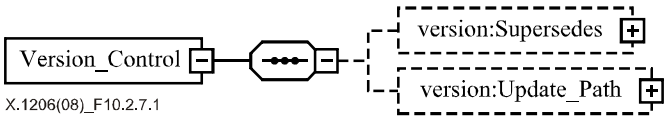
Byte – данные, представленные для анализа. [ФАКУЛЬТАТИВНО ОБЯЗАТЕЛЬНО]

Version_Control – Структура данных, используемая для указания связей между последовательными версиями и возможными маршрутами их обновления. [ФАКУЛЬТАТИВНО]

10.2.7 Version_Control

Контейнер для переноса списков последовательного ряда версий, заменяемых данной версией, а также списка, идентифицирующего, к каким версиям могут применяться идентифицированное обновление или корректировка.

10.2.7.1 Синтаксис



10.2.7.2 Семантика

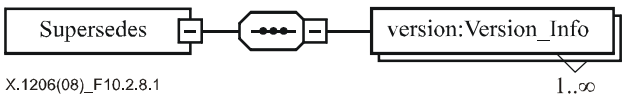
Supersedes – Последовательный перечень версий, заменяемых *Version_Info*. [ФАКУЛЬТАТИВНО]

Update_Path – Список версий, к которым может применяться текущее обновление или корректировка. [ФАКУЛЬТАТИВНО]

10.2.8 Supersedes

Список ресурсов, которые заменяются указанной версией.

10.2.8.1 Синтаксис



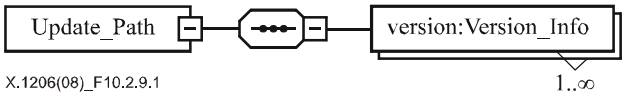
10.2.8.2 Семантика

Version_Info – (См. выше).

10.2.9 Update_Path

Список ресурсов, к которым может применяться содержащееся обновление или корректировка.

10.2.9.1 Синтаксис

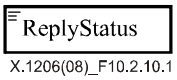


10.2.9.2 Семантика

Version_Info – (См. выше).

10.2.10 ReplyStatus

10.2.10.1 Синтаксис



10.2.10.2 Семантика

ReplyStatus – Содержит одно из следующих:

"Failed" – Запрашиваемая операция не могла быть завершена.

"Succeeded" – Запрашиваемая операция выполнена успешно.

"Major message Not Supported" – Запрашиваемый идентификатор основного сообщения или его использование в данном контексте не поддерживается.

"Minor message Not Supported" – Запрашиваемый идентификатор вспомогательного сообщения или его использование в данном контексте не поддерживается.

"Handler not available" – Тип идентифицированного сообщения не поддерживается.

11 Схемы

11.1 Message_Core

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:messagecore="http://www.itu.int//xml-namespace/itu-t/x.1206//CORE" xmlns:ns1="http://www.w3.org/2000/09/xmldsig#" targetNamespace="http://www.itu.int//xml-namespace/itu-t/x.1206//CORE" elementFormDefault="qualified" attributeFormDefault="unqualified">
    <xs:import namespace="http://www.w3.org/2000/09/xmldsig#" schemaLocation="xmldsig-core-schema.xsd"/>
    <xs:element name="Message_Core">
        <xs:annotation>
            <xs:documentation>Vulnerability Information, Update and Patch Notification Architecture</xs:documentation>
        </xs:annotation>
        <xs:complexType>
            <xs:sequence>
                <xs:element name="Header" type="messagecore:HeaderType"/>
                <xs:element name="Payload">
                    <xs:complexType>
                        <xs:choice>
                            <xs:element ref="messagecore:Message_Core" maxOccurs="unbounded"/>
                            <xs:element ref="messagecore:AbstractMessage" maxOccurs="unbounded"/>
                        </xs:choice>
```

```
                    </xs:complexType>
                </xs:element>
            </xs:sequence>
        </xs:complexType>
    </xs:element>
    <xs:complexType name="HeaderType">
        <xs:sequence>
            <xs:element name="PID" type="xs:string"/>
            <xs:element name="Server_GUID" type="xs:string" minOccurs="0"/>
            <xs:element name="Client_GUID" type="xs:string" minOccurs="0"/>
            <xs:element name="Message_ID" type="xs:string" minOccurs="0"/>
            <xs:element name="Sent_Time" type="xs:dateTime" minOccurs="0"/>
            <xs:element name="Signature" type="ns1:SignatureType" minOccurs="0"/>
        </xs:sequence>
    </xs:complexType>
    <xs:element name="AbstractMessage" type="messagecore:AbstractMessageType" abstract="true"/>
    <xs:complexType name="AbstractMessageType">
        <xs:sequence>
            <xs:element ref="messagecore:AbstractCID"/>
            <xs:element ref="messagecore:AbstractUUID" minOccurs="0"/>
            <xs:element ref="messagecore:AbstractMajor_Message"/>
            <xs:element ref="messagecore:AbstractMinor_Message" minOccurs="0"/>
            <xs:element ref="messagecore:AbstractMessage_Data" minOccurs="0"/>
            <xs:element name="Signature" type="ns1:SignatureType" minOccurs="0"/>
        </xs:sequence>
    </xs:complexType>
    <xs:element name="AbstractCID" type="messagecore:AbstractCIDType" abstract="true"/>
    <xs:complexType name="AbstractCIDType">
        <xs:simpleContent>
            <xs:extension base="xs:string"/>
        </xs:simpleContent>
    </xs:complexType>
    <xs:element name="AbstractUUID" type="messagecore:AbstractUUIDType" abstract="true"/>
    <xs:complexType name="AbstractUUIDType">
        <xs:simpleContent>
            <xs:extension base="xs:string"/>
        </xs:simpleContent>
    </xs:complexType>
    <xs:element name="AbstractMajor_Message" type="messagecore:AbstractMajorMessageType" abstract="true"/>
    <xs:complexType name="AbstractMajorMessageType">
        <xs:simpleContent>
            <xs:extension base="xs:string"/>
        </xs:simpleContent>
    </xs:complexType>
    <xs:element name="AbstractMinor_Message" type="messagecore:AbstractMinorMessageType" abstract="true"/>
    <xs:complexType name="AbstractMinorMessageType">
        <xs:simpleContent>
            <xs:extension base="xs:string"/>
```

```
        </xs:simpleContent>
    </xs:complexType>
    <xs:element name="AbstractMessage_Data" type="messagecore:AbstractMessage_DataType" abstract="true"/>
    <xs:complexType name="AbstractMessage_DataType"/>
    <xs:element name="Message_Core_Reply">
        <xs:complexType>
            <xs:sequence>
                <xs:element name="Header" type="messagecore:HeaderType"/>
                <xs:element name="Payload">
                    <xs:complexType>
                        <xs:choice>
                            <xs:element ref="messagecore:Message_Core_Reply" maxOccurs="unbounded"/>
                            <xs:element ref="messagecore:AbstractReply"/>
                        </xs:choice>
                    </xs:complexType>
                </xs:element>
            </xs:sequence>
        </xs:complexType>
    </xs:element>
    <xs:element name="AbstractReply" type="messagecore:AbstractReplyType"/>
    <xs:complexType name="AbstractReplyType">
        <xs:sequence>
            <xs:element name="CID" type="messagecore:AbstractCIDType"/>
            <xs:element name="UUID" type="messagecore:AbstractUUIDType" minOccurs="0"/>
            <xs:element name="Reply_ID" type="xs:string" minOccurs="0"/>
            <xs:element ref="messagecore:AbstractReplyStatus"/>
            <xs:element name="Reply_Description" type="messagecore:Description_Type" minOccurs="0"/>
            <xs:element ref="messagecore:AbstractReply_Data" minOccurs="0"/>
            <xs:element name="Signature" type="ns1:SignatureType" minOccurs="0"/>
        </xs:sequence>
    </xs:complexType>
    <xs:element name="AbstractReplyStatus" type="messagecore:AbstractReplyStatusType" abstract="true"/>
    <xs:complexType name="AbstractReplyStatusType">
        <xs:simpleContent>
            <xs:extension base="xs:string"/>
        </xs:simpleContent>
    </xs:complexType>
    <xs:element name="AbstractReply_Data" type="messagecore:AbstractReply_DataType" abstract="true"/>
    <xs:complexType name="AbstractReply_DataType"/>
    <xs:complexType name="Description_Type">
        <xs:attribute name="ref" type="xs:anyURI" use="optional"/><![CDATA[]]></xs:complexType>
</xs:schema>
```

11.2 Message_Version

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:messagecore="http://www.itu.int//xml-namespace/itu-t/x.1206///CORE"
xmlns:version="http://www.itu.int//xml-namespace/itu-t/x.1206///CORE/MESSAGE/VERSION/"
xmlns:xmldsig="http://www.w3.org/2000/09/xmldsig#" targetNamespace="http://www.itu.int//xml-namespace/itu-
t/x.1206///CORE/MESSAGE/VERSION/" elementFormDefault="qualified" attributeFormDefault="unqualified">

    <xs:import namespace="http://www.example.com/CORE" schemaLocation="Message_Core.xsd"/>
```

```
<xs:import namespace="http://www.w3.org/2000/09/xmldsig#" schemaLocation="xmldsig-core-schema.xsd"/>
<xs:element name="Message" type="messagecore:AbstractMessageType" substitutionGroup="messagecore:AbstractMessage">
    <xs:annotation>
        <xs:documentation>Vulnerability Information, Update and Patch Notification Message : Version</xs:documentation>
    </xs:annotation>
</xs:element>
<xs:element name="CID" type="messagecore:AbstractCIDType" substitutionGroup="messagecore:AbstractCID"/>
<xs:element name="UUID" type="messagecore:AbstractUUIDType" substitutionGroup="messagecore:AbstractUUID"/>
<xs:element name="Major_Message" type="version:Major_MessageType" substitutionGroup="messagecore:AbstractMajor_Message"/>
<xs:complexType name="Major_MessageType">
    <xs:simpleContent>
        <xs:restriction base="messagecore:AbstractMajorMessageType">
            <xs:enumeration value="Register"/>
            <xs:enumeration value="Request"/>
            <xs:enumeration value="Deliver"/>
            <xs:enumeration value="Analyse"/>
        </xs:restriction>
    </xs:simpleContent>
</xs:complexType>
<xs:element name="Minor_Message" type="version:Minor_MessageType" substitutionGroup="messagecore:AbstractMinor_Message"/>
<xs:complexType name="Minor_MessageType">
    <xs:simpleContent>
        <xs:restriction base="messagecore:AbstractMinorMessageType">
            <xs:enumeration value="Full"/>
            <xs:enumeration value="Catalog"/>
            <xs:enumeration value="End"/>
        </xs:restriction>
    </xs:simpleContent>
</xs:complexType>
<xs:element name="Message_Data" type="version:Message_DataType" substitutionGroup="messagecore:AbstractMessage_Data"/>
<xs:complexType name="Message_DataType">
    <xs:complexContent>
        <xs:extension base="messagecore:AbstractMessage_DataType">
            <xs:sequence>
                <xs:element name="Version_Info" type="version:Version_InfoType" minOccurs="0" maxOccurs="unbounded"/>
            </xs:sequence>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<xs:element name="Reply" type="messagecore:AbstractReplyType" substitutionGroup="messagecore:AbstractReply"/>
<xs:element name="ReplyStatus" substitutionGroup="messagecore:AbstractReplyStatus">
    <xs:complexType>
        <xs:simpleContent>
            <xs:restriction base="messagecore:AbstractReplyStatusType">
                <xs:enumeration value="Failed"/>
                <xs:enumeration value="Succeeded"/>
                <xs:enumeration value="Major Message Not Supported"/>
                <xs:enumeration value="Minor Message Not Supported"/>
```

```
                <xs:enumeration value="Handler not available"/>
            </xs:restriction>
        </xs:simpleContent>
    </xs:complexType>
</xs:element>
<xs:element name="Reply_Data" type="messagecore:AbstractReply_DataType" substitutionGroup="messagecore:AbstractReply_Data"/>
<xs:complexType name="Version_InfoType">
    <xs:complexContent>
        <xs:extension base="messagecore:AbstractReply_DataType">
            <xs:sequence>
                <xs:element name="Product_Name" type="xs:string" minOccurs="0"/>
                <xs:element name="Product_Version" type="xs:string" minOccurs="0"/>
                <xs:element name="Company" type="xs:string" minOccurs="0"/>
                <xs:element name="Info" minOccurs="0" maxOccurs="unbounded">
                    <xs:complexType>
                        <xs:sequence>
                            <xs:element name="Source" type="xs:string" minOccurs="0"/>
                            <xs:element name="ID" type="xs:string" minOccurs="0"/>
                            <xs:element name="Date" type="xs:string" minOccurs="0"/>
                            <xs:element name="Description" minOccurs="0">
                                <xs:complexType>
                                    <xs:attribute name="ref" type="xs:anyURI" use="optional"/><![CDATA[]]></xs:complexType>
                            </xs:element>
                            <xs:element name="Container">
                                <xs:complexType>
                                    <xs:sequence>
                                        <xs:element name="Package" type="xs:string" minOccurs="0"/>
                                        <xs:element name="Format" type="xs:string" minOccurs="0"/>
                                        <xs:element name="Version" type="xs:string"/>
                                        <xs:element name="Contained_Versions" type="version:Version_InfoType" minOccurs="0"/>
                                        <xs:element name="File_name" type="xs:string" minOccurs="0"/>
                                        <xs:element name="Language" type="xs:string" minOccurs="0"/>
                                        <xs:element name="Author_Comments" type="xs:string" minOccurs="0"/>
                                        <xs:element name="Description" type="messagecore:Description_Type" minOccurs="0"/>
                                        <xs:element name="Code" minOccurs="0">
                                            <xs:complexType>
                                                <xs:sequence>
                                                    <xs:choice>
                                                        <xs:element name="Script">
                                                            <xs:complexType>
                                                                <xs:attribute name="ref" type="xs:anyURI" use="optional"/>
                                                            </xs:complexType>
                                                        </xs:element>
                                                        <xs:element name="Binary">
                                                            <xs:complexType>
                                                                <xs:simpleContent>
                                                                    <xs:extension base="xs:base64Binary">
                                                                        <xs:attribute name="ref" type="xs:anyURI" use="optional"/>
```

```
                                    </xs:extension>
                                </xs:simpleContent>
                            </xs:complexType>
                        </xs:element>
                        <xs:element name="Byte">
                            <xs:complexType>
                                <xs:simpleContent>
                                    <xs:extension base="xs:base64Binary">
                                        <xs:attribute name="ref" type="xs:anyURI" use="optional"/>
                                    </xs:extension>
                                </xs:simpleContent>
                            </xs:complexType>
                        </xs:element>
                    </xs:choice>
                    <xs:element name="Version_Control" minOccurs="0">
                        <xs:complexType>
                            <xs:sequence>
                                <xs:element name="Supersedes" minOccurs="0">
                                    <xs:complexType>
                                        <xs:sequence>
                                            <xs:element name="Version_info"
type="version:Version_InfoType" maxOccurs="unbounded"/>
                                        </xs:sequence>
                                    </xs:complexType>
                                </xs:element>
                                <xs:element name="Update_Path" minOccurs="0">
                                    <xs:complexType>
                                        <xs:sequence>
                                            <xs:element name="Version_Info"
type="version:Version_InfoType" maxOccurs="unbounded"/>
                                        </xs:sequence>
                                    </xs:complexType>
                                </xs:element>
                            </xs:sequence>
                        </xs:complexType>
                    </xs:element>
                </xs:sequence>
            </xs:complexType>
        </xs:element>
        <xs:element name="Signature" type="xmldsig:SignatureType" minOccurs="0"/>
    </xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="Signature" type="xmldsig:SignatureType" minOccurs="0"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="Signature" type="xmldsig:SignatureType" minOccurs="0"/>
</xs:sequence>
```

```
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>
</xs:schema>
```

Библиография

[b-ITU-T X.667] Recommendation ITU-T X.667 (2004) | ISO/IEC 9834-8:2005, *Information technology – Open Systems Interconnection – Procedures for the operation of OSI Registration Authorities: Generation and registration of Universally Unique Identifiers (UUIDs) and their use as ASN.1 object identifier components*.

[b-IETF RFC 3023] IETF RFC 3023 (2001), *XML Media Types*. <http://www.ietf.org/rfc/rfc3023.txt?number=3023>

[b-IETF RFC 3075] IETF RFC 3075 (2001), *XML-Signature Syntax and Processing*. <http://www.ietf.org/rfc/rfc3075.txt?number=3075>

[b-XML Datatypes] W3C Datatypes:2001, *XML Schema Part 2: Datatypes*, W3C Recommendation, Copyright © [2 May 2001] World Wide Web Consortium, (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University), http://www.w3.org/TR/2001/RECxmlschema-2-20010502/.

[b-XML Signature] W3C Signature Schema:2001, *XML Signature Schema*, W3C Recommendation, Copyright © [1 March 2001] World Wide Web Consortium, (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University), http://www.w3.org/TR/xmldsigcore/xmldsig-core-schema.xsd.

[b-XML Structures] W3C XML Schema Part 1:2001, *XML Schema Part 1: Structures*, W3C Recommendation, Copyright © [2 May 2001] World Wide Web Consortium, (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University), http://www.w3.org/TR/2001/RECxmlschema-1-20010502/.

СЕРИИ РЕКОМЕНДАЦИЙ МСЭ-Т

Серия A	Организация работы МСЭ-Т
Серия D	Общие принципы тарификации
Серия E	Общая эксплуатация сети, телефонная служба, функционирование служб и человеческие факторы
Серия F	Нетелефонные службы электросвязи
Серия G	Системы и среда передачи, цифровые системы и сети
Серия H	Аудиовизуальные и мультимедийные системы
Серия I	Цифровая сеть с интеграцией служб
Серия J	Кабельные сети и передача сигналов телевизионных и звуковых программ и других мультимедийных сигналов
Серия K	Защита от помех
Серия L	Конструкция, прокладка и защита кабелей и других элементов линейно-кабельных сооружений
Серия M	Управление электросвязью, включая СУЭ и техническое обслуживание сетей
Серия N	Техническое обслуживание: международные каналы передачи звуковых и телевизионных программ
Серия O	Требования к измерительной аппаратуре
Серия P	Качество телефонной передачи, телефонные установки, сети местных линий
Серия Q	Коммутация и сигнализация
Серия R	Телеграфная передача
Серия S	Оконечное оборудование для телеграфных служб
Серия T	Оконечное оборудование для телематических служб
Серия U	Телеграфная коммутация
Серия V	Передача данных по телефонной сети
Серия X	**Сети передачи данных, взаимосвязь открытых систем и безопасность**
Серия Y	Глобальная информационная инфраструктура, аспекты протокола Интернет и сети последующих поколений
Серия Z	Языки и общие аспекты программного обеспечения для систем электросвязи

Отпечатано в Швейцарии
Женева, 2009 г.