



МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ, СВЯЗИ И МАССОВЫХ
КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ

МИНИСТЕРСТВО ЮСТИЦИИ РОССИЙСКОЙ ФЕДЕРАЦИИ

ЗАРЕГИСТРИРОВАНО

Регистрационный № 53207

от "27" декабря 2018.

ПРИКАЗ

29.10.2018

№ 571

Москва

**Об утверждении Требований к оборудованию
и программно-техническим средствам, используемым организатором
распространения информации в сети «Интернет» в эксплуатируемых им
информационных системах, для проведения уполномоченными
государственными органами, осуществляющими оперативно-розыскную
деятельность или обеспечение безопасности Российской Федерации,
мероприятий в целях реализации возложенных на них задач**

В соответствии с частью 4 статьи 10.1 Федерального закона от 27 июля 2006 г. № 149-ФЗ «Об информации, информационных технологиях и о защите информации» (Собрание законодательства Российской Федерации, 2006, № 31, ст. 3448; 2014, № 19, ст. 2302; 2016, № 28, ст. 4558; 2017, № 31, ст. 4790; 2018, № 30, ст. 4546), пунктом 11.1 Правил хранения организаторами распространения информации в информационно-телекоммуникационной сети «Интернет» информации о фактах приема, передачи, доставки и (или) обработки голосовой информации, письменного текста, изображений, звуков или иных электронных сообщений пользователей информационно-телекоммуникационной сети «Интернет» и информации об этих пользователях, предоставления ее уполномоченным государственным органам, осуществляющим оперативно-розыскную деятельность или обеспечение безопасности Российской Федерации, утвержденных постановлением Правительства Российской Федерации от 31 июля 2014 г. № 759 (Собрание законодательства Российской Федерации, 2014, № 32, ст. 4526; 2017, № 52, ст. 8153), пунктами 2 и 3 Правил хранения организатором распространения информации в информационно-телекоммуникационной сети «Интернет» текстовых сообщений пользователей информационно-телекоммуникационной сети «Интернет», голосовой информации, изображений, звуков, видео-, иных электронных сообщений пользователей информационно-телекоммуникационной сети «Интернет», утвержденных постановлением Правительства Российской Федерации от 26 июня 2018 г. № 728 (Собрание законодательства Российской Федерации, 2018, № 27, ст. 4081), и пунктами 5 и 8 Правил взаимодействия

организаторов распространения информации в информационно-телекоммуникационной сети «Интернет» с уполномоченными государственными органами, осуществляющими оперативно-разыскную деятельность или обеспечение безопасности Российской Федерации, утвержденных постановлением Правительства Российской Федерации от 31 июля 2014 г. № 743 (Собрание законодательства Российской Федерации, 2014, № 32, ст. 4516; 2018, № 4, ст. 638),

ПРИКАЗЫВАЮ:

1. Утвердить прилагаемые Требования к оборудованию и программно-техническим средствам, используемым организатором распространения информации в сети «Интернет» в эксплуатируемых им информационных системах, для проведения уполномоченными государственными органами, осуществляющими оперативно-разыскную деятельность или обеспечение безопасности Российской Федерации, мероприятий в целях реализации возложенных на них задач.

2. Настоящий приказ вступает в силу по истечении ста восьмидесяти дней после дня его официального опубликования.

3. Направить настоящий приказ на государственную регистрацию в Министерство юстиции Российской Федерации.

Министр



К.Ю. Носков

УТВЕРЖДЕНЫ

приказом Министерства цифрового
развития, связи и массовых коммуникаций
Российской Федерации

от 29.10.2018 2018 № 571

Требования

**к оборудованию и программно-техническим средствам,
используемым организатором распространения информации
в сети «Интернет» в эксплуатируемых им информационных системах,
для проведения уполномоченными государственными органами,
осуществляющими оперативно-розыскную деятельность или обеспечение
безопасности Российской Федерации, мероприятий
в целях реализации возложенных на них задач**

I. Общие положения

1. Настоящим актом устанавливаются требования к оборудованию и программно-техническим средствам, используемым организатором распространения информации в сети «Интернет» (далее соответственно – ПТС ОРИ, ОРИ) в эксплуатируемых им информационных системах (далее – ИС ОРИ), для проведения уполномоченными государственными органами, осуществляющими оперативно-розыскную деятельность или обеспечение безопасности Российской Федерации, мероприятий в целях реализации возложенных на них задач (далее – требования).

II. Требования к ПТС ОРИ

2. Состав и построение ПТС ОРИ, используемых для обеспечения проведения оперативно-розыскных мероприятий (далее – ОРМ), определяются особенностями построения ИС ОРИ, перечнем информационных услуг и коммуникационных интернет-сервисов.

3. При реализации ПТС ОРИ должен быть использован один из вариантов их построения:

- а) отдельный аппаратно-программный комплекс;
- б) отдельный программный модуль в ИС ОРИ;
- в) комбинированный вариант, предусматривающий совместное использование элементов в соответствии с подпунктами «а» и «б» настоящего пункта.

Для каждого варианта построения должна обеспечиваться защита информации от несанкционированного доступа, связанной с проведением ОРМ, в соответствии с подпунктами «з» и «и» пункта 14 настоящих требований.

III. Функциональные требования к ПТС ОРИ

4. ПТС ОРИ должны обеспечивать поиск, обработку и передачу на пункт управления уполномоченного подразделения органа федеральной службы безопасности (далее – ПУ) по его запросу или в автоматическом режиме следующих хранимых в ИС ОРИ и (или) ПТС ОРИ при реализации ПТС ОРИ в соответствии с пунктом 2 настоящих требований данных:

1) информации о зарегистрированных пользователях в соответствии с пунктами 12 и 14 Правил хранения организаторами распространения информации в информационно-телекоммуникационной сети «Интернет» информации о фактах приема, передачи, доставки и (или) обработки голосовой информации, письменного текста, изображений, звуков или иных электронных сообщений пользователей информационно-телекоммуникационной сети «Интернет» и информации об этих пользователях, предоставления ее уполномоченным государственным органам, осуществляющим оперативно-разыскную деятельность или обеспечение безопасности Российской Федерации, утвержденные постановлением Правительства Российской Федерации от 31 июля 2014 г. № 759 (далее – Правила хранения № 759);

2) информации о фактах приема, передачи, доставки и (или) обработки голосовой информации, письменного текста, изображений, звуков, видео- или иных электронных сообщений пользователей информационно-телекоммуникационной сети «Интернет» (далее – пользователь) и информации об этих пользователях в соответствии с пунктами 3, 13 и 14 Правил хранения № 759;

3) текстовых сообщений пользователей информационно-телекоммуникационной сети «Интернет», голосовой информации, изображений, звуков, видео-, иных электронных сообщений пользователей в соответствии с Правилами хранения организатором распространения информации в информационно-телекоммуникационной сети «Интернет» текстовых сообщений пользователей информационно-телекоммуникационной сети «Интернет», голосовой информации, изображений, звуков, видео-, иных электронных сообщений пользователей информационно-телекоммуникационной сети «Интернет», утвержденных постановлением Правительства Российской Федерации от 26 июня 2018 г. № 728;

4) информации, необходимой для декодирования принимаемых, передаваемых, доставляемых и (или) обрабатываемых электронных сообщений;

5) информации о событиях, зафиксированных в системных журналах в соответствии с подпунктом «т» пункта 13 настоящих требований.

5. ПТС ОРИ с ПУ должно взаимодействовать с ПУ по четырем каналам передачи данных (далее – кпд):

кпд1 – канал управления;

кпд2 – канал данных;

кпд3 – канал мониторинга;

кпд4 – канал неформатированных данных.

Суммарная пропускная способность каналов передачи данных между ПТС ОРИ и ПУ, указанных в настоящем пункте, должна соответствовать данным, приведенным в таблице № 1.

Таблица № 1.

№№	Количество зарегистрированных пользователей, тысяч	Среднесуточный объем данных, доставленных пользователям, Гбайт	Суммарная пропускная способность каналов передачи данных между ПТС ОРИ и ПУ, не менее Мбит/с
1	< 1	< 1	10
2	1 – 10	1 – 10	50
3	10 – 100	10 – 100	100
4	100 – 500	100 – 500	500
5	500 – 1000	500 – 1000	800
6	1000 – 10000	1000 – 10000	1000
7	> 10000	> 10000	10000

6. Требования к ПТС ОРИ в части обработки информации приведены в приложении № 1 к настоящим требованиям.

7. Требования, предъявляемые к интерфейсу взаимодействия между ПУ и ПТС ОРИ, приведены в приложении № 2 к настоящим требованиям.

8. Требования к форматам сообщений ПТС ОРИ приведены в приложении № 3 к настоящим требованиям.

9. Описание протокола взаимодействия ПУ и ПТС ОРИ приведено в приложении № 4 к настоящим требованиям.

10. Требования, предъявляемые к функционированию канала управления кпд1, приведены в приложении № 5 к настоящим требованиям.

11. Требования, предъявляемые к функционированию канала данных кпд2, приведены в приложении № 6 к настоящим требованиям.

12. Требования, предъявляемые к функционированию канала мониторинга кпд3, приведены в приложении № 7 к настоящим требованиям.

13. Требования, предъявляемые к функционированию канала неформатированных данных кпд4, приведены в приложении № 8 к настоящим требованиям.

14. ПТС ОРИ должны обеспечивать:

а) подключение к ПУ в соответствии с пунктами 5 и 8 Правил взаимодействия организаторов распространения информации в информационно-телекоммуникационной сети «Интернет» с уполномоченными государственными органами, осуществляющими оперативно-разыскную деятельность или обеспечение безопасности Российской Федерации, утвержденных постановлением Правительства Российской Федерации от 31 июля 2014 г. № 743;

б) отбор и передачу на ПУ поступающей в ИС ОРИ информации по количеству параметров контроля, указанному в таблице № 2 настоящих требований, в соответствии с временными характеристиками, указанными в пункте 17 настоящих требований;

Таблица № 2.

№№	Количество зарегистрированных пользователей, тысяч	Количество параметров контроля, не менее
1	< 1	100
2	1 – 10	2000
3	10 – 100	5000
4	100 – 500	7000
5	500 – 1000	10000
6	1000 – 10000	20000
7	> 10000	50000

в) передачу на ПУ по запросу с ПУ получаемых из ИС ОРИ данных в соответствии пунктом 3 настоящих требований без дополнительной обработки;

г) прием от ПУ критериев поиска и передачу на ПУ статистической, текстовой, мультимедийной, звуковой, графической и иной информации в исходном виде, отбираемой из ИС ОРИ по критериям поиска, без дополнительной обработки;

д) контроль времени поступления передаваемой ПТС ОРИ на ПУ информации и информирование ПУ о превышении временных параметров, предусмотренных разделом IV настоящих требований;

е) круглосуточный удаленный доступ ПУ к ПТС ОРИ для получения информации в соответствии с пунктом 3 настоящих требований;

ж) возможность управления ПТС ОРИ с ПУ;

з) защиту от несанкционированного доступа к информации, связанной с проведением ОРМ с использованием ПТС ОРИ

и) информирование ПУ о следующих попытках несанкционированного доступа:

1) для варианта исполнения ПТС ОРИ в виде отдельного аппаратно-программного комплекса:

доступ к данным, связанным с проведением ОРМ с использованием команд или сервисных программ;

резервное копирование данных, связанных с проведением ОРМ;

доступ к ПТС ОРИ через интерфейсы, не предусмотренные для доступа к ним;

вскрытие корпуса ПТС ОРИ;

2) для варианта исполнения ПТС ОРИ в виде отдельного программного модуля в ИС ОРИ:

доступ к данным, связанным с проведением ОРМ с использованием команд или сервисных программ;

резервное копирование данных, связанных с проведением ОРМ;

3) для комбинированного варианта исполнения ПТС ОРИ:

доступ к данным, связанным с проведением ОРМ с использованием команд или сервисных программ;

резервное копирование данных, связанных с проведением ОРМ;

доступ к ПТС ОРИ через интерфейсы, не предусмотренные для доступа к ним;

вскрытие корпуса ПТС ОРИ;

к) доступ уполномоченного технического персонала для обслуживания ПТС ОРИ в соответствии с установленными правами доступа;

л) реализацию протокола взаимодействия оборудования ПТС ОРИ с оборудованием ПУ в соответствии с приложением № 4 к настоящим требованиям;

м) прием от ПУ запросов в соответствии с протоколом взаимодействия, приведенным в приложении № 4 к настоящим требованиям;

н) передачу на ПУ из ПТС ОРИ отобранных данных, соответствующих поступившим с ПУ запросам в соответствии с протоколом взаимодействия, приведенным в приложении № 4 к настоящим требованиям;

о) прием от ПУ поисковых критериев и параметров контроля в соответствии с протоколом взаимодействия, приведенным в приложении № 4 к настоящим требованиям;

п) доступ к автоматически сохраненным в ИС ОРИ, подвергшимся модификации данным пользователей в виде отдельных идентифицируемых образцов и времени сохранения в ИС ОРИ, внесенных изменений в случае возможности сохранения в ИС ОРИ промежуточных модификаций данных;

р) доступ к копии информационного содержимого сообщения, хранящейся в ИС ОРИ. При этом допускается хранение одной копии информационного содержимого сообщения вне зависимости от количества операций по его доставке авторизованным и гостевым пользователям ИС ОРИ;

с) выполнение регистрации фактически выполненных пользователем действий по выбору получаемого фрагмента сообщения, прекращение его передачи. При этом допускается хранение одной копии информационного содержимого сообщения вне зависимости от количества операций по его доставке авторизованным и гостевым пользователям ИС ОРИ;

т) ведение системных журналов, содержащих информацию о работе ПТС ОРИ, а также связанных с проведением ОРМ данных:

1) о сеансах связи с ПУ, а также о попытках установления таких сеансов;

2) о запросах ПУ на получение данных;

3) об ответах на запросы ПУ на получение данных;

4) о текущей конфигурации ПТС ОРИ;

5) об изменениях в конфигурации ПТС ОРИ, системного и прикладного ПО;

6) о сбоях в ПТС ОРИ, системном и прикладном ПО;

7) об обращении и доступе обслуживающего технического персонала к ПТС ОРИ;

у) доступ уполномоченного технического персонала для выполнения регламентных и ремонтных работ к системным журналам ПТС ОРИ в соответствии с правами доступа, установленными системой разграничения и контроля доступа с обязательной регистрацией команд и сообщений, используемых техническим персоналом при обращении к ПТС ОРИ;

ф) сохранность и доступность для дальнейшего использования ранее накопленных данных при модернизации аппаратных и программных средств ПТС ОРИ.

15. ПТС ОРИ не должны оказывать влияния на работоспособность ИС ОРИ, при этом ПТС ОРИ не должны иметь иных интерфейсов взаимодействия кроме интерфейсов взаимодействия с ИС ОРИ и ПУ.

16. Предельная временная задержка с момента поступления с ПУ на ПТС ОРИ критерия отбора до начала доставки на ПУ отобранной информации должна составлять не более 5 минут.

17. ПТС ОРИ должны обеспечивать выполнение заданных с ПУ поисковых задач для ИС ОРИ, при этом запрос должен выполняться в соответствии с временными ограничениями, приведенными в главе IV настоящих требований.

IV. Требования, предъявляемые к ПТС ОРИ в части временных параметров обработки информации

18. Продолжительность предварительной обработки информации с момента ее появления в ИС ОРИ до момента, когда эта информация становится доступной ПТС ОРИ для выполнения запросов с ПУ, не должна превышать следующих значений:

1) 60 секунд – для внесенных пользовательских данных после завершения процедуры регистрации, изменения регистрационных данных, иных дополнений, внесенных пользователями коммуникационного интернет-сервиса;

2) 5 минут – для данных о фактах авторизаций и выхода из коммуникационного интернет-сервиса, регистрации, прекращения регистрации пользователей;

3) 5 минут с момента наступления события – для следующих данных:

данные о фактах приема, передачи и (или) обработки голосовой информации, текстовых сообщений, изображений, звуков, видео- или иных электронных сообщений пользователей сети «Интернет» (информация о точном времени приема, передачи, доставки и (или) обработки электронных сообщений с указанием информации об адресатах этих сообщений, в том числе информации для неопределенного круга пользователей сети «Интернет»);

фиксируемая коммуникационным интернет-сервисом информация об организации приема, передачи, доставки и (или) обработки электронных сообщений, осуществляемых с использованием технологий электронных платежных систем, в том числе информация о произведенных денежных операциях (с указанием информации о корреспонденте - идентификаторе платежной системы, валюты, суммы, оплачиваемой услуги или товаров (при наличии), об иных данных, указанных при проведении денежной операции), осуществленных транзакциях (с указанием идентификатора платежной системы, суммы прихода либо расхода, иных данных, указанных при осуществлении транзакции);

4) 5 минут – для текстовых сообщений пользователей сети «Интернет», голосовой информации, изображений, звуков, видео-, иных электронных сообщений пользователей сети «Интернет»;

5) 10 секунд – для иной информации.

19. Продолжительность выполнения поисковой задачи ПТС ОРИ не должна превышать следующих значений:

1) 5 секунд – для данных о принадлежности идентификатора пользователя ИС ОРИ;

2) 10 секунд – для данных, внесенных при регистрации пользователя, дополненных, измененных в регистрационных данных пользователя;

3) 10 секунд – по идентификатору пользователя для данных об изменениях, связанных с пользователем идентификаторов других пользователей ИС ОРИ;

4) 10 секунд – для иной информации.

20. Продолжительность выполнения задач поиска информации о фактах приема, передачи, доставки и (или) обработки голосовой информации, текстовых сообщений, изображений, звуков, местоположения, случаях авторизации или деавторизации пользователей в ИС ОРИ, или иных электронных сообщений пользователей сети «Интернет», о зарегистрированных пользователях, изменениях, дополнениях регистрационных данных, текстовых сообщений пользователей сети «Интернет», голосовой информации, изображений, звуков, видео-, иных электронных сообщений пользователей сети «Интернет», иной информации, накопленных в ПТС ОРИ, не должна превышать значений, приведенных в таблице № 3 настоящих требований.

21. ПТС ОРИ должны обеспечивать одновременное выполнение устанавливаемых с ПУ поисковых задач в соответствии с временными и количественными характеристиками, указанными в таблицах № 3 и № 4 настоящих требований.

Таблица № 3.

№№ п.п.	Содержание запроса	Время выполнения запроса, секунда		
		За интервал до суток	За интервал до 30 суток	За интервал до 180 суток
1	Запрос по информации о регистрационных данных пользователя коммуникационного интернет-сервиса	< 5	< 10	< 30
2	Стандартные запросы по информации о фактах приема, передачи и (или) обработки голосовой информации, текстовых сообщений текста, изображений, звуков или иных электронных сообщений пользователей сети «Интернет» (информация о точном времени приема, передачи, доставки и (или) обработки электронных сообщений с указанием информации об адресатах этих сообщений, в том числе	< 5	< 15	< 30

	информации для неопределенного круга лиц пользователей сети «Интернет»)			
3	Иные запросы по информации о фактах приема, передачи и (или) обработки голосовой информации, текстовых сообщений, изображений, звуков, видео или иных электронных сообщений пользователей сети «Интернет» (информация о точном времени приема, передачи, доставки и (или) обработки электронных сообщений с указанием информации об адресатах этих сообщений, в том числе информации для неопределенного круга лиц пользователей сети «Интернет», а также содержимое текстовых сообщений)	< 10	< 30	< 60
4	Запрос по информации о фактах авторизаций и выхода из информационного сервиса, регистрации, прекращения регистрации	< 5	< 10	< 30
5	Запрос по получению содержимого голосовой информации, изображений, звуков, видео-, иных электронных сообщений	< 2	< 5	< 10
6	Запрос иной информации, необходимой для выполнения возложенных на уполномоченные органы задач	< 10	< 30	< 60

22. ПТС ОРИ должны обеспечивать одновременную передачу на ПУ результатов выполнения поисковых задач в соответствии с временными характеристиками, указанными в таблице № 3.

23. Количественные характеристики одновременно выполняемых ПТС ОРИ поисковых задач по запросам ПУ и количество одновременно передаваемых на ПУ результатов выполнения поисковых задач приведены в таблице № 4.

Таблица № 4.

№	Содержание запроса	Количество одновременно выполняемых ПТС ОРИ поисковых задач по запросам ПУ с учетом коэффициента «К»*	Количество одновременно передаваемых на ПУ результатов выполнения поисковых задач с учетом коэффициента «К»*
1	Запрос по информации о регистрационных данных пользователя коммуникационного интернет-сервиса	не менее $K \times 5$	не менее $K \times 5$
2	Стандартные запросы по информации о фактах приема, передачи и (или) обработки голосовой информации, текстовых сообщений текста, изображений, звуков или иных электронных сообщений пользователей сети «Интернет» (информация о точном времени приема, передачи, доставки и (или) обработки электронных сообщений с указанием информации об адресатах этих сообщений, в том числе информации для неопределенного круга лиц пользователей сети «Интернет»)	не менее $K \times 3$	не менее $K \times 3$

3	Иные запросы по информации о фактах приема, передачи и (или) обработки голосовой информации, текстовых сообщений, изображений, звуков, видео или иных электронных сообщений пользователей сети «Интернет» (информация о точном времени приема, передачи, доставки и (или) обработки электронных сообщений с указанием информации об адресатах этих сообщений, в том числе информации для неопределенного круга лиц пользователей сети «Интернет», а также содержимое текстовых сообщений)	не менее $K \times 3$	не менее $K \times 3$
4	Запрос по информации о фактах авторизаций и выхода из информационного сервиса, регистрации, прекращения регистрации	не менее $K \times 5$	не менее $K \times 5$
5	Запрос по получению содержимого голосовой информации, изображений, звуков, видео-, иных электронных сообщений	не менее $K \times 10$	не менее $K \times 3$
6	Запрос иной информации, необходимой для выполнения возложенных на уполномоченные органы задач	не менее $K \times 3$	не менее $K \times 3$
<p>* «К» является коэффициентом, зависящим от количества зарегистрированных пользователей ОРИ, равным:</p> <p>$K = 1$ при количестве менее 1 тысячи зарегистрированных пользователей; $K = 3$ при количестве (1 – 10) тысяч зарегистрированных пользователей; $K = 5$ при количестве (10 – 100) тысяч зарегистрированных пользователей; $K = 7$ при количестве (100 – 500) тысяч зарегистрированных пользователей; $K = 10$ при количестве (500 – 1000) тысяч зарегистрированных пользователей; $K = 15$ при количестве (1000 – 10000) тысяч зарегистрированных пользователей; $K = 20$ при количестве более 10000 тысяч зарегистрированных пользователей.</p>			

24. ПТС ОРИ должны поддерживать следующие логические операции для объединения поисковых критериев: логического «И», логического «ИЛИ», операции группировки критериев «(», «)» и логическое «НЕ».

25. Применение логического «НЕ» допускается в случаях, если к его результату или результату включающей его операции объединения применяется операция логического «И» с допустимым поисковым критерием.

При этом необходимо учитывать следующее:

операция логического отрицания имеет наивысший приоритет;

приоритет логического «И» выше приоритета логического «ИЛИ»;

последовательности одинаковых логических операций вычисляются последовательно;

для изменения приоритета допускается использовать скобки.

26. К временным параметрам поиска информации в базе ПТС ОРИ при выполнении комбинированных поисковых запросов устанавливаются следующие требования:

1) для операции «И»: интервал времени выполнения поискового запроса по двум и более критериям должен быть меньше суммарного интервала времени выполнения поисковых запросов по каждому из критериев;

2) для операции «ИЛИ»: интервал времени выполнения поискового запроса по двум и более критериям должен быть не больше суммарного интервала времени выполнения поисковых запросов по каждому из критериев;

3) для операции «НЕ»: интервал времени выполнения поискового запроса, в котором один из критериев задан с операцией «НЕ», не должен превышать интервала времени выполнения аналогичного запроса, в котором используется соответствующий критерий без операции «НЕ»;

4) в случае если одним из поисковых критериев является составной критерий с использованием операций группировки, для определения временных интервалов выполнения исходного запроса, интервал времени выполнения запроса с составным критерием должен соответствовать требованиям подпунктов 1, 2 и 3 настоящего пункта.

27. ПТС ОРИ должны поддерживать возможность выполнения поисковых запросов с использованием символов-заменителей «*», «?» в каждом критерии поиска, за исключением интервала времени поступления информации в систему хранения. В случае использования символов-заменителей интервал времени выполнения задачи ПТС ОРИ может превышать значения, приведенные в Таблице № 3 настоящих требований.

Применение символов-заменителей «*», «?» допускается в случае, если к его результату или результату включающей его операции объединения применяется операция логического «И» с допустимым поисковым критерием.

Приложение № 1
к Требованиям к оборудованию и программно-техническим средствам, используемым организатором распространения информации в сети «Интернет» в эксплуатируемых им информационных системах, для проведения уполномоченными государственными органами, осуществляющими оперативно-розыскную деятельность или обеспечение безопасности Российской Федерации, мероприятий в целях реализации возложенных на них задач, утвержденным приказом Министерства цифрового развития, связи и массовых коммуникаций Российской Федерации от 19.10.2018 2018 № 571

Требования к ПТС ОРИ в части обработки информации

1. ПТС ОРИ должны обрабатывать следующую информацию:

1.1 сведения о пользователях и регистрационных данных, в том числе регистрационных и иных данных, изменениях и дополнениях регистрационных данных, внесенных пользователем:

1) сведения о пользователе, включая идентификатор пользователя в коммуникационном интернет-сервисе;

2) сведения регистрационных данных:

информация о сетевом адресе, с которого осуществлена регистрация пользователя, с указанием времени регистрации;

информация, которая должна вноситься в ИС ОРИ при регистрации пользователя:

а) дата и время регистрации пользователя;

б) номер договора, дата и время заключения договора на обслуживание пользователя в ИС ОРИ (при наличии такой информации);

информация, внесенная пользователем:

а) фамилия, имя, отчество (при наличии);

б) псевдоним пользователя;

в) дата рождения;

г) адрес места жительства;

д) реквизиты документа, удостоверяющего личность;

е) список идентификаторов в ИС ОРИ для лиц, указанных пользователем в качестве родственников;

внесенные пользователем принадлежащие ему идентификаторы в других средствах электронного взаимодействия, в том числе наименование сервиса и идентификатор;

дата и время последнего обновления пользователем регистрационной информации;

дата и время прекращения регистрации пользователя в ИС ОРИ;

дополнительная информация о пользователе, фиксируемая ИС ОРИ;

1.2 сведения о добавлении (исключении) зарегистрированным пользователем связанных с ним других зарегистрированных идентификаторов:

дата и время;

вид коммуникационного интернет-сервиса (oriServices); идентификатор пользователя ИС ОРИ, устанавливающего связь;

вид события (oriEvents);

идентификатор пользователя (ресурса) в ИС ОРИ, с которым устанавливается связь;

1.3 сведения о фактах авторизации и выхода из коммуникационного интернет-сервиса, прекращениях регистрации:

дата и время;

вид коммуникационного интернет-сервиса (oriServices);

идентификатор пользователя ИС ОРИ;

технические идентификаторы пользователя ИС ОРИ, переданные в ИС ОРИ в силу используемых коммуникационных протоколов:

IP-адрес и порт;

номера телефонов пользователя, являющегося абонентом (пользователем) услуг подвижной радиотелефонной связи;

адрес электронной почты;

наименование программы клиента (текстовая строка в произвольном виде, содержащая сведения о наименовании программы, версии, дате обновления, а также иные регистрируемые сведения при передаче);

иная фиксируемая техническая информация;

вид события (oriEvents);

коммуникационный интернет-сервис ИС ОРИ (telcosRecords);

1.4 информацию о фактах приема, передачи и (или) обработки голосовой информации, текстовых сообщений, изображений, звуков, видео-или иных электронных сообщений пользователей сети «Интернет» (информация о точном времени приема, передачи, доставки и (или) обработки электронных сообщений с указанием информации об адресатах этих сообщений, в том числе информации для неопределенного круга лиц пользователей сети «Интернет»), фиксируемая информация об организации приема, передачи, доставки и (или) обработки электронных сообщений, осуществляемых с использованием технологий электронных платежных систем, в том числе информация о произведенных денежных операциях (с указанием информации о корреспонденте - идентификаторе платежной системы, валюты, суммы, оплачиваемой услуги или товаров (при наличии), об иных данных, указанных при проведении денежной операции),

осуществленных транзакциях (с указанием идентификатора платежной системы, суммы прихода либо расхода, иных данных, указанных при осуществлении транзакции):

- а) дата и время события;
- б) вид информационного сервиса;
- в) идентификатор пользователя ИС ОРИ;
- г) технические идентификаторы пользователя ИС ОРИ переданные в ИС

в силу используемых коммуникационных протоколов:

IP-адрес и порт;

номер телефона (MSISDN) подвижной станции пользователя;

адрес электронной почты;

наименование программы клиента;

иная фиксируемая техническая информация;

д) вид события (oriEvents);

е) идентификаторы пользователей ИС ОРИ, с которыми было взаимодействие (контакты);

ж) местоположение пользователя в случае его фиксации ИС ОРИ (широта/долгота, иное описание местоположения);

з) информационный ресурс созданный в ИС ОРИ, на котором выполнено информационное взаимодействие (тип ресурса, полное наименование ресурса);

и) техническая информация о ресурсе;

к) идентификатор пользователя ИС ОРИ-владельца ресурса;

л) текст сообщения и иная служебная коммуникационная информации;

м) техническая информация о платеже:

вид платежной услуги (платежного сервиса);

идентификатор платежа в сервисе;

иная фиксируемая информация о платеже;

коммуникационный интернет-сервис ИС ОРИ (telcosRecords);

1.5 информацию по пунктам 3 и 4 настоящего приложения в соответствии с приложением № 4 к настоящим требованиям (ReportsConnectionsORI.asn);

1.6 информацию о зарегистрированных пользователях ИС ОРИ в соответствии с Приложением № 4 к настоящим требованиям (ReportsAbonentsORI.asn);

1.7 справочную информацию в ИС ОРИ в соответствии с приложением № 4 к настоящим требованиям (Dictionaries.asn):

а) виды информационных сервисов, предоставляемых ОРИ для пользователей;

б) виды событий, регистрируемых ИС ОРИ при взаимодействии ИС ОРИ с пользователем;

в) типы информационных ресурсов, создаваемых пользователями в ИС ОРИ;

г) типы пользователей, обслуживаемых ИС ОРИ;

д) виды платежных услуг (платежных сервисов), используемых ОРИ;

е) коммуникационные интернет-сервисы ИС ОРИ, обслуживаемые ПТС ОРИ;

1.8 информацию, необходимую для декодирования принимаемых, передаваемых, доставляемых и (или) обрабатываемых электронных сообщений.

Приложение № 2
к Требованиям к оборудованию и программно-техническим средствам, используемым организатором распространения информации в сети «Интернет» в эксплуатируемых им информационных системах, для проведения уполномоченными государственными органами, осуществляющими оперативно-розыскную деятельность или обеспечение безопасности Российской Федерации, мероприятий в целях реализации возложенных на них задач, утвержденным приказом Министерства цифрового развития, связи и массовых коммуникаций Российской Федерации от 29.10.2018 2018 № 571

Требования, предъявляемые к интерфейсу взаимодействия между ПУ и ПТС ОРИ

1. ПТС ОРИ должны подключаться к ПУ в точках подключения выделенных каналов связи.

2. В качестве набора протоколов передачи данных следует использовать набор протоколов ТСР/ІР для:

2.1 организации линий (каналов) связи, соединяющих ПТС ОРИ и ПУ, должен быть использован сетевой интерфейс первого уровня;

2.2 ПТС ОРИ должна предусматриваться возможность создания виртуальной сети VPN (Virtual Private Network) для туннелирования всего рабочего ТСР/ІР трафика между ПТС ОРИ и ПУ;

2.3 получения ПУ иной информации, предоставляемой ПТС ОРИ должен организовываться отдельный канал передачи данных с использованием средств криптографической защиты информации.

3. Взаимодействие ПУ и ПТС ОРИ должно осуществляться по четырем каналам передачи данных (далее – кпд):

кпд1 – канал управления (Приложение № 5 к настоящим требованиям);

кпд2 – канал данных (Приложение № 6 к настоящим требованиям);

кпд3 – канал мониторинга (Приложение № 7 к настоящим требованиям);

кпд4 – канал неформатированных данных (Приложение № 8 к настоящим требованиям).

4. Канал управления (кпд1) предназначен для передачи:

1) от ПУ в ПТС ОРИ запросов (команд);

2) от ПТС ОРИ на ПУ ответов и сигналов.

5. Канал данных (кпд2) предназначен для передачи:
 - 1) от ПТС ОРИ на ПУ блоков данных отчетов, генерируемых ПТС ОРИ в качестве ответов на запросы от ПУ, сигнала HeartBeat;
 - 2) от ПУ в ПТС ОРИ подтверждений о принятии блоков данных отчетов.
6. Канал мониторинга (кпд3) предназначен для передачи:
 - 1) от ПУ в ПТС ОРИ запросов о текущей конфигурации оборудования, системного и прикладного ПО ПТС ОРИ и запросов на модификацию конфигурации;
 - 2) от ПТС ОРИ на ПУ ответов на запросы ПУ сигналов.
7. Канал неформатированных данных (кпд4) предназначен для передачи:
 - 1) от ПУ в ПТС ОРИ команд на виды передаваемых неформатированных данных;
 - 2) от ПТС ОРИ на ПУ неформатированных данных, сигналов.
8. Каналы передачи данных (кпд1, кпд2, кпд3, кпд4) должны подключаться к заранее определенным портам ПТС ОРИ в виде TCP-соединений. Номера портов должны передаваться на ПУ на внешних носителях. ПТС ОРИ должен выполнять мониторинг данных портов для создания TCP-соединений с ПУ.
9. ПУ должны осуществлять попытки установления подключения к ПТС ОРИ в соответствии с задаваемым интервалом по предоставленным ПТС ОРИ TCP-портам.
10. Установление соединений ПУ с ПТС ОРИ по каналам кпд1, кпд2 необходимо осуществлять в следующей последовательности:
 - 1) установление ПУ TCP-соединения к ПТС ОРИ по порту канала кпд1;
 - 2) выполнение процедуры взаимной SSL/TLS аутентификации в соответствии с пунктом 28 настоящего приложения;
 - 3) установление ПУ TCP-соединения к ПТС ОРИ по порту канала кпд2;
 - 4) выполнение процедуры взаимной SSL/TLS аутентификации в соответствии с пунктом 28 настоящего приложения;
 - 5) после успешной аутентификации создание ПУ сессии к ПТС ОРИ;
 - 6) ожидание ПУ данных и сигналов по кпд2 только после того, как была создана сессия по каналу кпд1. При приеме данных и сигналов по кпд2 при отсутствии установленной сессии по кпд1 ПУ должен разорвать соединение по кпд2.
11. Установление ПУ соединений к ПТС ОРИ по каналам кпд3 и кпд4 необходимо осуществлять независимо друг от друга и от наличия установленных соединений по каналам кпд1 и кпд2:
 - 1) установление ПУ TCP-соединения к ПТС ОРИ по TCP-порту канала кпд3/кпд4;
 - 2) выполнение процедуры взаимной SSL/TLS аутентификации в соответствии с пунктом 28 настоящего приложения;
 - 3) после успешной аутентификации создание ПУ сессии к ПТС ОРИ;
 - 4) после успешной аутентификации посылка ПУ на ПТС ОРИ команды в соответствии с приложением № 4 к настоящим требованиям.

12. ПУ должен разорвать соединения к ПТС ОРИ, если в течение трех периодов приема сигнала HeartBeat в каналах не было сетевой активности. При отсутствии подтверждения посланного сигнала HeartBeat в течение времени «максимальный размер задержки подтверждения запроса или сигнала» ПТС ОРИ должны разорвать соединения с ПУ по соответствующему каналу.

13. Обмен данными в кпд1, кпд2, кпд3, кпд4 должен осуществляться с использованием единицы обмена «Сообщение» (Message) в соответствии с форматами, представленными в приложении № 3 к настоящим требованиям на языке абстрактного описания синтаксиса протокола взаимодействия пункта управления ОРМ и технических и программных средств ПТС ОРИ согласно ГОСТ Р ИСО/МЭК 8824-1-2001¹. Способ кодирования сериализованных «Сообщений» должен соответствовать отличительным (DER) правилам кодирования по ГОСТ Р ИСО/МЭК 8825-1-2003².

14. Интерфейс взаимодействия ПУ с ПТС ОРИ должен предусматривать использование следующих «Сообщений»:

«запросы» – передаются от ПУ в ПТС ОРИ по кпд1;

«ответы» – передаются из ПТС ОРИ на ПУ по кпд1;

«сигналы» – передаются из ПТС ОРИ на ПУ по кпд1 и кпд2 (для кпд2 только «сигнал» Heartbeat);

«отчеты» – формируются ПТС ОРИ в качестве ответов на запросы из ПУ, передаются на ПУ по кпд2;

«подтверждения» о принятии «отчетов» – передаются из ПУ в ПТС ОРИ по кпд2;

«подтверждения» о принятии «сигналов» – передаются из ПУ в ПТС ОРИ по кпд1 и кпд2 (для кпд2 только для сигнала Heartbeat);

15. Интерфейс взаимодействия между ПУ и ПТС ОРИ по каналу кпд3 должен предусматривать использование следующих «Сообщений»:

«запросы»;

«ответы»;

«сигнал» Heartbeat;

«подтверждения» о принятии «сигнала» Heartbeat и «ответов»;

16. Интерфейс взаимодействия между ПУ и ПТС ОРИ по каналу кпд4 должен предусматривать использование следующих «Сообщений»:

«запросы» – передаются от ПУ в ПТС ОРИ;

«ответы» – передаются из ПТС ОРИ на ПУ;

«сигналы» – передаются из ПТС ОРИ на ПУ;

¹ ГОСТ Р ИСО/МЭК 8824-1-2001 «Информационная технология. Абстрактная синтаксическая нотация версии один (ASN.1). Часть 1. Спецификация основной нотации», введенного в действие постановлением Государственного комитета Российской Федерации по стандартизации и метрологии от 6 сентября 2001 г. № 375-ст (ИПК Издательство стандартов, 2001)

² ГОСТ Р ИСО/МЭК 8825-1-2003 «Информационная технология. Правила кодирования. Часть 1. Спецификация базовых (BER), канонических (CER) и отличительных (DER) правил кодирования», введенному в действие постановлением Государственного комитета Российской Федерации по стандартизации и метрологии от 13 мая 2003 г. № 140-ст (ИПК Издательство стандартов, 2003)

«отчеты» – формируются ПТС ОРИ в качестве «ответов» на запросы от ПУ;

«подтверждения» о принятии «отчетов» – передаются от ПУ в ПТС ОРИ;

«подтверждения» о принятии «сигналов» – передаются от ПУ в ПТС ОРИ.

17. ПТС ОРИ должны выполнять любые действия, связанные с выдачей информации о пользователях, управлением и мониторингом ПТС ОРИ, только по «запросам» с ПУ.

18. ПТС ОРИ должны обеспечивать прием и обработку следующих сообщений, передаваемых с ПУ:

1) «Запрос на открытие сессии» (ConnectRequest);

2) «Запрос на закрытие сессии» (DisconnectRequest);

3) «Запрос готовности данных» (DataReadyRequest);

4) «Запрос загрузки данных» (DataLoadRequest);

5) «Запрос удаления данных» (DataDropRequest);

6) «Запрос прерывания загрузки данных» (DataInterruptRequest);

7) «Запрос на создание задачи по обработке информации» (CreateTaskRequest) – предназначен для создания запросов по информации:

«Задачи пополнения справочников (нормативно-справочная информация)» (DictionaryTask);

«Задачи поисков по принадлежности пользователей» (AbonentsTask);

«Задачи поисков по коммуникации пользователей» (ConnectionsTask);

«Задачи получения данных местоположения пользователей» (LocationTask);

«Задачи поисков по совершенным платежам» (PaymentsTask);

«Задачи предоставления сведений о наличии данных» (PresenseTask);

«Задача получения сообщений пользователей коммуникационного интернет-сервиса» (DataContentTask);

8) «Запрос на создание задачи по обработке неформализованных данных» (NonFormalizedTaskRequest) – предназначен для получения иной информации, не определяемой подпунктом 7 настоящего пункта и предоставляемой при технической возможности ПТС ОРИ;

9) «Запрос на создание объекта контроля» (CreateUNIRrequest);

10) «Запрос на снятие объекта с контроля» (DropUNIRrequest).

19. В зависимости от типа завершённой поисковой задачи передача блоков данных должна осуществляться по каналу кпд2.

20. ПТС ОРИ должны обеспечивать:

1) одновременную передачу блоков данных отчетов для нескольких завершённых поисковых задач по каналам кпд2 - кпд4. ПТС ОРИ должны обеспечивать возможность многократной передачи отчетов выполненных задач на ПУ;

2) одновременную передачу блоков данных отчетов по разным объектам контроля по каналу кпд2.

21. По запросу ПУ «Запрос загрузки данных» по каналу кпд1 ПТС ОРИ должны обеспечивать передачу блоков отчетов по каналам кпд2 - кпд4

по запрошенной задаче без внесения дополнительных временных задержек между операциями по получению записей результата задачи и преобразования их в блоки.

22. На каждый «запрос» по кпд1 ПТС ОРИ на ПУ должны посылать «ответ» о принятии к обработке этого запроса. ПТС ОРИ должны обеспечивать посылку по кпд1 на ПУ следующих «ответов»:

- 1) «Ответ на запрос открытия сессии» (ConnectResponse);
- 2) «Ответ на запрос закрытия сессии» (DisconnectResponse);
- 3) «Ответ на запрос готовности данных» (DataReadyResponse);
- 4) «Ответ на запрос загрузки данных» (DataLoadResponse);
- 5) «Ответ на запрос удаления данных» (DataDropResponse);
- 6) «Ответ на запрос прерывания загрузки данных» (DataInterruptResponse);
- 7) «Ответ на запрос на создание задачи» (TaskResponse);
- 8) «Ответ на запрос на создание задачи по обработке неформализованных данных» (NonFormalizedTaskResponse);
- 9) «Ответ на запрос на создание объекта контроля» (UNIControlTaskResponse);
- 10) «Ответ на запрос на снятие объекта с контроля» (CreateUNIResponse).

23. По запросу ПУ «Запрос на создание задачи по обработке информации» ПТС ОРИ должны обеспечивать подготовку и выдачу информации из ИС ОРИ и (или) ПТС ОРИ, для следующих групп задач:

- 1) «Задачи пополнения справочников (справочная информация)» (DictionaryTask):

виды информационных сервисов, предоставляемых ОРИ для пользователей;

виды событий, регистрируемых ИС ОРИ при взаимодействии ИС ОРИ с пользователем;

типы информационных ресурсов, создаваемых пользователями в ИС ОРИ;

типы пользователей, обслуживаемых ИС ОРИ;

виды платежных услуг (платежных сервисов), используемых ОРИ;

коммуникационные интернет-сервисы ИС ОРИ, обслуживаемые ПТС ОРИ;

- 2) «Задачи поисков по принадлежности пользователей» (AbonentsORITask):

«Задача на поиск принадлежности пользователей организаторов распространения информации»;

«Задача на поиск изменений в связанных объектах ИС ОРИ с пользователем»;

- 3) «Задачи поисков по соединениям пользователей» (ConnectionsORITask):

«Задача на поиск информации о фактах информационного взаимодействия»;

«Задача на поиск авторизаций, выходов из информационного сервиса, регистрации, прекращения регистрации»;

4) «Задача предоставления сведений о наличии данных» (PresenseTask).

24. ПТС ОРИ должны обеспечивать выполнение поисковых задач по текстовым критериям, содержащими символы-заменители, включающие:

- 1) «*» – любая комбинация символов;
- 2) «?» – любой один возможный символ.

25. ПТС ОРИ должны обеспечивать выполнение поисковых задач по критериям, содержащим последовательность цифр с символом маскирования пробел (« »), означающим любую одну цифру.

Результат выполнения поисковой задачи с критерием, содержащим символы маскирования, должен содержать все записи, соответствующие заданной маске.

26. ПТС ОРИ должны обеспечивать контроль функционирования собственных параметров и передачу на ПУ следующих сигналов:

- 1) «Перезапуск ПО» (RestartDB);
 - 2) «Попытка несанкционированного доступа» (UnauthorizedAccess);
 - 3) «Критическая ошибка ПО, потеря данных, дальнейшая работа невозможна» (CriticalError);
 - 4) «Серьезная ошибка ПО, потеря данных, но дальнейшая работа возможна» (MajorError);
 - 5) «Незначительная ошибка ПО, данные не потеряны, дальнейшая работа возможна» (MinorError);
 - 6) «Тестовый пакет» (Heartbeat) должен передаваться по кпд1, кпд2, кпд3, кпд4 для поддержания соединений в отсутствие иной сетевой активности.
- В ответ на «сигналы», поступившие от ПТС ОРИ, ПУ должен передавать «подтверждения сигнала» об их приеме.

27. ПТС ОРИ по «Запросу удаления данных» (DataDropRequest) должен осуществлять:

- 1) прерывание задачи, находящейся на выполнении;
- 2) удаление данных отчета по завершившейся задаче.

28. По запросу ПУ «Запрос на создание объекта контроля» (CreateUNIRRequest) ПТС ОРИ должны:

производить отбор из поступающей в ИС ОРИ информации по критериям отбора, заданным в запросе ПУ;

передавать на ПУ статистическую информацию, а также содержимое сообщений (в зависимости от режима контроля, заданного при постановке объекта на контроль в запросе CreateUNIRRequest);

передавать отобранные данные по каналу кпд2 одновременно с передачей на ПУ информации по поисковым задачам.

29. По запросу ПУ «Запрос на снятие объекта с контроля» (DropUNIRRequest) ПТС ОРИ должны обеспечивать:

прекращение отбора информации по критериям отбора, заданным в запросе ПУ;

прекращение передачи на ПУ статистической информации, а также содержимого сообщений, отобранных по критериям отбора, заданным в запросе ПУ для объекта, снимаемого с контроля.

30. Соединение ПУ и ПТС ОРИ должно устанавливаться по протоколу TLS актуальной версии. При установлении соединения ПУ и ПТС ОРИ должны осуществлять взаимную аутентификацию. Для аутентификации ПУ и ПТС ОРИ используются сертификаты в формате X.509.

В случае невозможности аутентифицировать одну из сторон TCP-соединение должно разрываться. Созданный для ПУ сертификат должен использоваться для аутентификации ПУ на одном ПТС ОРИ по всем каналам передачи данных: кпд1, кпд2, кпд3, кпд4 ПУ и ПТС ОРИ.

31. Установление сеанса связи должно осуществляться выполнением процедуры аутентификации перед выполнением первого запроса. Открытие сессии должно осуществляться посылкой по каналу кпд1 от ПУ к ПТС ОРИ «Запроса на открытие сессии» (ConnectRequest).

32. «Запрос на открытие сессии» должен устанавливать следующие параметры сессии:

1) интервал времени, по истечении которого сессия принудительно прерывается ПТС ОРИ («максимальное время отсутствия активности сессии» (session-timeout);

2) максимальный размер блока данных отчетов в строках записей ПТС ОРИ («максимальный размер блока данных отчетов» (max-data-length);

3) «размер окна для канала передачи данных» (data-packet-window-size);

4) «максимальная длительность задержки начала передачи блоков отчетов» (data-load-timeout);

5) «максимальный размер задержки подтверждения о получении данных» (data-packet-response-timeout);

6) «максимальный размер задержки подтверждения запроса или сигнала» (request-response-timeout).

Любое сообщение в соответствии с протоколом взаимодействия ПТС ОРИ и ПУ считается сетевой активностью.

33. ПТС ОРИ при получении сообщения «Запрос на открытие сессии» должны:

анализировать параметр «Размер окна для канала передачи данных»;

определять максимально возможный «Размер окна для канала передачи данных», который не должен превышать полученного от ПУ;

определять минимальные значения таймаутов из параметров сессии, выбирая их не меньше, чем переданные в сообщении «Запрос на открытие сессии» (ConnectRequest).

Расчитанные значения размеров окна и таймаутов ПТС ОРИ должны передавать ПУ в сообщении «Ответ на открытие сессии» (ConnectResponse). Полученные ПУ значения в сообщении «Ответ на открытие сессии» являются параметрами сессии между ПУ и ПТС ОРИ.

34. Закрытие сессии должно осуществляться посылкой по кпд1 или кпд3 от ПУ к ПТС ОРИ «Запроса на закрытие сессии» или по истечению

допустимого времени отсутствия активности ПТС ОРИ, с посылкой сообщения-сигнала «Прерывание текущей сессии по таймауту». При этом ПТС ОРИ и ПУ должны осуществлять разрыв текущих ТСР-соединений кпд1 и кпд2 или кпд3, или кпд4.

35. ПУ должен посылать ПТС ОРИ «запросы» асинхронно, независимо от получения от ПТС ОРИ «ответа» о приеме предыдущего «запроса».

36. «Запрос на создание задачи по обработке информации» должен приводить к созданию в ПТС ОРИ задачи по обработке данных, которой присваивается номер (идентификатор) задачи, передаваемый в «Ответе на запрос создания задачи» (TaskResponse). ПТС ОРИ должны генерировать уникальные идентификаторы задач независимо от сессий.

37. ПУ, послав запрос «Запрос готовности данных», должен получить информацию о ходе выполнения и завершения обработки задач в ПТС ОРИ. После завершения выполнения задачи, данные, сформированные в результате выполнения задачи, становятся доступными для загрузки в ПУ или для удаления.

38. ПТС ОРИ должны поддерживать приостановку выполнения задачи с сохранением промежуточных результатов и возможностью их выгрузки на ПУ и (или) удаления.

39. ПТС ОРИ при получении «Запрос загрузки данных» по кпд1 должны сформировать сообщения типа «отчет», состоящие из блоков данных обработанной задачи и передать их на ПУ по кпд2.

40. Количество строк в каждом блоке не должно превышать параметр «максимальный размер блока данных отчетов», заданный при открытии сессии.

41. В каждом последовательном блоке данных из серии должен указываться идентификатор задачи, сгенерировавшей данный отчет, общее количество блоков в отчете, порядковый номер каждого блока.

42. Задачи, полученные в одной сессии, могут быть считаны и (или) удалены инициировавшим данную задачу ПУ в другой сессии. Данные по завершенной задаче должны быть доступны между сессиями по идентичному идентификатору задачи.

43. ПТС ОРИ должны удалить без возможности восстановления данные, сформированные в результате выполнения задачи и самой выполненной задачи, после поступления с ПУ запроса на удаление данных.

В случае возникновения в ПТС ОРИ или кпд1 – кпд4 критических ситуаций на ПУ должны передаваться «Сообщения» типа «Сигнал».

Приложение № 3
к Требованиям к оборудованию и программно-техническим средствам, используемым организатором распространения информации в сети «Интернет» в эксплуатируемых им информационных системах, для проведения уполномоченными государственными органами, осуществляющими оперативно-розыскную деятельность или обеспечение безопасности Российской Федерации, мероприятий в целях реализации возложенных на них задач, утвержденным приказом Министерства цифрового развития, связи и массовых коммуникаций Российской Федерации от 29.10.2018 2018 № 571

Требования к форматам сообщений ПТС ОРИ

1. Структура видов сообщений протокола взаимодействия ПУ с ПТС ОРИ приведена на схеме 1.

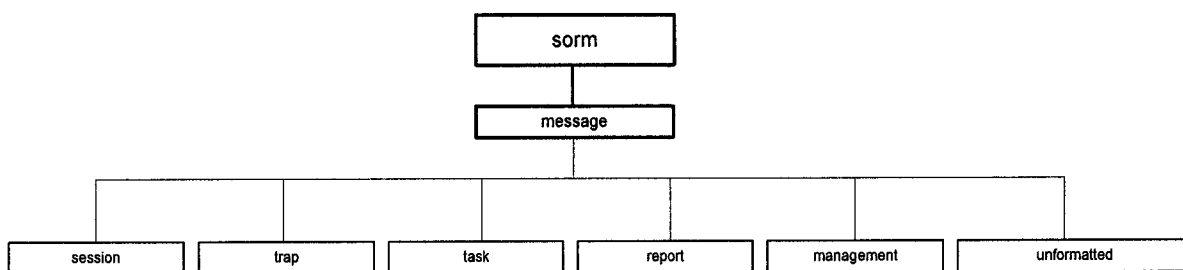
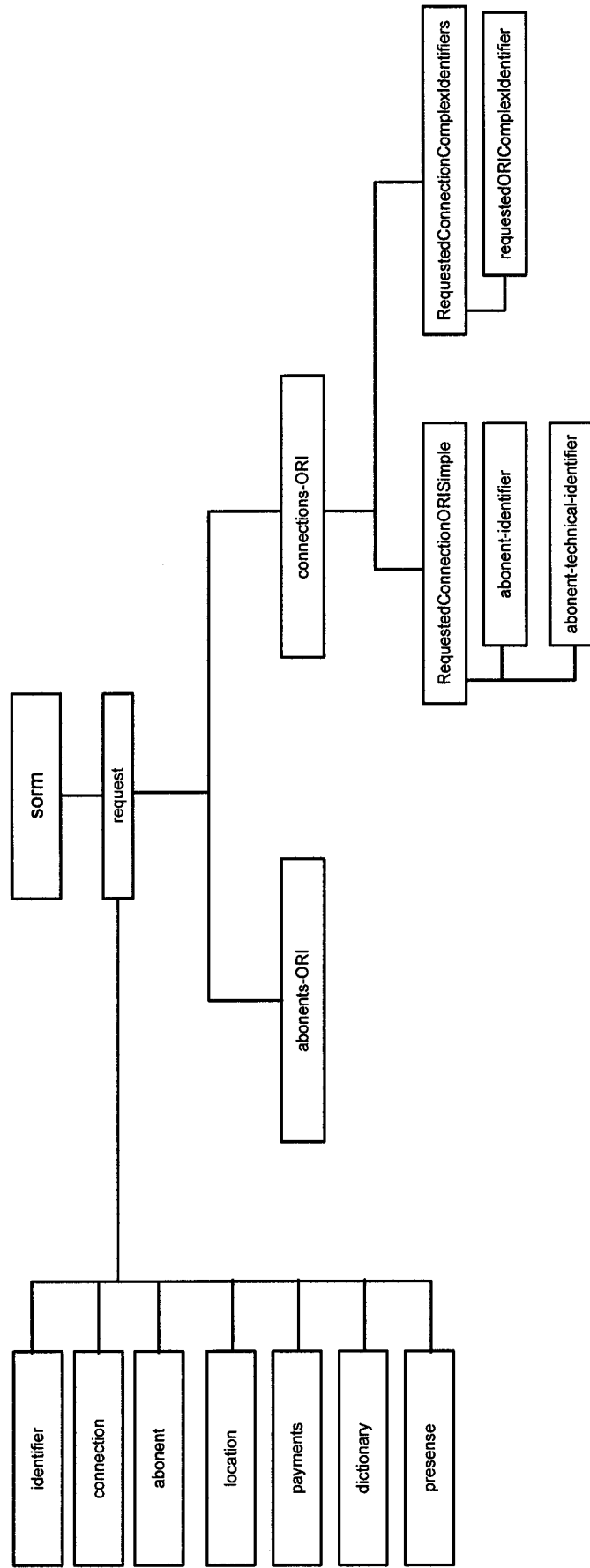


Схема 1

2. Структура разделения поисковых критериев кпд1 приведена на схеме 2.



Cxema 2

3. Структура разделения видов отчетов кпд2 приведена на схеме 3.

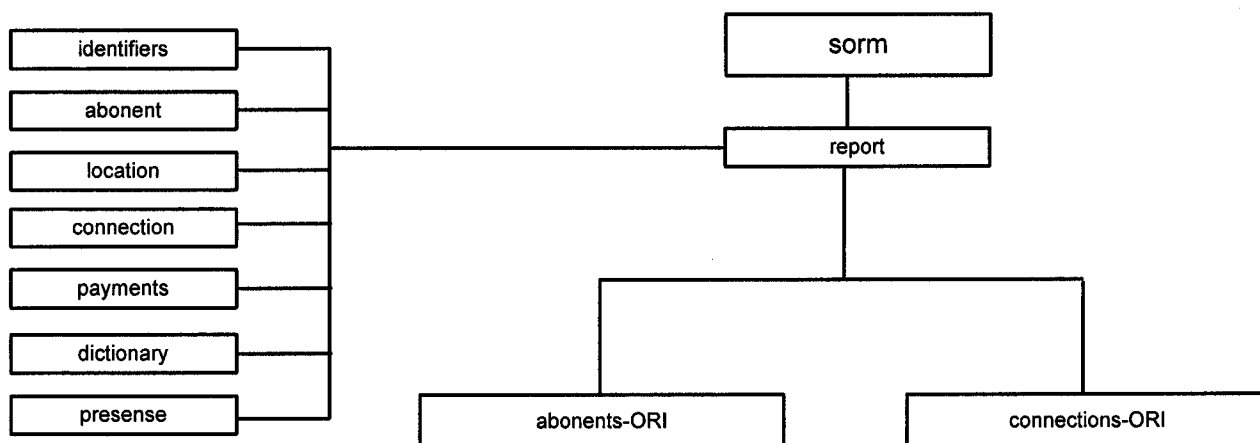


Схема 3

4. Структура модулей протокола взаимодействия ПУ и ПТС ОРИ (ASN.1) приведена на схеме 4.

5. Модуль Classification.asn должен задавать правила, в соответствии с которыми:

1) выполняется расширение:

списка типов запросов к ПТС ОРИ;

списка видов поисковых критериев ПТС ОРИ;

списка типов отчетов, формируемых ПТС ОРИ;

2) выполняется ввод новых версий сообщений протокола.

6. В соответствии со схемами 2, 3 в модулях протокола взаимодействия ПУ и ПТС ОРИ (ASN.1) должна выполняться подстановка соответствующих версий форматов поисковых критериев, отчетов, справочников, «сообщений» интерфейса взаимодействия ПУ и ПТС ОРИ (ASN.1).

7. Модуль «Classification.asn» протокола взаимодействия ПУ и ПТС ОРИ (ASN.1) должен содержать кодированные в иерархическом виде идентификаторы:

1) видов «Сообщений» верхнего уровня интерфейса взаимодействия ПУ и ПТС ОРИ, составляющих кпд1, кпд2, кпд3, кпд4;

2) видов поисковых критериев для формирования задач к ПТС ОРИ;

3) видов форматов отчетов, формируемых ПТС ОРИ.

8. Соответствующие идентификаторы используются в других модулях интерфейса взаимодействия ПУ с ПТС ОРИ (ASN.1), при этом идентификатор должен определять конкретную версию и расширения формата соответствующего элемента (поисковых критериев, отчетов, справочников «сообщений»).

9. При создании сессии ПТС ОРИ должны предоставить ПУ на согласование перечень идентификаторов; подмножество согласованных ПУ идентификаторов определяет конкретные возможности взаимодействия ПУ и ПТС ОРИ.

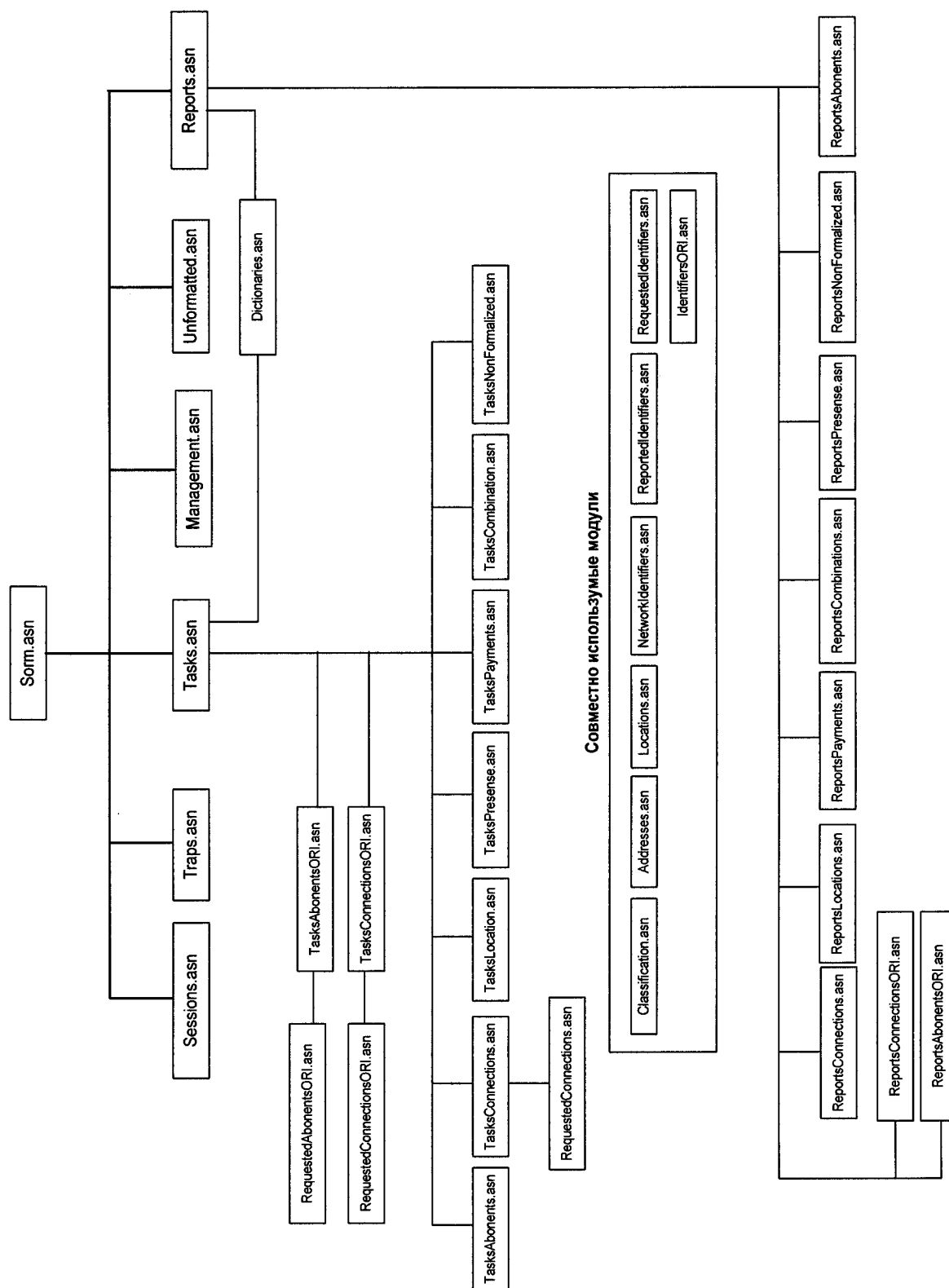


Схема 4

10. Расширение интерфейса взаимодействия ПУ с ПТС ОРИ должно обеспечиваться введением новых идентификаторов, определяющих соответствующие расширенные элементы (поисковые критерии, отчеты, справочники, «сообщения»). Кодирование новых вводимых идентификаторов элементов осуществляется в соответствии со структурами, представленными на схемах 2 и 3, и стандартным кодированием модуля «Classification.asn».

Приложение № 4
к Требованиям к оборудованию и программно-техническим средствам, используемым организатором распространения информации в сети «Интернет» в эксплуатируемых им информационных системах, для проведения уполномоченными государственными органами, осуществляющими оперативно-розыскную деятельность или обеспечение безопасности Российской Федерации, мероприятий в целях реализации возложенных на них задач, утвержденным приказом Министерства цифрового развития, связи и массовых коммуникаций Российской Федерации от 29.10.2018 2018 № 571

Описание протокола взаимодействия ПУ и ПТС ОРИ

Classification.asn

Classification DEFINITIONS IMPLICIT TAGS ::= BEGIN

EXPORTS TAGGED,

sorm-message-session,
sorm-message-trap,
sorm-message-task,
sorm-message-report,
sorm-message-management,
sorm-message-unformatted,
sorm-message-report-uni,

sorm-request-connection-ori-data-standart,
sorm-request-connection-ori-data-extended,
sorm-request-connection-ori-link,
sorm-request-connection-ori-aaa,
sorm-request-connection-ori-streams-content,
sorm-request-connection-ori-cert,
sorm-request-connection-ori-keys,

sorm-report-connection-ori-data,
sorm-report-connection-ori-link,
sorm-report-connection-ori-aaa,
sorm-report-connection-ori-streams-content,
sorm-report-connection-ori-cert,
sorm-report-connection-ori-keys,

sorm-request-abonent-ori,
sorm-report-abonent-ori,

sorm-request-dictionaries,
sorm-report-dictionary-telcos,
sorm-report-dictionary-ori-services,
sorm-report-dictionary-ori-events,
sorm-report-dictionary-ori-resources,
sorm-report-dictionary-ori-user-types,
sorm-report-dictionary-ori-payment-services,

sorm-request-presense,
sorm-report-presense-dictionaries,
sorm-report-presense-abonents-ori,
sorm-report-presense-connections-ori;

```
TAGGED ::= CLASS {
  &id ObjectDescriptor UNIQUE,
  &Data
}
WITH SYNTAX {
  OID &id
  DATA &Data
}
```

--- Классификация
OID ::= ObjectDescriptor

--- Подструктура сообщений
sorm-message-session OID ::= "280"
sorm-message-trap OID ::= "281"
sorm-message-task OID ::= "282"
sorm-message-report OID ::= "283"
sorm-message-management OID ::= "284"
sorm-message-unformatted OID ::= "285"
sorm-message-report-uni OID ::= "289"

--- Параметры соединений
sorm-request-connection-ori-data-standart OID ::= "265"
sorm-request-connection-ori-data-extended OID ::= "266"
sorm-request-connection-ori-link OID ::= "267"
sorm-request-connection-ori-aaa OID ::= "268"
sorm-request-connection-ori-streams-content OID ::= "269"
sorm-request-connection-ori-cert OID ::= "270"
sorm-request-connection-ori-keys OID ::= "271"

sorm-report-connection-ori-data OID ::= "10"
sorm-report-connection-ori-link OID ::= "11"
sorm-report-connection-ori-aaa OID ::= "12"
sorm-report-connection-ori-streams-content OID ::= "13"
sorm-report-connection-ori-cert OID ::= "14"
sorm-report-connection-ori-keys OID ::= "15"

--- Абоненты
sorm-request-abonent-ori OID ::= "245"
sorm-report-abonent-ori OID ::= "70"

--- Справочники
sorm-request-dictionaries OID ::= "240"

```
sorm-report-dictionary-telcos OID ::= "112"
sorm-report-dictionary-ori-services OID ::= "117"
sorm-report-dictionary-ori-events OID ::= "118"
sorm-report-dictionary-ori-resources OID ::= "119"
sorm-report-dictionary-ori-user-types OID ::= "130"
sorm-report-dictionary-ori-payment-services OID ::= "131"
```

```
--- Запрос о наличии данных
sorm-request-presense OID ::= "260"
```

```
sorm-report-presense-dictionaries OID ::= "123"
sorm-report-presense-abonents-ori OID ::= "125"
sorm-report-presense-connections-ori OID ::= "126"
```

```
END
```

Dictionaries.asn

```
Dictionaries DEFINITIONS IMPLICIT TAGS ::=
BEGIN
```

```
EXPORTS DictionaryTask,
       DictionaryReport;
```

```
IMPORTS DateAndTime
       FROM Sorm
```

```
       TAGGED,
       sorm-request-dictionaries,
       sorm-report-dictionary-telcos,
       sorm-report-dictionary-ori-services,
       sorm-report-dictionary-ori-events,
       sorm-report-dictionary-ori-resources,
       sorm-report-dictionary-ori-user-types,
       sorm-report-dictionary-ori-payment-services
       FROM Classification;
```

```
--- Запрос
DictionaryTask ::= SEQUENCE {
  id TAGGED.&id ({DictionaryTaskVariants}),
  data TAGGED.&Data ({DictionaryTaskVariants}){@id}
}
```

```
DictionaryTaskVariants TAGGED ::= {dictionaryTask}
```

```
dictionaryTask TAGGED ::= {
  OID {sorm-request-dictionaries}
  DATA ObjectDescriptor      --- тип запрашиваемого справочника (идентификатор отчёта)
}
```

```
--- DATA ObjectDescriptor принимает значение одно из:
--- sorm-report-dictionary-telcos
--- sorm-report-dictionary-ori-services
--- sorm-report-dictionary-ori-events
--- sorm-report-dictionary-ori-resources
--- sorm-report-dictionary-ori-user-types
--- sorm-report-dictionary-ori-payment-services
```

```
--- Отчёт
```

```

DictionaryReport ::= SEQUENCE {
  id TAGGED.&id ({DictionaryRecordsVariants}),    --- идентификтор записи справочника
  data TAGGED.&Data ({DictionaryRecordsVariants}@id) --- данные записи справочника
}

DictionaryRecordsVariants TAGGED ::= {
  telcosRecords          ---коммуникационные интернет-сервисы ИС ОРИ
| oriServices            --- виды информационных сервисов, предоставляемых ОРИ для пользователей
| oriEvents              --- виды событий, регистрируемых ИС ОРИ при взаимодействии ИС ОРИ с
пользователем
| oriResources           --- типы информационных ресурсов, создаваемых пользователями в ИС ОРИ
| oriUserTypes           --- типы пользователей, обслуживаемых ИС ОРИ
| oriPaymentServices     --- виды платежных услуг (платежных сервисов), используемых ОРИ
}

--- коммуникационные интернет-сервисы ИС ОРИ
telcosRecords TAGGED ::= {
  OID {sorm-report-dictionary-telcos}
  DATA SEQUENCE OF TelcosRecord
}

TelcosRecord ::= SEQUENCE {
  telco-id TelcoID,          --- идентификатор ИС ОРИ
  begin-time DateAndTime,    --- время начала действия
  end-time DateAndTime OPTIONAL, --- время конца действия
  description UTF8String (SIZE (1 .. 256)), --- описание (наименование) коммуникационного интернет-сервиса
  ИС ОРИ
  mcc [0] NumericString (SIZE (3)) OPTIONAL, --- код страны
  mnc [1] NumericString (SIZE (3)) OPTIONAL --- код ОРИ
}

--- виды информационных сервисов, предоставляемых ОРИ для пользователей
oriServices TAGGED ::= {
  OID {sorm-report-dictionary-ori-services}
  DATA SEQUENCE OF OriServiceRecord
}

OriServiceRecord ::= SEQUENCE {
  telco-id TelcoID,          --- идентификатор ИС ОРИ
  service-id INTEGER (0 .. 65535), --- идентификатор сервиса
  description UTF8String (SIZE (1 .. 256)), --- описание
  begin-time DateAndTime,    --- время начала действия
  end-time DateAndTime OPTIONAL --- время конца действия
}

--- виды событий, регистрируемых ИС ОРИ при взаимодействии ИС ОРИ с пользователем
oriEvents TAGGED ::= {
  OID {sorm-report-dictionary-ori-events}
  DATA SEQUENCE OF OriEventRecord
}

OriEventRecord ::= SEQUENCE {
  telco-id TelcoID,          --- идентификатор ИС ОРИ
  service-id INTEGER (0 .. 65535), --- идентификатор сервиса
  event-id INTEGER (0 .. 65535), --- идентификатор события
  description UTF8String (SIZE (1 .. 256)), --- описание
  begin-time DateAndTime,    --- время начала действия
  end-time DateAndTime OPTIONAL --- время конца действия
}

```


--- типы информационных ресурсов, создаваемых пользователями в ИС ОРИ

```
oriResources TAGGED ::= {
  OID {sorm-report-dictionary-ori-resources}
  DATA SEQUENCE OF OriResourceRecord
}
```

```
OriResourceRecord ::= SEQUENCE {
  telco-id TelcoID,          --- идентификатор ИС ОРИ
  service-id INTEGER (0 .. 65535), --- идентификатор сервиса
  resource-id INTEGER (0 .. 65535), --- идентификатор ресурса
  description UTF8String (SIZE (1 .. 256)), --- описание
  begin-time DateAndTime,    --- время начала действия
  end-time DateAndTime OPTIONAL --- время конца действия
}
```

--- типы пользователей, обслуживаемых ИС ОРИ

```
oriUserTypes TAGGED ::= {
  OID {sorm-report-dictionary-ori-user-types}
  DATA SEQUENCE OF OriUserTypeRecord
}
```

```
OriUserTypeRecord ::= SEQUENCE {
  telco-id TelcoID,          --- идентификатор ИС ОРИ
  service-id INTEGER (0 .. 65535), --- идентификатор сервиса
  user-type-id INTEGER (0 .. 65535), --- тип пользователя
  description UTF8String (SIZE (1 .. 256)), --- описание
  begin-time DateAndTime,    --- время начала действия
  end-time DateAndTime OPTIONAL --- время конца действия
}
```

--- виды платежных услуг (платежных сервисов), используемых ОРИ

```
oriPaymentServices TAGGED ::= {
  OID {sorm-report-dictionary-ori-payment-services}
  DATA SEQUENCE OF OriPaymentServiceRecord
}
```

```
OriPaymentServiceRecord ::= SEQUENCE {
  telco-id TelcoID,          --- идентификатор ИС ОРИ
  service-id INTEGER (0 .. 65535), --- идентификатор сервиса
  payment-service-id INTEGER (0 .. 65535), --- идентификатор платежной услуги
  description UTF8String (SIZE (1 .. 256)), --- описание
  begin-time DateAndTime,    --- время начала действия
  end-time DateAndTime OPTIONAL --- время конца действия
}
```

END

IdentifiersORI.asn

```
IdentifiersORI DEFINITIONS IMPLICIT TAGS ::=
BEGIN
```

```
EXPORTS UserID,
  UserIdentifier,
  UserTechnicalIdentifier,
  ResourceIdentifier,
  ResourceName,
  ORISreamID;
```

```

UserIdentifier ::= SEQUENCE {
  user-id [0] UserID,          --- идентификатор пользователя в системе (имя пользователя)
  user-type [1] INTEGER (0 .. 65535) --- тип пользователя (расшифровывается по справочнику)
}

UserTechnicalIdentifier ::= CHOICE {
  ip-address [0] UserIPAddress,      --- ip-адрес и порт
  msisdn [1] UTF8String (SIZE (1 .. 64)), --- номер телефона
  email [2] UTF8String (SIZE (1 .. 256)), --- электронная почта
  program-name [3] UTF8String (SIZE (1 .. 1024)), --- имя программы клиента
  other [4] UTF8String (SIZE (1 .. 1024)) --- прочая техническая информация
}

UserIPAddress ::= SEQUENCE {
  ip-address [0] UTF8String (SIZE (1 .. 64)), --- ip-адрес
  ip-port [1] INTEGER (0 .. 65535) OPTIONAL --- TCP/UDP порт
}

UserID ::= UTF8String (SIZE (1 .. 128))

ResourceIdentifier ::= SEQUENCE {
  resource-name [0] ResourceName,      --- наименование ресурса
  resource-type [1] INTEGER (0 .. 65535) OPTIONAL --- тип ресурса (расшифровывается по справочнику)
}

ResourceName ::= UTF8String (SIZE (1 .. 4096))

ORISStreamID ::= OCTET STRING (SIZE (8))

END

```

Locations.asn

```

Locations DEFINITIONS IMPLICIT TAGS ::=
BEGIN

EXPORTS LocationInfo;

IMPORTS ProjectionType
  FROM NetworkIdentifiers;

MobileORILocation ::= SEQUENCE {
  mcc [0] UTF8String (SIZE (1 .. 4)) OPTIONAL, --- код страны оператора связи пользователя ОРИ
  mnc [1] UTF8String (SIZE (1 .. 3)) OPTIONAL, --- код оператора связи пользователя ОРИ
  lac [2] INTEGER (0 .. 65535) OPTIONAL, --- код LAC оператора связи пользователя ОРИ
  cell [3] INTEGER (0 .. 4294967295) OPTIONAL, --- код БС оператора связи пользователя ОРИ
  network-operator [4] UTF8String (SIZE (1 .. 64)) OPTIONAL --- наименование оператора связи пользователя ОРИ
}

GeoLocation ::= SEQUENCE {
  latitude-grade REAL, --- широта
  longitude-grade REAL, --- долгота
  projection-type ProjectionType --- тип проекции координат
}

LocationInfo ::= SEQUENCE {
  mobile-ori-location [0] MobileORILocation OPTIONAL,

```

```

geo-location [1]   GeoLocation OPTIONAL,
description [2]   UTF8String (SIZE (1 .. 4096)) OPTIONAL
}

END

```

Management.asn

```

Management DEFINITIONS IMPLICIT TAGS ::=
BEGIN

```

```

EXPORTS managementMessage;

```

```

IMPORTS TAGGED,
    sorm-message-management
FROM Classification;

```

```

managementMessage TAGGED ::= {
    OID {sorm-message-management}
    DATA CHOICE {
        request [0] ManagementRequest,
        response [1] ManagementResponse
    }
}

```

```

--- тип сообщения "команда управления ИС СОРМ"

```

```

ManagementRequest ::= CHOICE {
    get-structure [0] GetStructureRequest, --- запрос на получение структуры ИС СОРМ - КТС и модулей СПО
    get-module-config [1] GetModuleConfigRequest, --- запрос на получение конфигурации КТС / модуля СПО
    set-module-config [2] SetModuleConfigRequest, --- запрос на изменение конфигурации КТС / модуля СПО
    check-module [3] CheckModuleRequest, --- запрос на получение состояния модуля
    get-module-types [4] GetModuleTypesRequest --- запрос на получение типов модулей КТС и СПО
}

```

```

--- запрос на получение структуры ИС СОРМ - КТС и модулей СПО

```

```

GetStructureRequest ::= NULL

```

```

--- запрос на получение конфигурации КТС / модуля СПО

```

```

GetModuleConfigRequest ::= CHOICE {
    hw-modules-list [0] RequestedHardwareModules, --- перечень идентификаторов узлов КТС ИС СОРМ
    sw-modules-list [1] RequestedSoftwareModules --- перечень идентификаторов модулей СПО ИС СОРМ
}

```

```

RequestedHardwareModules ::= SEQUENCE OF ModuleId

```

```

RequestedSoftwareModules ::= SEQUENCE OF ModuleId

```

```

--- запрос на изменение конфигурации КТС / модуля СПО

```

```

SetModuleConfigRequest ::= SEQUENCE {
    module-id ModuleId, --- идентификатор конфигулируемого модуля
    module-config ConfiguredModule --- устанавливаемая в модуль конфигурация
}

```

```

ConfiguredModule ::= CHOICE {
    sw-module [0] SormSoftwareModule, --- для модуля СПО
    hw-module [1] SormHardwareModule --- для узла КТС
}

```

```

--- запрос на получение состояния модуля

```

CheckModuleRequest ::= RequestedModulesList

```
RequestedModulesList ::= CHOICE {
  hw-modules [0] RequestedHardwareModules, --- идентификаторы узлов КТС, для которых запрашивается
  состояние
  sw-modules [1] RequestedSoftwareModules --- идентификаторы модулей СПО, для которых запрашивается
  состояние
}
```

--- запрос на получение типов модулей КТС и СПО

GetModuleTypesRequest ::= NULL

--- тип сообщения "ответ на команду управления ИС СОРМ"

```
ManagementResponse ::= CHOICE {
  get-structure [0] GetStructureResponse, --- ответ на запрос получения структуры ИС СОРМ - КТС и модулей
  СПО
  get-module-config [1] GetModuleConfigResponse, --- ответ на запрос получения конфигурации КТС / модуля
  СПО
  set-module-config [2] SetModuleConfigResponse, --- ответ на запрос изменения конфигурации КТС / модуля
  СПО
  check-module [3] CheckModuleResponse, --- ответ на запрос получения состояния модуля
  get-module-types [4] GetModuleTypesResponse --- ответ на запрос получения типов модулей КТС и СПО
}
```

--- ответ на запрос получения структуры ИС СОРМ - КТС и модулей СПО

```
GetStructureResponse ::= SEQUENCE {
  hw-modules SormHardwareModules, --- перечень всех узлов КТС
  sw-modules SormSoftwareModules --- перечень всех модулей СПО
}
```

--- ответ на запрос получения конфигурации КТС / модуля СПО

```
GetModuleConfigResponse ::= SEQUENCE {
  hw-modules SormHardwareModules, --- конфигурации запрошенных узлов КТС
  sw-modules SormSoftwareModules --- конфигурации запрошенных модулей СПО ИС СОРМ
}
```

--- отчет на запрос изменения конфигурации КТС / модуля СПО

SetModuleConfigResponse ::= ConfiguredModule --- установленная в модуль конфигурация

--- ответ на запрос получения состояния модуля

```
CheckModuleResponse ::= CHOICE {
  hw-modules [0] SormHardwareModules, --- текущее состояние запрошенных узлов КТС
  sw-modules [1] SormSoftwareModules --- текущее состояние запрошенных модулей СПО ИС СОРМ
}
```

--- ответ на запрос получения типов модулей КТС и СПО

GetModuleTypesResponse ::= SEQUENCE OF ModuleType

```
ModuleType ::= SEQUENCE {
  module-type INTEGER (1 .. 512), --- идентификатор типа модуля
  type-description UTF8String (SIZE (1 .. 128)) --- расшифровка типа модуля
}
```

SormHardwareModules ::= SEQUENCE OF SormHardwareModule

```
SormHardwareModule ::= SEQUENCE {
  module-id ModuleId, --- уникальный идентификатор данного модуля
  block-name INTEGER (0 .. 1024), --- номер блока КТС
  module-name UTF8String (SIZE (1 .. 512)), --- наименование модуля
}
```

```

module-parameters HwParameterGroups      --- значение группы параметров КТС
}

HwParameterGroups ::= SEQUENCE OF HwParameterGroup

HwParameterGroup ::= SEQUENCE {
  group-name      UTF8String (SIZE (1 .. 512)), --- наименование группы параметров для КТС
  module-parameters ModuleParameters        --- перечень параметров для КТС
}

SormSoftwareModules ::= SEQUENCE OF SormSoftwareModule

SormSoftwareModule ::= SEQUENCE {
  module-id      ModuleId,      --- уникальный идентификатор данного модуля
  hardware-module-id ModuleId,  --- идентификатор КТС, на котором работает данный блок модуля
  СПО
  block-name     INTEGER (0 .. 1024), --- номер блока СПО модуля
  module-name    UTF8String (SIZE (1 .. 512)), --- наименование модуля
  module-type    INTEGER (1 .. 512), --- идентификатор типа модуля
  module-parameters ModuleParameters, --- список параметров модуля
  sub-modules-list SormSoftwareModules OPTIONAL --- submodule
}

ModuleParameters ::= SEQUENCE OF ModuleParameter

ModuleParameter ::= SEQUENCE {
  parameter-name UTF8String (SIZE (1 .. 256)), --- наименование параметра
  read-only     BOOLEAN, --- контролируемый или измеряемый параметр
  parameter-value ParameterValue --- значение параметра
}

ParameterValue ::= CHOICE {
  string [0] UTF8String (SIZE (1 .. 256)),
  integer [1] INTEGER (0 .. 999999999),
  boolean [2] BOOLEAN
}

--- уникальный идентификатор КТС / модуля СПО ИС СОРМ
ModuleId ::= OCTET STRING (SIZE (8))

END

```

NetworkIdentifiers.asn

```

NetworkIdentifiers DEFINITIONS IMPLICIT TAGS ::=
BEGIN

EXPORTS NetworkPeerInfo,
  ProjectionType;

--- информация об участнике соединения передачи данных
NetworkPeerInfo ::= SEQUENCE {
  ip-address IPAddress, --- IP-адрес
  ip-port   IPPort OPTIONAL --- IP-порт
}

--- IP-адрес
IPAddress ::= CHOICE {
  ipv4 [0] IPV4Address, --- IPv4-адрес
  ipv6 [1] IPV6Address --- IPv6-адрес
}

```

```

}

--- IPv4-адрес
IPV4Address ::= OCTET STRING (SIZE (4))

--- IPv6-адрес
IPV6Address ::= OCTET STRING (SIZE (16))

--- IP/UDP/TCP-порт
IPPort ::= OCTET STRING (SIZE (2)) --- порт

--- тип проекции координат
ProjectionType ::= ENUMERATED {
  wgs84 (0),
  utm (1),
  sgs85 (2)
}

END

```

Reports.asn

```

Reports DEFINITIONS IMPLICIT TAGS ::=
BEGIN

EXPORTS reportMessage,
  Acknowledgement;

IMPORTS TAGGED,
  sorm-message-report
  FROM Classification

  MessageID
  FROM Sorm

  TaskID
  FROM Tasks

  DictionaryReport
  FROM Dictionaries

  PresenseReport
  FROM ReportsPresense

  AbonentsORIReport
  FROM ReportsAbonentsORI

  ConnectionsORIReport
  FROM ReportsConnectionsORI;

reportMessage TAGGED ::= {
  OID {sorm-message-report}
  DATA CHOICE {
    report [0] Report,      --- тип сообщения "отчет"
    ack [1] Acknowledgement --- тип сообщения "подтверждение"
  }
}

--- Блок данных сообщения типа "отчет"

```

```

Report ::= SEQUENCE {
  request-id MessageID,      --- идентификатор запроса, запросивший отчёт
  task-id TaskID,           --- идентификатор задачи, сгенерировавшей данный отчет
  total-blocks-number INTEGER, --- общее количество блоков в отчете
  block-number INTEGER,     --- порядковый номер текущего блока
  report-block ReportDataBlock --- блок данных отчета
}

ReportDataBlock ::= CHOICE {
  dictionary [0] DictionaryReport, --- отчеты задач пополнения справочников (нормативно-справочная информация)
  presense [6] PresenseReport, --- отчеты задач по запросу наличия в ИС СОРМ информации
  abonents-ori [8] AbonentsORIReport, --- отчеты задач по принадлежности абонентов организаторов распространения информации
  connections-ori [9] ConnectionsORIReport --- отчеты задач по соединениям абонентов организаторов распространения информации
}

--- Подтверждение приёма блока, передаётся с номером сообщения соответствующему номеру сообщения блока отчёта
Acknowledgement ::= SEQUENCE {
  successful BOOLEAN, --- признак успешного приёма блока
  broken-record INTEGER OPTIONAL, --- номер записи в отчете, обработанной на ПУ с ошибкой
  error-description UTF8String (SIZE (1 .. 1024)) OPTIONAL --- описание ошибки приёма блока в произвольной форме
}

END

```

ReportsAbonentsORI.asn

```

ReportsAbonentsORI DEFINITIONS IMPLICIT TAGS ::=
BEGIN

EXPORTS AbonentsORIReport;

IMPORTS TAGGED,
  sorm-report-abonent-ori
  FROM Classification

  DateAndTime
  FROM Sorm

  TelcoID
  FROM Tasks

  UserID,
  UserIdentifier
  FROM IdentifiersORI;

AbonentsORIReport ::= SEQUENCE {
  id TAGGED.&id {{AbonentsORIReportVariants}},
  data TAGGED.&Data {{AbonentsORIReportVariants}}{@id}
}

AbonentsORIReportVariants TAGGED ::= {
  reportAbonentORI
}

reportAbonentORI TAGGED ::= {

```

```

OID {sorm-report-abonent-ori}
DATA SEQUENCE OF AbonentsORISRecord
}

```

```

AbonentsORISRecord ::= SEQUENCE {
telco-id          TelcoID,                --- идентификатор ИС ОРИ, обслуживаемый ПТС ОРИ
user-identifier   UserIdentifier,         --- идентификатор пользователя
datetime-registered DateAndTime,         --- дата и время регистрации
abonent-info      AbonentInfoORI,        --- информация о пользователе
abonent-contacts [0] AbonentContactsORI OPTIONAL, --- контактные данные пользователя
im-identifiers [1] SEQUENCE OF AbonentImIdentifierORI OPTIONAL, --- идентификаторы пользователя в других
средствах электронного взаимодействия
datetime-updated [2] DateAndTime OPTIONAL, --- дата и время обновления информации
datetime-unregistered [3] DateAndTime OPTIONAL, --- дата и время прекращения регистрации
contract-date [4] DateAndTime OPTIONAL, --- дата и время заключения договора
contract [5] UTF8String (SIZE (1 .. 64)) OPTIONAL, --- номер договора
additional [6] SEQUENCE OF AdditionalInfo OPTIONAL --- дополнительная информация о
пользователе
}

```

--- информация о пользователе

```

AbonentInfoORI ::= SEQUENCE {
nick-name [0] UTF8String (SIZE (1 .. 1024)) OPTIONAL, --- псевдоним пользователя
birth-date GeneralizedTime OPTIONAL, --- дата рождения
address [1] ReportedAddressORI OPTIONAL, --- адресные данные
name-info [2] ReportedNameInfoORI OPTIONAL, --- ФИО
passport-info [3] ReportedPassportInfoORI OPTIONAL, --- паспортные данные
langs [4] SEQUENCE OF UTF8String (SIZE (1 .. 64)) OPTIONAL, --- список языков, которыми владеет
пользователь
relatives [5] SEQUENCE OF UserID OPTIONAL --- список родственников пользователя
}

```

--- адресные данные

```

ReportedAddressORI ::= CHOICE {
struct-info [0] ReportedStructAddressORI,
unstruct-info [1] UTF8String (SIZE (1 .. 1024))
}

```

```

ReportedStructAddressORI ::= SEQUENCE {
country [0] UTF8String (SIZE (1 .. 128)) OPTIONAL, --- страна
region [1] UTF8String (SIZE (1 .. 128)) OPTIONAL, --- область
city [2] UTF8String (SIZE (1 .. 128)) OPTIONAL, --- город, поселок, деревня
street [3] UTF8String (SIZE (1 .. 128)) OPTIONAL, --- улица
building [4] UTF8String (SIZE (1 .. 128)) OPTIONAL, --- дом, строение
apartment [5] UTF8String (SIZE (1 .. 128)) OPTIONAL --- квартира, офис
}

```

--- ФИО

```

ReportedNameInfoORI ::= CHOICE {
struct-info [0] ReportedStructNameInfoORI,
unstruct-info [1] UTF8String (SIZE (1 .. 1024))
}

```

```

ReportedStructNameInfoORI ::= SEQUENCE {
given-name [0] UTF8String (SIZE (1 .. 128)) OPTIONAL, --- имя
initial [1] UTF8String (SIZE (1 .. 128)) OPTIONAL, --- отчество
family-name [2] UTF8String (SIZE (1 .. 128)) OPTIONAL --- фамилия
}

```



```

--- паспортные данные
ReportedPassportInfoORI ::= CHOICE {
  struct-info [0] ReportedStructPassroptInfoORI,
  unstruct-info [1] UTF8String(SIZE (1 .. 1024))
}

ReportedStructPassroptInfoORI ::= SEQUENCE {
  passport-serial [0] UTF8String (SIZE (1..16)) OPTIONAL, --- серия паспорта
  passport-number [1] UTF8String (SIZE (1..16)) OPTIONAL --- номер паспорта
}

--- контактные данные пользователя
AbonentContactsORI ::= SEQUENCE {
  msisdn [0] SEQUENCE OF UTF8String (SIZE (1 .. 64)) OPTIONAL, --- список номеров телефона пользователя
  email [1] SEQUENCE OF UTF8String (SIZE (1 .. 256)) OPTIONAL --- список адресов электронной почты
  пользователя
}

--- идентификаторы пользователя в других средствах электронного взаимодействия
AbonentImIdentifierORI ::= SEQUENCE {
  service-name UTF8String (SIZE (1 .. 128)), --- наименование сервиса
  service-id UTF8String (SIZE (1 .. 256)) --- идентификатор пользователя в сервисе
}

--- дополнительная информация о пользователе
AdditionalInfo ::= SEQUENCE {
  title UTF8String (SIZE (1 .. 128)), --- наименование о дополнительных сведениях
  content UTF8String (SIZE (1 .. 4096)) --- содержание дополнительных сведений
}
}
END

```

ReportsConnectionsORI.asn

```

ReportsConnectionsORI DEFINITIONS IMPLICIT TAGS ::=
BEGIN

```

```

EXPORTS ConnectionsORIReport,
  reportConnectionsDataORI,
  reportConnectionsLinkORI,
  reportConnectionsAAAORI,
  reportConnectionsStreamsContentORI,
  reportConnectionsCertORI,
  reportConnectionsKeysORI;

```

```

IMPORTS TAGGED,
  sorm-report-connection-ori-data,
  sorm-report-connection-ori-link,
  sorm-report-connection-ori-aaa,
  sorm-report-connection-ori-streams-content,
  sorm-report-connection-ori-cert,
  sorm-report-connection-ori-keys
FROM Classification

```

```

DateAndTime
FROM Sorm

```

```

TelcoID
FROM Tasks

```

```

UserID,
UserIdentifier,

```

```
UserTechnicalIdentifier,
ResourceIdentifier,
ORISStreamID
FROM IdentifiersORI
```

```
NetworkPeerInfo
FROM NetworkIdentifiers
```

```
LocationInfo
FROM Locations;
```

```
ConnectionsORIReport ::= SEQUENCE {
  id TAGGED.&id {{ReportedConnectionsORIVariants}},
  data TAGGED.&Data {{ReportedConnectionsORIVariants}{@id}}
}
```

```
ReportedConnectionsORIVariants TAGGED ::= {
  reportConnectionsDataORI      --- события информационного взаимодействия пользователей в ИС ОПИ
  | reportConnectionsLinkORI     --- события добавления/исключения связанных пользователей
  | reportConnectionsAAAORI     --- события регистрации, прекращения регистрации, авторизации, выхода
  из информационного сервиса
  | reportConnectionsStreamsContentORI --- содержимое изображений, звуков, голосовой информации, видео-
  и иных электронных сообщений
  | reportConnectionsCertORI
  | reportConnectionsKeysORI
}
```

--- события информационного взаимодействия пользователей в ИС ОПИ

```
reportConnectionsDataORI TAGGED ::= {
  OID {sorm-report-connection-ori-data}
  DATA SEQUENCE OF ConnectionDataORIRecord
}
```

```
ConnectionDataORIRecord ::= SEQUENCE {
  telco-id          TelcoID,          --- идентификатор ИС ОПИ, обслуживаемый ПТС ОПИ
  arrive-datetime  DateAndTime,      --- дата и время поступления информации
  datetime         DateAndTime,      --- дата и время
  service-id      INTEGER (0 .. 65535), --- идентификатор сервиса (расшифровывается по
  справочнику)
  abonent-identifier UserIdentifier,  --- идентификатор пользователя
  abonent-technical-identifier SEQUENCE OF UserTechnicalIdentifier, --- технические данные, идентифицирующие
  пользователя
  event-id        INTEGER (0 .. 65535), --- тип события (расшифровывается по справочнику)
  contacts-identifier SEQUENCE OF UserIdentifier, --- идентификаторы контактов
  abonent-location [0] LocationInfo OPTIONAL, --- местоположение пользователя
  resource-identifier [1] ResourceIdentifier OPTIONAL, --- ресурс
  resource-info [2] ResourceMetadataInfoORI OPTIONAL, --- техническая информация о ресурсе
  owner-identifier [3] UserIdentifier OPTIONAL, --- идентификатор владельца ресурса
  message-text [4] UTF8String OPTIONAL, --- текст сообщения без разметки и иной служебной
  коммуникационной информации
  payment-id [5] PaymentInfoORI OPTIONAL, --- техническая информация о платеже
  stream-id [6] ORISStreamID OPTIONAL --- последовательный идентификатор файла
}
```

```
ResourceMetadataInfoORI ::= CHOICE {
  files-metadata [0] SEQUENCE OF FileMetadata,
  stream-metadata [1] StreamMetadata
}
```

--- техническая информация о файловых данных

```
FileMetadata ::= SEQUENCE {
  filename UTF8String (SIZE (1 .. 256)),
  filesize INTEGER
}
```

--- техническая информация о потоковых данных

```
StreamMetadata ::= SEQUENCE {
  timestamp [0] GeneralizedTime OPTIONAL,
  duration [1] INTEGER OPTIONAL,
  start-offset [2] INTEGER OPTIONAL,
  stop-offset [3] INTEGER OPTIONAL
}
```

```
PaymentInfoORI ::= SEQUENCE {
  payment-identifier UTF8String (SIZE (1 .. 128)), --- идентификатор транзакции, счета, кошелька и т.д.
  payment-service-id INTEGER (0 .. 65535), --- идентификатор платежного сервиса (расшифровывается
по справочнику)
  payment-info UTF8String (SIZE (1 .. 4096)) OPTIONAL --- дополнительная информация
}
```

--- события добавления/исключения связанных пользователей

```
reportConnectionsLinkORI TAGGED ::= {
  OID {sorm-report-connection-ori-link}
  DATA SEQUENCE OF ConnectionLinkORIRcord
}
```

```
ConnectionLinkORIRcord ::= SEQUENCE {
  telco-id TelcoID, --- идентификатор ИС ОПИ, обслуживаемый ПТС ОПИ
  arrive-datetime DateAndTime, --- дата и время поступления информации
  datetime DateAndTime, --- дата и время
  service-id INTEGER (0 .. 65535), --- идентификатор сервиса (расшифровывается по справочнику)
  abonent-id UserIdentifier, --- идентификатор пользователя, устанавливающего связь
  event-id INTEGER (0 .. 65535), --- тип события (расшифровывается по справочнику)
  linked-identifier LinkedIdentifier --- идентификатор объекта, с которым установлена связь
}
```

```
LinkedIdentifier ::= CHOICE {
  connected-user-identifier [0] UserIdentifier, --- идентификатор пользователя
  connected-resource-identifier [1] ResourceIdentifier --- ресурс
}
```

--- события регистрации, прекращения регистрации, авторизации, выхода из информационного сервиса

```
reportConnectionsAAAORI TAGGED ::= {
  OID {sorm-report-connection-ori-aaa}
  DATA SEQUENCE OF ConnectionAAAORIRcord
}
```

```
ConnectionAAAORIRcord ::= SEQUENCE {
  telco-id TelcoID, --- идентификатор ИС ОПИ, обслуживаемый ПТС ОПИ
  arrive-datetime DateAndTime, --- дата и время поступления информации
  datetime DateAndTime, --- дата и время
  service-id INTEGER (0 .. 65535), --- идентификатор сервиса (расшифровывается по
справочнику)
  abonent-identifier UserIdentifier, --- идентификатор пользователя
  abonent-technical-identifier SEQUENCE OF UserTechnicalIdentifier, --- технические данные, идентифицирующие
пользователя
  event-id INTEGER (0 .. 65535), --- тип события (расшифровывается по справочнику)
  abonent-location [0] LocationInfo OPTIONAL --- местоположение пользователя
}
```

```

}

--- содержимое изображений, звуков, голосовой информации, видео- и иных электронных сообщений
reportConnectionsStreamsContentORI TAGGED ::= {
  OID {sorm-report-connection-ori-streams-content}
  DATA ConnectionsStreamsContentORIRecords
}

ConnectionsStreamsContentORIRecords ::= CHOICE {
  flow-data [0] SEQUENCE OF ConnectionsStreamsContentORIRecordFlow,
  file-data [1] SEQUENCE OF ConnectionsStreamsContentORIRecordFile
}

ConnectionsStreamsContentORIRecordFlow ::= SEQUENCE {
  data OCTET STRING (SIZE (1 .. 1048576)), --- содержимое блока
  codec UTF8String (SIZE (1 .. 4096)) OPTIONAL, --- описание способа кодирования в формате SDP в
соответствии RFC 2327
  direction StreamsContentORIDirection OPTIONAL --- направление передачи
}

StreamsContentORIDirection ::= ENUMERATED {
  client-server (0),
  server-client (1)
}

ConnectionsStreamsContentORIRecordFile ::= OCTET STRING (SIZE (1 .. 1048576))

reportConnectionsCertORI TAGGED ::= {
  OID {sorm-report-connection-ori-cert}
  DATA SEQUENCE OF ConnectionCertORIRecord
}

ConnectionCertORIRecord ::= SEQUENCE {
  telco-id TelcoID,
  arrive-datetime DateAndTime,
  datetime-from DateAndTime,
  datetime-to DateAndTime OPTIONAL,
  client NetworkPeerInfo,
  server NetworkPeerInfo,
  client-random OCTET STRING (SIZE (1 .. 512)),
  pre-master-key OCTET STRING (SIZE (1 .. 512)),
  server-random [0] OCTET STRING (SIZE (1 .. 512)) OPTIONAL
}

reportConnectionsKeysORI TAGGED ::= {
  OID {sorm-report-connection-ori-keys}
  DATA SEQUENCE OF ConnectionKeysORIRecord
}

ConnectionKeysORIRecord ::= SEQUENCE {
  telco-id TelcoID,
  arrive-datetime DateAndTime,
  datetime-from DateAndTime,
  datetime-to [0] DateAndTime OPTIONAL,
  as-server [1] INTEGER OPTIONAL,
  as-client [2] INTEGER OPTIONAL,
  client NetworkPeerInfo,
  server NetworkPeerInfo,
  client-random OCTET STRING (SIZE (1 .. 512)),

```

```

pre-master-key OCTET STRING (SIZE (1 .. 512)),
server-random [0] OCTET STRING (SIZE (1 .. 512)) OPTIONAL,
sni [1] UTF8String (SIZE (1 .. 128)) OPTIONAL,
user-id [2] UTF8String (SIZE (1 .. 64)) OPTIONAL,
user-login [3] UTF8String (SIZE (1 .. 64)) OPTIONAL,
key-exchange [4] UTF8String (SIZE (1 .. 16)) OPTIONAL,
cipher-suite [5] INTEGER OPTIONAL,
client-region [6] UTF8String (SIZE (1 .. 10)) OPTIONAL,
server-region [7] UTF8String (SIZE (1 .. 10)) OPTIONAL,
service-name [8] UTF8String (SIZE (1 .. 16)) OPTIONAL
}

END

```

ReportsPresense.asn

```

ReportsPresense DEFINITIONS IMPLICIT TAGS ::=
BEGIN

EXPORTS PresenseReport;

IMPORTS TAGGED,
    sorm-report-presense-dictionaries,
    sorm-report-presense-abonents-ori,
    sorm-report-presense-connections-ori
    FROM Classification

    FindRange
    FROM Sorm

    TelcoID
    FROM Tasks;

--- отчет по запросу наличия информации
PresenseReport ::= SEQUENCE {
    id TAGGED.&id ({ReportedPresensesVariants}),
    data TAGGED.&Data ({ReportedPresensesVariants}{@id})
}

ReportedPresensesVariants TAGGED ::= {
    dictionariesPresence --- отчет по наличию информации по справочникам
    | abonentsORIPresence --- отчет по наличию информации по информации по пользователям ОРИ и их
идентификаторам
    | connectionsORIPresence --- отчет по наличию информации по соединениям пользователя ОРИ
}

--- отчет по наличию информации по справочникам.
--- Если какой-либо из справочников не публикуется ИС СОПМ, запись о нем отсутствует
dictionariesPresence TAGGED ::= {
    OID {sorm-report-presense-dictionaries}
    DATA SEQUENCE OF DictionaryInfo
}

--- запись отчета о наличии справочной информации
DictionaryInfo ::= SEQUENCE {
    telco-id TelcoID, --- идентификатор ИС ОРИ
    dict ObjectDescriptor, --- тип справочника, по которому есть информация
    count INTEGER (1 .. 4294967295), --- количество записей в справочнике
    change-dates FindRange --- минимальное и максимальное дата/время изменения записей в
справочнике
}

```

--- отчет по наличию информации по информации по пользователям ОРИ и их идентификаторам

```
abonentsORIPresence TAGGED ::= {
  OID {sorm-report-presense-abonents-ori}
  DATA SEQUENCE OF AbonentsORIPresenceRecord
}
```

```
AbonentsORIPresenceRecord ::= SEQUENCE {
  range FindRange, --- интервал времени, на который имеются данные в ИС ОРИ
  count INTEGER, --- количество записей
  service-id INTEGER (0 .. 65535) OPTIONAL --- идентификатор сервиса
}
```

--- отчет по наличию информации по соединениям пользователя ОРИ

```
connectionsORIPresence TAGGED ::= {
  OID {sorm-report-presense-connections-ori}
  DATA SEQUENCE OF ConnectionsORIPresenceRecord
}
```

```
ConnectionsORIPresenceRecord ::= SEQUENCE {
  range FindRange, --- интервал времени, на который имеются данные в ИС ОРИ
  count INTEGER, --- количество записей
  data-type ConnectionsORIPresenceType, --- вид событий в ИС ОРИ, информация по которым есть в ИС СОПМ
  service-id INTEGER (0 .. 65535) OPTIONAL --- идентификатор сервиса
}
```

```
ConnectionsORIPresenceType ::= ENUMERATED {
  data (0), --- записи об информационном взаимодействии пользователей в ИС ОРИ
  link (1), --- записи о добавлении/исключении связанных пользователей
  aaa (2), --- записи о регистрации, прекращения регистрации, авторизации, выхода из информационного
  сервиса
  certs (3),
  keys (4)
}
```

END

ReportsUNI.asn

```
ReportsUNI DEFINITIONS IMPLICIT TAGS ::=
BEGIN
```

```
EXPORTS reportMessageUNI;
```

```
IMPORTS TAGGED,
```

```
  sorm-message-report-uni
  FROM Classification
```

```
  ObjectUNI
  FROM Tasks
```

```
  Acknowledgement
  FROM Reports
```

```
  reportConnectionsDataORI,
  reportConnectionsLinkORI,
  reportConnectionsAAAORI,
  reportConnectionsStreamsContentORI
  FROM ReportsConnectionsORI;
```

```

reportMessageUNI TAGGED ::= {
  OID {sorm-message-report-uni}
  DATA CHOICE {
    report [0] ReportUNI,    --- тип сообщения "отчет"
    ack [1] Acknowledgement --- тип сообщения "подтверждение"
  }
}

--- Блок данных сообщения типа "отчет"
ReportUNI ::= SEQUENCE {
  object-unis SEQUENCE OF ObjectUNI, --- идентификаторы объектов наблюдения, по которым отобраны
данные
  report-block ReportBlockUNI      --- блок данных отчета
}

ReportBlockUNI ::= CHOICE {
  statistics [0] StatisticsReportedConnections,
  content [1] ContentReportBlockUNI
}

StatisticsReportedConnections ::= SEQUENCE {
  id TAGGED.&id {{StatisticsReportedConnectionsVariants}},
  data TAGGED.&Data {{StatisticsReportedConnectionsVariants}{@id}}
}

StatisticsReportedConnectionsVariants TAGGED ::= {
  reportConnectionsDataORI      --- события информационного взаимодействия пользователей в ИС ОРИ
| reportConnectionsLinkORI      --- события добавления/исключения связанных пользователей
| reportConnectionsAAAORI      --- события регистрации, прекращения регистрации, авторизации, выхода
из информационного сервиса
}

ContentReportBlockUNI ::= SEQUENCE {
  report-block ContentReportedConnections, --- блок данных отчета
  stream-id ORISStreamID,      --- идентификатор содержимого
  last-block BOOLEAN          --- признак последнего блока
}

ContentReportedConnections ::= SEQUENCE {
  id TAGGED.&id {{ContentReportedConnectionsVariants}},
  data TAGGED.&Data {{ContentReportedConnectionsVariants}{@id}}
}

ContentReportedConnectionsVariants TAGGED ::= {
  reportConnectionsStreamsContentORI --- содержимое изображений, звуков, голосовой информации, видео-
и иных электронных сообщений
}

END

```

RequestedAbonentsORI.asn

```

RequestedAbonentsORI DEFINITIONS IMPLICIT TAGS ::=
BEGIN

EXPORTS RequestedAbonentORI;

IMPORTS TAGGED,
  sorm-request-abonent-ori
FROM Classification

```

```
UserID
FROM IdentifiersORI;
```

```
RequestedAbonentORI ::= SEQUENCE {
  id TAGGED.&id ({RequestedAbonentORIVariants}),
  data TAGGED.&Data ({RequestedAbonentORIVariants}{@id})
}
```

```
RequestedAbonentORIVariants TAGGED ::= {
  requestedORlabonent
}
```

```
requestedORlabonent TAGGED ::= {
  OID {sorm-request-abonent-ori}
  DATA CHOICE {
    user-id [0] UserID,          --- идентификатор пользователя
    nick-name [1] UTF8String (SIZE (1 .. 1024)), --- псевдоним пользователя
    given-name [2] UTF8String (SIZE (1 .. 128)), --- имя
    initial [3] UTF8String (SIZE (1 .. 128)), --- отчество
    family-name [4] UTF8String (SIZE (1 .. 128)), --- фамилия
    address [5] RequestedAddressORI, --- адресные данные
    passport [6] RequestedPassportORI, --- паспортные данные
    relatives [7] UserID, --- идентификатор родственника
    msisdn [8] UTF8String (SIZE (1 .. 64)), --- номер телефона пользователя
    email [9] UTF8String (SIZE (1 .. 256)), --- электронная почта пользователя
    im-id [10] UTF8String (SIZE (1 .. 256)), --- идентификатор в сетях мгновенного обмена сообщениями
    contract [11] UTF8String (SIZE (1 .. 64)) --- номер договора
  }
}
```

```
RequestedAddressORI ::= SEQUENCE {
  country [0] UTF8String (SIZE (1 .. 128)) OPTIONAL, --- страна
  region [1] UTF8String (SIZE (1 .. 128)) OPTIONAL, --- область
  city [2] UTF8String (SIZE (1 .. 128)) OPTIONAL, --- город, поселок, деревня, населенный пункт
  street [3] UTF8String (SIZE (1 .. 128)) OPTIONAL, --- улица
  building [4] UTF8String (SIZE (1 .. 128)) OPTIONAL, --- дом, строение
  apartment [5] UTF8String (SIZE (1 .. 128)) OPTIONAL --- квартира, офис
}
```

```
RequestedPassportORI ::= SEQUENCE {
  passport-serial [0] UTF8String (SIZE (1 .. 16)) OPTIONAL, --- серия паспорта
  passport-number [1] UTF8String (SIZE (1 .. 16)) OPTIONAL --- номер паспорта
}
```

```
END
```

RequestedConnectionsORI.asn

```
RequestedConnectionsORI DEFINITIONS IMPLICIT TAGS ::=
BEGIN
```

```
EXPORTS RequestedConnectionORI;
```

```
IMPORTS TAGGED,
  sorm-request-connection-ori-data-standart,
  sorm-request-connection-ori-data-extended,
  sorm-request-connection-ori-link,
  sorm-request-connection-ori-aaa,
```



```
sorm-request-connection-ori-streams-content,
sorm-request-connection-ori-cert,
sorm-request-connection-ori-keys
FROM Classification
```

```
UserID,
UserTechnicalIdentifier,
ResourceName,
ORISStreamID
FROM IdentifiersORI
```

```
NetworkPeerInfo
FROM NetworkIdentifiers
```

```
LocationInfo
FROM Locations;
```

```
RequestedConnectionORI ::= SEQUENCE {
  id TAGGED.&id ({RequestedConnectionORIVariants}),
  data TAGGED.&Data ({RequestedConnectionORIVariants}{@id})
}
```

```
RequestedConnectionORIVariants TAGGED ::= {
  requestedORIDataStandart --- события информационного взаимодействия пользователей в ИС ОРИ
(базовый набор критериев)
  | requestedORIDataExtended --- события информационного взаимодействия пользователей в ИС ОРИ
(расширенный набор критериев)
  | requestedORILink --- события добавления/исключения связанных пользователей
  | requestedORIAAA --- события регистрации, прекращения регистрации, авторизации, выхода из
информационного сервиса
  | requestedORISStreamsContent --- содержимое изображений, звуков, голосовой информации, видео- и иных
электронных сообщений
  | requestedORICert
  | requestedORIKeys
}
```

```
--- события информационного взаимодействия пользователей в ИС ОРИ (базовый набор критериев)
requestedORIDataStandart TAGGED ::= {
  OID {sorm-request-connection-ori-data-standart}
  DATA CHOICE {
    abonent-id [0] UserID, --- идентификатор пользователя
    abonent-technical-identifier [1] UserTechnicalIdentifier --- технические данные, идентифицирующие
пользователя
  }
}
```

```
--- события информационного взаимодействия пользователей в ИС ОРИ (расширенный набор критериев)
requestedORIDataExtended TAGGED ::= {
  OID {sorm-request-connection-ori-data-extended}
  DATA CHOICE {
    service-id [0] INTEGER (0 .. 65535), --- идентификатор сервиса (расшифровывается по
справочнику)
    abonent-id [1] UserID, --- идентификатор абонента
    abonent-technical-identifier [2] UserTechnicalIdentifier, --- технические данные, идентифицирующие
абонента
    abonent-location [3] LocationInfo, --- местоположение абонента
    contact-id [4] UserID, --- идентификатор контакта
    resource-name [5] ResourceName, --- наименование ресурса
    event-id [6] INTEGER (0 .. 65535), --- тип события (расшифровывается по справочнику)
```

```

owner-id [7]          UserID,          --- идентификатор владельца ресурса
message-text [8]     UTF8String,      --- текст сообщения без разметки и иной служебной
коммуникационной информации
payment-identifier [9] UTF8String (SIZE (1 .. 128)), --- идентификатор платежа
payment-info [10]    UTF8String (SIZE (1 .. 4096)) --- платежная информация
}
}

```

--- события добавления/исключения связанных пользователей

```

requestedORILink TAGGED ::= {
  OID {sorm-request-connection-ori-link}
  DATA CHOICE {
    user-id [0]          UserID,      --- идентификатор пользователя, установившего связь
    connected-user-id [1] UserID,    --- идентификатор пользователя, с которым установили связь
    connected-resource-name [2] ResourceName --- наименование ресурса, с которым установили связь
  }
}

```

--- события регистрации, прекращения регистрации, авторизации, выхода из информационного сервиса

```

requestedORIAAA TAGGED ::= {
  OID {sorm-request-connection-ori-aaa}
  DATA CHOICE {
    service-id [0]       INTEGER (0 .. 65535), --- идентификатор сервиса (расшифровывается по справочнику)
    user-id [1]          UserID,              --- идентификатор пользователя
    user-technical-identifier [2] UserTechnicalIdentifier, --- технические данные, идентифицирующие пользователя
    user-location [3]    LocationInfo,        --- местоположение пользователя
    event-id [4]         INTEGER (0 .. 65535) --- тип события (расшифровывается по справочнику)
  }
}

```

--- содержимое изображений, звуков, голосовой информации, видео- и иных электронных сообщений

```

requestedORISStreamsContent TAGGED ::= {
  OID {sorm-request-connection-ori-streams-content}
  DATA ORISStreamsContentRequest
}

```

```

ORISStreamsContentRequest ::= SEQUENCE {
  stream-id ORISStreamID, --- идентификатор запрашиваемых данных
  duration [1] INTEGER OPTIONAL, --- длительность потоковых данных
  start-offset [2] INTEGER OPTIONAL, --- начальное смещение потоковых данных
  stop-offset [3] INTEGER OPTIONAL --- конечное смещение потоковых данных
}

```

```

requestedORICert TAGGED ::= {
  OID {sorm-request-connection-ori-cert}
  DATA CHOICE {
    client [0] NetworkPeerInfo,
    server [1] NetworkPeerInfo
  }
}

```

```

requestedORIKeys TAGGED ::= {
  OID {sorm-request-connection-ori-keys}
  DATA CHOICE {
    as-server [0] INTEGER,
    as-client [1] INTEGER,
    client [2] NetworkPeerInfo,
    server [3] NetworkPeerInfo,
    sni [4] UTF8String (SIZE (1 .. 128)),
  }
}

```

```

user-id [5]   UTF8String (SIZE (1 .. 64)),
user-login [6] UTF8String (SIZE (1 .. 64)),
client-region [7] UTF8String (SIZE (1 .. 10)),
server-region [8] UTF8String (SIZE (1 .. 10)),
service-name [9] UTF8String (SIZE (1 .. 16))
}
}

```

```
END
```

Sessions.asn

```
Sessions DEFINITIONS IMPLICIT TAGS ::=
BEGIN
```

```
EXPORTS sessionMessage;
```

```
IMPORTS TAGGED,
    sorm-message-session
FROM Classification;
```

```
sessionMessage TAGGED ::= {
  OID {sorm-message-session}
  DATA CHOICE {
    connect [0]      ConnectRequest,    --- запрос на открытие сессии
    connect-response [1] ConnectResponse, --- ответ на запрос открытия сессии
    adjustment [2]   AdjustmentRequest, --- согласование поддерживаемых типов со стороны ПУ
    adjustment-response [3] AdjustmentResponse, --- ответ на запрос согласования данных
    disconnect [4]   DisconnectRequest, --- запрос на закрытие сессии
    disconnect-response [5] DisconnectResponse --- ответ на запрос закрытия сессии
  }
}
```

```
--- запрос на открытие сессии
```

```
ConnectRequest ::= SEQUENCE {
  session-timeout      INTEGER (60 .. 2592000), --- максимальное время неактивности
  max-data-length      INTEGER (10 .. 100000), --- максимальная длина блока отчета (в строках)
  data-packet-window-size INTEGER (4 .. 256), --- окно канала передачи данных максимальное число
  блоков данных, которое может быть отправлено без подтверждения приема
  data-load-timeout    INTEGER (1 .. 60), --- таймаут начала передачи блоков отчетов
  request-response-timeout INTEGER (1 .. 60), --- таймаут ответа на запрос
  data-packet-response-timeout INTEGER (1 .. 60) --- таймаут подтверждения приема блока данных отчета
}
```

```
--- ответ на запрос создания сессии
```

```
ConnectResponse ::= SEQUENCE {
  confirmed-data-packet-window-size INTEGER (4 .. 256), --- подтвержденное окно передачи данных то
  окно, которое может обеспечить ИС COPM; должно быть меньше или равно окну, переданному в
  ConnectRequest
  confirmed-session-timeout INTEGER (60 .. 2592000), --- подтвержденное максимальное время
  неактивности; должно быть больше или равно значению, переданному в ConnectRequest
  confirmed-data-load-timeout INTEGER (1 .. 60), --- подтвержденный таймаут начала передачи блоков
  отчетов; должен быть больше или равен значению, переданному в ConnectRequest
  confirmed-request-response-timeout INTEGER (1 .. 60), --- подтвержденный таймаут ответа на запрос;
  должен быть больше или равен значению, переданному в ConnectRequest
  supports SEQUENCE OF ObjectDescriptor --- весь список поддерживаемых COPM типов запросов,
  типов отчётов
}
```

```

--- согласование поддерживаемых типов со стороны ПУ
AdjustmentRequest ::= SEQUENCE {
  supports SEQUENCE OF ObjectDescriptor --- список поддерживаемых ПУ типов запросов, типов отчётов;
  данный список должен быть меньшим либо равным списку в сообщении ConnectRequest
}

--- ответ на запрос согласования данных
AdjustmentResponse ::= NULL

--- запрос на закрытие сессии
DisconnectRequest ::= NULL

--- ответ на запрос закрытия сессии
DisconnectResponse ::= NULL

END

```

Sorm.asn

```

Sorm DEFINITIONS IMPLICIT TAGS ::=
BEGIN

```

```

EXPORTS DateAndTime,
  FindRange,
  MessageID;

```

```

IMPORTS TAGGED
  FROM Classification

```

```

  sessionMessage
  FROM Sessions

```

```

  trapMessage
  FROM Traps

```

```

  taskMessage
  FROM Tasks

```

```

  reportMessage
  FROM Reports

```

```

  managementMessage
  FROM Management

```

```

  unformattedMessage
  FROM Unformatted

```

```

  reportMessageUNI
  FROM ReportsUNI;

```

```

Version ::= PrintableString
vers Version ::= "3.0.0" --- текущая версия протокола

```

```

--- Оболочка сообщения COPM

```

```

Message ::= SEQUENCE {
  version Version DEFAULT vers, --- версия протокола
  message-id MessageID, --- номер запроса
  message-time DateAndTime, --- время и дата запроса
  operator-name PrintableString (SIZE (1 .. 128)) OPTIONAL, --- наименование оператора связи
  id TAGGED.&id ({{SormPDUs}}, --- идентификатор блока данных

```

```

data    TAGGED.&Data {{SormPDUs}@id}}    --- данные блока данных
}

--- Блок данных сообщения
SormPDUs TAGGED ::= {
  sessionMessage    --- сообщения организации сессии
  | trapMessage      --- сообщения сигналов
  | taskMessage      --- сообщения работы с задачами
  | reportMessage    --- сообщения работы с отчётами
  | managementMessage --- сообщения канала передачи мониторинга (КПМ)
  | unformattedMessage --- сообщения канала передачи неформатированных данных (КПНФ)
  | reportMessageUNI --- сообщения с отобранными по объектам контроля данными
}

--- Номер сообщения
MessageID ::= INTEGER (0 .. 4294967295)

--- Дата и время
DateAndTime ::= UTCTime

--- Диапазон поиска
FindRange ::= SEQUENCE {
  begin-find [0] DateAndTime OPTIONAL, --- время и дата начала поиска информации
  end-find [1] DateAndTime OPTIONAL --- время и дата окончания поиска информации
}

END

```

Sorm743.set.asn

```

Classification.asn
Dictionaries.asn
IdentifiersORI.asn
Locations.asn
Management.asn
NetworkIdentifiers.asn
Reports.asn
ReportsAbonentsORI.asn
ReportsConnectionsORI.asn
ReportsPresense.asn
ReportsUNI.asn
RequestedAbonentsORI.asn
RequestedConnectionsORI.asn
Sessions.asn
Sorm.asn
Tasks.asn
TasksAbonentsORI.asn
TasksConnectionsORI.asn
TasksPresense.asn
TasksUNI.asn
Traps.asn
Unformatted.asn

```

Tasks.asn

```

Tasks DEFINITIONS IMPLICIT TAGS ::=
BEGIN

EXPORTS taskMessage,
  TaskID,
  TelcoID,
  TelcoList,

```

```

ObjectUNI,
LogicalOperation,
CreateTaskResponse;

```

```

IMPORTS TAGGED,
sorm-message-task
FROM Classification

```

```

FindRange,
MessageID
FROM Sorm

```

```

DictionaryTask
FROM Dictionaries

```

```

PresenseTask
FROM TasksPresense

```

```

AbonentsORITask
FROM TasksAbonentsORI

```

```

ConnectionsORITask
FROM TasksConnectionsORI

```

```

UNIControlTaskRequest,
UNIControlTaskResponse
FROM TasksUNI;

```

```

taskMessage TAGGED ::= {
  OID {sorm-message-task}
  DATA CHOICE {
    data-ready-request [0]      DataReadyRequest,      --- запрос готовности данных
    data-ready-response [1]     DataReadyResponse,     --- ответ на запрос готовности данных
    data-load-request [2]       DataLoadRequest,        --- запрос загрузки данных
    data-load-response [3]      DataLoadResponse,       --- ответ на запрос загрузки данных
    data-drop-request [4]       DataDropRequest,        --- запрос удаления данных
    data-drop-response [5]      DataDropResponse,       --- ответ на запрос удаления данных
    data-interrupt-request [6]   DataInterruptRequest,  --- запрос прерывания загрузки данных
    data-interrupt-response [7]  DataInterruptResponse, --- ответ на запрос прерывания загрузки данных
    create-task-request [8]      CreateTaskRequest,     --- запрос на создание задачи по обработке информации
    create-task-response [9]     CreateTaskResponse,   --- ответ на запрос создания задачи
    uni-task-request [12]        UNIControlTaskRequest, --- запрос на постановку/снятие объекта наблюдения на
    контроль
    uni-task-response [13]       UNIControlTaskResponse --- ответ на запрос постановки/снятия объекта
    наблюдения с контроля
  }
}

```

```

--- запрос готовности данных
DataReadyRequest ::= NULL

```

```

--- ответ на запрос готовности данных
DataReadyResponse ::= SEQUENCE OF DataReadyTaskRecord

```

```

DataReadyTaskRecord ::= SEQUENCE {
  task-id TaskID,    --- идентификатор задачи
  result TaskResult --- результат выполнения задачи
}

```

```

TaskResult ::= SEQUENCE {
  result TaskStatus,
  report-records-number [0] INTEGER (0 .. 99999999999) OPTIONAL, --- для выполненной задачи - количество
  записей в отчете
  report-limit-exceeded [1] BOOLEAN OPTIONAL, --- количество записей превысило лимит, заданный
  при создании задачи
  error-description [2] UTF8String (SIZE (1 .. 256)) OPTIONAL --- краткое описание произошедшей ошибки, если
  обнаружена
}

```

```

TaskStatus ::= ENUMERATED {
  data-not-ready (0), --- данные не готовы, задача еще выполняется
  data-ready (1), --- данные есть, задача выполнена
  data-not-found (2), --- данных нет, задача выполнена
  error (3) --- в процессе выполнения задачи произошла ошибка
}

```

```

--- запрос загрузки данных
DataLoadRequest ::= TaskID

```

```

--- ответ на запрос загрузки данных
DataLoadResponse ::= SEQUENCE {
  task-id TaskID, --- идентификатор задачи, сгенерировавшей данный отчет
  data-exists BOOLEAN, --- признак существования результатов исполнения задачи (есть
  данные или нет)
  data-blocks-number INTEGER (0 .. 99999999999) OPTIONAL, --- количество блоков в отчете
  error-description UTF8String (SIZE (1 .. 256)) OPTIONAL --- краткое описание ошибки, если обнаружена
}

```

```

--- запрос удаления данных
DataDropRequest ::= TaskID

```

```

--- ответ на запрос удаления данных
DataDropResponse ::= SEQUENCE {
  task-id TaskID, --- идентификатор задачи, данные которой будут удалены
  successful BOOLEAN, --- признак успешного выполнения запроса
  error-description UTF8String (SIZE (1 .. 256)) OPTIONAL --- краткое описание ошибки, если обнаружена
}

```

```

--- запрос прерывания загрузки данных
DataInterruptRequest ::= TaskID

```

```

--- ответ на запрос прерывания загрузки данных
DataInterruptResponse ::= SEQUENCE {
  request-id MessageID, --- идентификатор прерванного запроса загрузки данных
  successful BOOLEAN, --- признак успешного выполнения запроса
  data-blocks-available INTEGER (0 .. 99999999999) OPTIONAL, --- количество оставшихся переданными
  блоками
  error-description UTF8String (SIZE (1 .. 256)) OPTIONAL --- краткое описание ошибки, если обнаружена
}

```

```

--- запрос на создание задачи по обработке информации

```

```

CreateTaskRequest ::= SEQUENCE {
  telcos [0] TelcoList OPTIONAL, --- список ИС ОПИ (других обслуживаемых ОПИ)
  range [1] FindRange OPTIONAL, --- временной диапазон поиска
  report-limit [2] INTEGER (1 .. 10000000) OPTIONAL, --- ограничение на максимальное количество
  возвращаемых записей
  task [3] CHOICE {

```

```

dictionary [0] DictionaryTask, --- задачи пополнения справочников (нормативно-справочная
информация)
presense [6] PresenseTask, --- задачи предоставления сведений о наличии данных
abonents-ori [7] AbonentsORITask, --- задачи поисков по принадлежности абонентов организаторов
распространения информации
connections-ori [8] ConnectionsORITask --- задачи поисков по соединениям абонентов организаторов
распространения информации
},
find-by-arrive-time BOOLEAN OPTIONAL --- режим поиска информации (true - по дате и времени
поступления; false - по дате и времени события; по умолчанию - false)
}

--- ответ на запрос создания задачи
CreateTaskResponse ::= SEQUENCE {
task-id TaskID OPTIONAL, --- идентификатор задачи
successful BOOLEAN, --- признак успешного выполнения запроса
error-description UTF8String (SIZE (1 .. 256)) OPTIONAL --- краткое описание ошибки, если обнаружена
}

--- идентификатор задачи
TaskID ::= INTEGER (0 .. 4294967295)

--- идентификатор ОРИ или коммуникационного интернет-сервиса ИС ОРИ
TelcoID ::= INTEGER (0 .. 65535)

--- список ИС ОРИ (других обслуживаемых ОРИ)
TelcoList ::= SEQUENCE OF TelcoID

--- идентификатор объекта наблюдения
ObjectUNI ::= INTEGER (0 .. 4294967295)

LogicalOperation ::= ENUMERATED {
operation-open-bracket (0), --- открывающая скобка - "("
operation-close-bracket (1), --- закрывающая скобка - ")"
operation-or (2), --- логическое "или"
operation-and (3), --- логическое "и"
operation-not (4) --- логическое "не"
}

END

```

TasksAbonentsORI.asn

```

TasksAbonentsORI DEFINITIONS IMPLICIT TAGS ::=
BEGIN

EXPORTS AbonentsORITask;

IMPORTS LogicalOperation
FROM Tasks

RequestedAbonentORI
FROM RequestedAbonentsORI;

AbonentsORITask ::= CHOICE {
validate-abonents [0] ValidateAbonentsORITask
}

ValidateAbonentsORITask ::= RequestedAbonentORIIdentifiers

```



```
RequestedAbonentORIdentifiers ::= SEQUENCE OF RequestedAbonentORIdentifier
```

```
RequestedAbonentORIdentifier ::= CHOICE {
  separator [0] LogicalOperation,
  find-mask [1] RequestedAbonentORI
}
```

```
END
```

TasksConnectionsORI.asn

```
TasksConnectionsORI DEFINITIONS IMPLICIT TAGS ::=
BEGIN
```

```
EXPORTS ConnectionsORITask;
```

```
IMPORTS LogicalOperation
  FROM Tasks
```

```
  RequestedConnectionORI
  FROM RequestedConnectionsORI;
```

```
ConnectionsORITask ::= CHOICE {
  validate-standart [0] ValidateStandartTask, --- используется для запросов стандартного класса сложности
  validate-extended [1] ValidateExtendedTask, --- используется для расширенных запросов
  validate-streams-content [2] ValidateStreamsContentTask, --- используется для получения содержимого
  изображений, звуков, голосовой информации, видео- и иных электронных сообщений
  validate-keys [3] ValidateTaskF1
}
```

```
ValidateStandartTask ::= RequestedConnectionIdentifiersORI
ValidateExtendedTask ::= RequestedConnectionIdentifiersORI
ValidateStreamsContentTask ::= RequestedConnectionORI
ValidateTaskF1 ::= RequestedConnectionIdentifiersORI
```

```
RequestedConnectionIdentifiersORI ::= SEQUENCE OF RequestedConnectionParameterORI
```

```
RequestedConnectionParameterORI ::= CHOICE {
  separator [0] LogicalOperation,
  find-mask [1] RequestedConnectionORI
}
```

```
END
```

TasksPresense.asn

```
TasksPresense DEFINITIONS IMPLICIT TAGS ::=
BEGIN
```

```
EXPORTS PresenseTask;
```

```
IMPORTS TAGGED,
  sorm-request-presense
  FROM Classification;
```

```
PresenseTask ::= SEQUENCE {
  id TAGGED.&id ({PresenseListVariants}),
  data TAGGED.&Data ({PresenseListVariants}@id)
}
```

```
PresenseListVariants TAGGED ::= {presenseInfo}
```

```
presenseInfo TAGGED ::= {
  OID {sorm-request-presense}
  DATA RequestPresenseData
}
```

```
RequestPresenseData ::= ENUMERATED {
  dictionaries (3), --- запрос наличия справочников
  abonents-ori (5), --- запрос наличия информации по информации по пользователям ОРИ и их
  идентификаторам
  connections-ori (6) --- запрос наличия информации по соединениям пользователя ОРИ
}
```

```
END
```

TasksUNI.asn

```
TasksUNI DEFINITIONS IMPLICIT TAGS ::=
BEGIN
```

```
EXPORTS UNIControlTaskRequest,
  UNIControlTaskResponse;
```

```
IMPORTS TAGGED
  FROM Classification
```

```
  UserID,
  UserTechnicalIdentifier
  FROM IdentifiersORI
```

```
  TelcoList,
  ObjectUNI
  FROM Tasks;
```

```
UNIControlTaskRequest ::= CHOICE {
  create-uni [0] CreateUNIRequest, --- запрос на создание объекта наблюдения и постановки его на контроль
  drop-uni [1] DropUNIRequest --- запрос на снятие объекта наблюдения с контроля и удаление объекта
  наблюдения
}
```

```
UNIControlTaskResponse ::= CHOICE {
  create-uni [0] CreateUNIResponse, --- ответ на запрос создания объекта наблюдения и постановки его на
  контроль
  drop-uni [1] DropUNIResponse --- ответ на запрос снятия объекта наблюдения с контроля и удаление
  объекта наблюдения
}
```

```
CreateUNIRequest ::= SEQUENCE {
  uni-id ObjectUNI, --- идентификатор объекта наблюдения, переданный ПУ
  uni-criteria UNIParameterORI, --- критерии отбора для объекта наблюдения
  content-load BOOLEAN, --- режим выгрузки содержимого (true - статистика + содержимое; false - только
  статистика)
  telcos TelcoList OPTIONAL --- список ИС ОРИ (других обслуживаемых ОРИ)
}
```

```
UNIParameterORI ::= CHOICE {
  user-id [0] UserID, --- идентификатор пользователя в системе (имя пользователя)
  user-technical-identifier [1] UserTechnicalIdentifier, --- технические данные, идентифицирующие
  пользователя
}
```

```

resource-name [2]      UTF8String (SIZE (1 .. 4096)) --- наименование ресурса
}

CreateUNIResponse ::= SEQUENCE {
  uni-successful      BOOLEAN,                --- признак успешной постановки объекта наблюдения на
  контроль
  uni-error-description UTF8String (SIZE (1 .. 256)) OPTIONAL --- краткое описание ошибки, если обнаружена
}

DropUNIRequest ::= ObjectUNI --- идентификатор объекта наблюдения для снятия с контроля

DropUNIResponse ::= SEQUENCE {
  uni-successful      BOOLEAN,                --- признак успешного снятия объекта наблюдения с контроля
  uni-error-description UTF8String (SIZE (1 .. 256)) OPTIONAL --- краткое описание ошибки, если обнаружена
}

END

```

Traps.asn

```

Traps DEFINITIONS IMPLICIT TAGS ::=
BEGIN

EXPORTS trapMessage;

IMPORTS TAGGED,
  sorm-message-trap
  FROM Classification

  MessageID
  FROM Sorm;

trapMessage TAGGED ::= {
  OID {sorm-message-trap}
  DATA CHOICE {
    trap [0] Trap, --- тип сообщения "сигнал"
    trap-ack [1] TrapAck --- тип сообщения "подтверждение сигнала"
  }
}

--- блок данных сообщения типа "сигнал"
Trap ::= SEQUENCE {
  trap-type TrapType, --- тип сообщения
  trap-message UTF8String (SIZE (1 .. 256)) OPTIONAL, --- описание сообщения
  reference-message MessageID OPTIONAL --- номер сообщение к которому относится данный сигнал
  (например номер сообщения запросившего отчет при прерывании передачи)
}

TrapType ::= ENUMERATED {
  heartbeat (0), --- тестовый пакет
  restart-software (1), --- перезапуск ПО
  unauthorized-access (2), --- попытка несанкционированного доступа
  critical-error (3), --- критическая ошибка ПО, потеря данных, дальнейшая работа невозможна
  major-error (4), --- серьезная ошибка ПО, потеря данных, но дальнейшая работа возможна
  minor-error (5) --- незначительная ошибка ПО, данные не потеряны, дальнейшая работа возможна
}

--- блок данных сообщения типа "подтверждение сигнала" (номер сообщения TrapAck должен соответствовать
номеру сообщения Trap)
TrapAck ::= NULL

```

END

Unformatted.asn

```
Unformatted DEFINITIONS IMPLICIT TAGS ::=
BEGIN
```

```
EXPORTS unformattedMessage;
```

```
IMPORTS TAGGED,
  sorm-message-unformatted
  FROM Classification
```

```
  TelcoList
  FROM Tasks
```

```
  Acknowledgement
  FROM Reports
```

```
  ConnectionsORIReport
  FROM ReportsConnectionsORI
```

```
  DateAndTime,
  MessageID
  FROM Sorm;
```

```
unformattedMessage TAGGED ::= {
  OID {sorm-message-unformatted}
  DATA CHOICE {
    request [0] RawRequest,
    response [1] RawResponse,
    report [2] RawReport,
    report-ack [3] RawAcknowledgement
  }
}
```

```
RawRequest ::= SEQUENCE {
  telcos TelcoList, --- список ИС ОРИ (других обслуживаемых ОРИ)
  raw-task RawRequestTask --- запрос получения неформатированных данных
}
```

```
RawRequestTask ::= CHOICE {
  data-types-request [0] DataTypesRequest, --- запрос проверки наличия вида неформатированных данных в ИС
  COPM
  data-start-request [1] DataStartRequest, --- запрос на начало передачи неформатированных данных
  data-stop-request [2] DataStopRequest --- запрос на остановку передачи неформатированных данных
}
```

```
RawResponse ::= CHOICE {
  data-types-response [0] DataTypesResponse, --- ответ на запрос проверки наличия неформатированных вида
  данных в ИС COPM
  data-start-response [1] DataStartResponse, --- ответ на запрос начала передачи неформатированных данных
  data-stop-response [2] DataStopResponse --- ответ на запрос остановки передачи неформатированных
  данных
}
```

```
DataTypesRequest ::= RawDataType
```

```
DataTypesResponse ::= SEQUENCE {
```

```

successful BOOLEAN, --- признак наличия в ИС COPM запрошенного вида неформатированных данных
selected-type RawDataType, --- выбранный вид данных для передачи
time-from DateAndTime, --- начало временного периода в буфере, начиная с которого накоплены данные
time-to DateAndTime --- конец временного периода в буфере, по которому накоплены данные
}

DataStartRequest ::= SEQUENCE {
time-from DateAndTime, --- начало временного периода в буфере, с которого необходимо получить данные
time-to DateAndTime, --- конец временного периода в буфере, с которого необходимо получить данные
raw-type RawDataType --- тип неформатированных данных передачи
}

DataStartResponse ::= BOOLEAN

DataStopRequest ::= NULL

DataStopResponse ::= BOOLEAN

--- типы данных, передаваемых ИС COPM
RawDataType ::= ENUMERATED {
data-reports-ori (3), --- записи о соединениях абонентов организаторов распространения информации
raw-ori (4) --- записи о соединениях абонентов организаторов распространения информации в
бинарном виде
}

RawReport ::= SEQUENCE {
request-id MessageID, --- идентификатор запроса
stream-id UTF8String (SIZE (1 .. 256)), --- идентификатор потока в сессии
total-blocks-number INTEGER (0 .. 99999999999), --- общее количество блоков в отчете
block-number INTEGER (1 .. 1000000000000), --- порядковый номер текущего блока
report-block RawDataBlock --- блок данных отчета
}

RawDataBlock ::= CHOICE {
reports-ori [3] ConnectionsORIRreport,
raw-ori [4] RawBytesBlock
}

RawBytesBlock ::= SEQUENCE OF RawBytes

RawBytes ::= OCTET STRING (SIZE (1 .. 4096))

RawAcknowledgement ::= Acknowledgement

END

```

Приложение № 5
к Требованиям к оборудованию и программно-техническим средствам, используемым организатором распространения информации в сети «Интернет» в эксплуатируемых им информационных системах, для проведения уполномоченными государственными органами, осуществляющими оперативно-розыскную деятельность или обеспечение безопасности Российской Федерации, мероприятий в целях реализации возложенных на них задач, утвержденным приказом Министерства цифрового развития, связи и массовых коммуникаций Российской Федерации от 29.10.2018 2018 № 571

Требования, предъявляемые к функционированию канала управления кпд1

1. ПТС ОРИ должны обеспечивать подключение ПУ и обработку поступающих запросов по каналу кпд1 в соответствии с приведенными диаграммами состояний переходов.

2. Диаграмма состояний переходов ПТС ОРИ по кпд1 приведена на схеме 1 настоящего приложения:

1) ПТС ОРИ по TCP-порту кпд1 должны находиться в ожидании входящих соединений; после установления соединения с ПУ должна выполняться взаимная SSL/TLS-аутентификация;

2) если SSL/TLS-соединения по каналам управления и данных установлены, а сессия не открыта, ПТС ОРИ следует реагировать только на сообщение «Запрос на открытие сессии»; при попытке посылок каких-либо других сообщений со стороны ПУ ПТС ОРИ должны разорвать TCP-соединения по каналу управления и каналу данных и перевести каналный уровень подключения в исходное состояние;

3) при получении сообщения «Запрос на открытие сессии» ПТС ОРИ должны создать список поддерживаемых ПТС ОРИ типов запросов, отчетов и сигналов (в том числе и предыдущих версий) и отослать его на ПУ;

4) после отсылки списка поддерживаемых типов ПТС ОРИ должны ожидать от ПУ списка поддерживаемых им запросов, отчетов и сигналов; список поддерживаемых ПУ типов является подмножеством списка типов в ПТС ОРИ;

5) при получении от ПУ списка поддерживаемых ПУ запросов ПТС ОРИ должны послать сообщение «Ответ на согласование списка поддерживаемых типов» и создать сессию;

6) после создания сессии кпд1 ПТС ОРИ должны быть переведены в режим ожидания команд от ПУ; при поступлении команды со стороны ПУ ПТС ОРИ должны ее выполнить и сформировать результат, который в виде сообщения «ответа» должен поступить на ПУ;

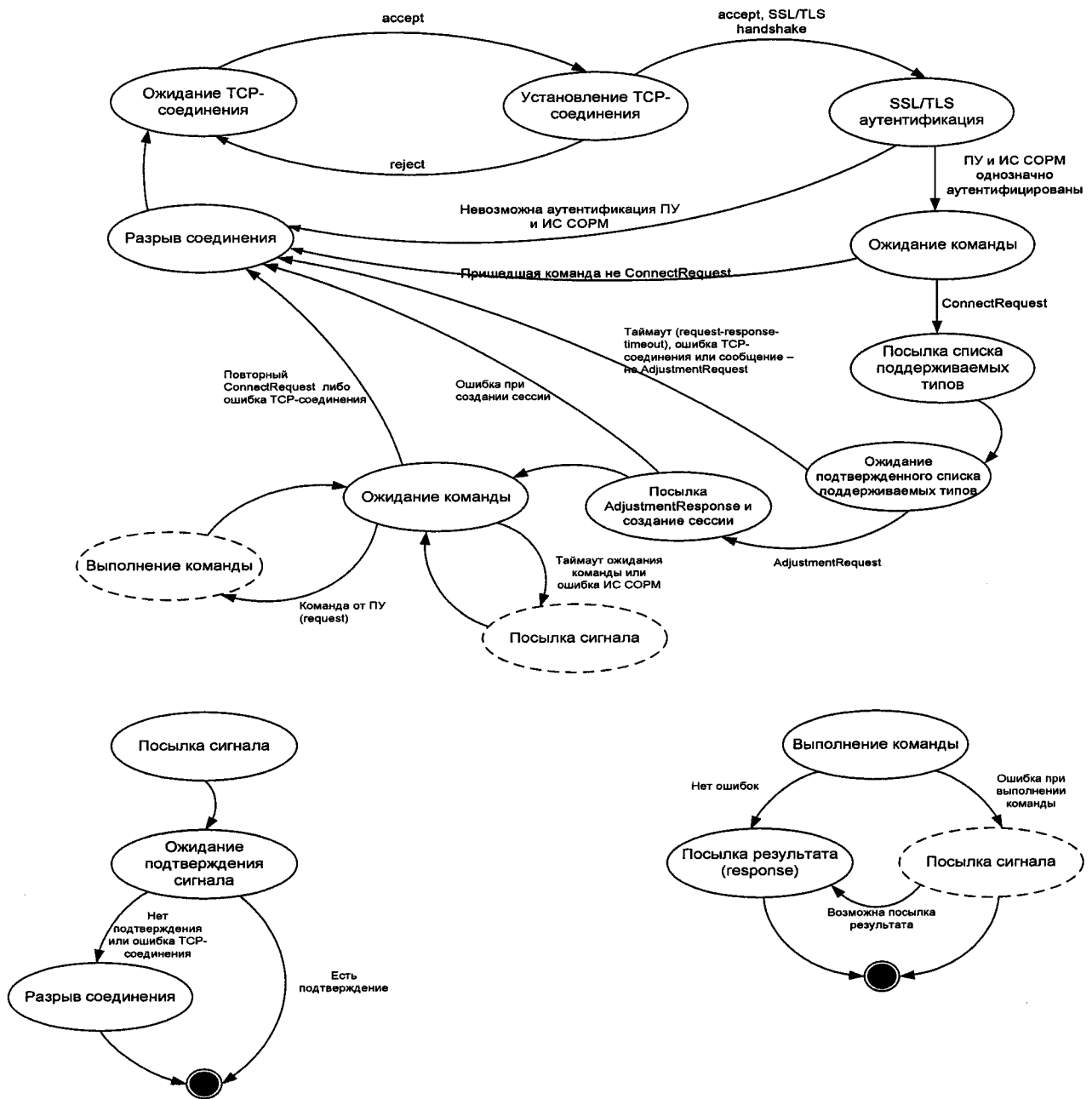


Схема 1

7) в случае сбоя в функционировании ПТС ОРИ, вызванного причинами, не предусмотренными режимом нормального функционирования системы, по каналу управления должен передаваться «сигнал» - «Критическая ошибка ПО, потеря данных, дальнейшая работа невозможна»

с соответствующим описанием проблемы; все задачи, которые были в процессе выполнения, когда произошел сбой, а также данные выполненных задач, поврежденные в результате произошедшего сбоя, имеют «признак результата выполнения задачи» (TaskResult), равный значению «ошибка» (error), с соответствующим описанием проблемы. В случае, если для восстановления работоспособности ПТС ОРИ требуется их перезагрузка, то по каналу управления следует активировать команду «Перезапуск ПО». При этом ПТС ОРИ и ПУ должны закрыть все открытые на текущий момент сессии;

8) в случае наличия признаков сбоя или ошибки выполнения конкретной задачи ПТС ОРИ в режиме нормального функционирования по каналу управления должны передать прерывание «Серьезная ошибка ПО, потеря данных, но дальнейшая работа возможна» с соответствующим описанием проблемы. Результат выполнения данной задачи имеет «признак результата выполнения задачи» (TaskResult), соответствующий режиму «ошибка» (error), с описанием причин сбоя или ошибки;

9) на каждый «сигнал», переданный ПТС ОРИ, ПУ должен ответить сообщением «подтверждение сигнала» по кпд1. Отсутствие подтверждения в течение времени RequestResponseTimeout, которое задается при открытии сессии, свидетельствует о прерывании соединения и активирует операции, описанные в пункте 12 приложения № 2 к настоящим требованиям;

10) при отсутствии команд ПУ в течение «максимального времени неактивности» (session-timeout, задается в ConnectRequest) ПТС ОРИ должны направить на ПУ «сигнал» Heartbeat и ожидать его подтверждения согласно подпункту 9 настоящего пункта.

2. ПУ с задаваемым интервалом должен выполнять попытки установления TCP-соединения с ПТС ОРИ по заданному порту кпд1. Диаграмма состояний переходов ПУ по кпд1 приведена на схеме 2 настоящего приложения:

1) ПУ должен ожидать установления соединения по кпд1. После установления соединения выполняется взаимная SSL/TLS-аутентификация;

2) после установления TCP-соединения по кпд1 и проведения взаимной аутентификации ПТС ОРИ и ПУ по кпд1 должны направить команду создания сессии (ConnectRequest);

3) ПУ должен ожидать сообщения «ответ» от ПТС ОРИ на отправленную команду в течение времени «таймаут ответа на запрос» (request-response-timeout);

4) если сообщение не получено в течение времени «таймаут ответа на запрос», ПУ должен разорвать соединения к ПТС ОРИ по кпд1 и кпд2 и перевести их в начальное состояние согласно подпункту 1 настоящего пункта. Период ожидания сообщения «ответ» не должно зависеть от приема сообщений «сигналов», поступающих в этот интервал времени;

5) при получении сообщения «Ответ на запрос создания сессии» ПУ должен создать список поддерживаемых ПУ типов запросов, отчетов и сигналов и отправить его ПТС ОРИ;

6) после отправки сообщения «Согласование поддерживаемых типов со стороны ПУ» ПУ должен ожидать ответа на сообщение от ПТС ОРИ. Поведение ПУ при ожидании ответа должно соответствовать подпункту 4 настоящего пункта;

7) во время ожидания сообщения «ответ» ПТС ОРИ должны послать на ПУ сообщение «сигнал» (в том числе HeartBeat), а ПУ направить «подтверждение» о принятии «сигнала» (в том числе HeartBeat) и продолжать ожидать сообщения «ответ» на отосланную команду;

8) после создания сессии ПУ должен послать поступающие команды на ПТС ОРИ и ожидать сообщений «ответов» на них согласно подпунктам 3, 4 и 7 настоящего пункта;

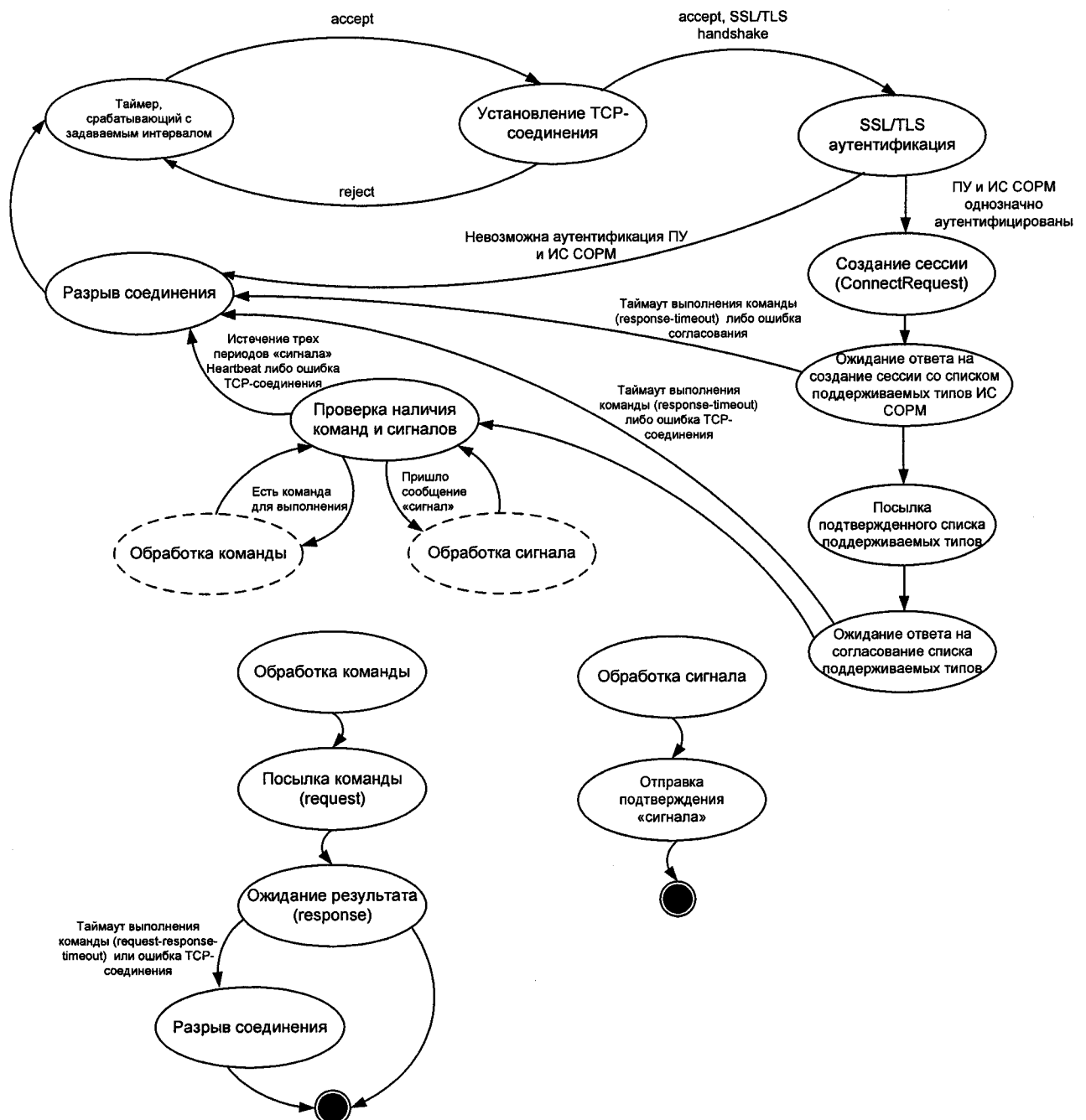


Схема 2

9) если при ожидании поступления в ПУ команд ПТС ОРИ не посылали «сигнал» (HeartBeat) в течение трех интервалов «максимального времени неактивности» (session-timeout, задается в ConnectRequest), ПУ должен разорвать соединения к ПТС ОРИ по кпд1 и кпд2 и перевести их в начальное состояние согласно подпункту 1 настоящего пункта;

10) при поступлении сообщения «Запрос на закрытие сессии» (DisconnectRequest) ПУ должен отослать его на ПТС ОРИ, ожидать сообщения «ответ» согласно подпунктам 3, 4, 7 настоящего пункта, после чего разорвать соединение по кпд2 и кпд1 и перевести их в начальное состояние согласно подпункту 1 настоящего пункта.

Приложение № 6
к Требованиям к оборудованию и программно-техническим средствам, используемым организатором распространения информации в сети «Интернет» в эксплуатируемых им информационных системах, для проведения уполномоченными государственными органами, осуществляющими оперативно-розыскную деятельность или обеспечение безопасности Российской Федерации, мероприятий в целях реализации возложенных на них задач, утвержденным приказом Министерства цифрового развития, связи и массовых коммуникаций Российской Федерации от 19.10.2018 2018 № 571

Требования, предъявляемые к функционированию канала данных кпд2

1. ПТС ОРИ должны обеспечивать подключение ПУ и обработку поступающих запросов по каналу кпд2 в соответствии с приведенными диаграммами состояний переходов.

2. Диаграмма состояний переходов ПТС ОРИ по кпд2 приведена на схеме 1:

1) ПТС ОРИ по TCP-порту кпд2 должны ожидать входящих соединений. После установления соединения должна выполняться взаимная SSL/TLS-аутентификация;

2) если на ПТС ОРИ был передан запрос ПУ «Запрос загрузки данных» (DataLoadRequest), ПТС ОРИ должны послать «ответ на запрос загрузки данных» (DataLoadResponse) по каналу кпд1 и начать передачу данных блоков отчетов по кпд2 при их наличии. ПУ должен получать блоки отчетов по кпд2 до получения «ответа на запрос загрузки данных» (DataLoadResponse) по кпд1;

3) если количество переданных без получения «подтверждения» о принятии серии блоков «отчетов» по всем задачам, по которым выполняется загрузка на ПУ данных, меньше «окна канала передачи данных» (параметр data-racket-window-size в запросе ПУ «Запрос на открытие сессии» ConnectRequest), то ПТС ОРИ должны выполнить подготовку новых блоков отчетов по загружаемым задачам и послать их на ПУ. Количество подготовленных и переданных без получения «подтверждения» блоков не должно превышать «размер окна канала передачи данных»;

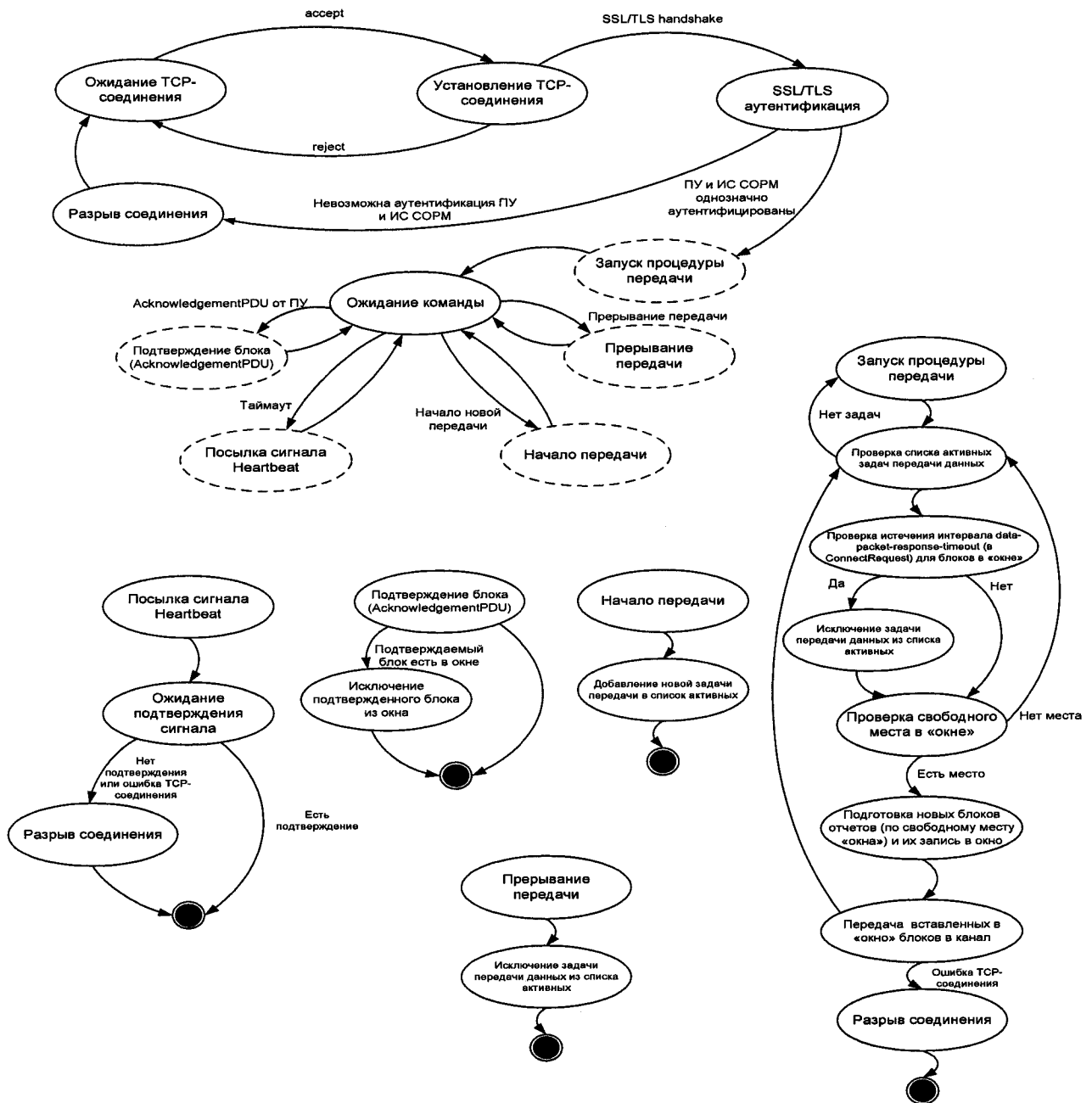


Схема 1

4) максимальная задержка подтверждения приема блока данных со стороны ПУ не должна превышать параметр «таймаут подтверждения приема блока данных отчета» (data-packet-response-timeout), указываемый при создании сессии. Если задержка подтверждения превысила заданное значение, то оставшиеся для передачи блоки данных не должны отправляться, а по каналу управления должен передаваться «сигнал» «Незначительная ошибка ПО, данные не потеряны, дальнейшая работа возможна» с соответствующим описанием проблемы. При этом в поле «reference-message» сообщения «сигнал» должен указываться идентификатор сообщения блока отчета, по которому не поступило подтверждение приема;

5) при получении «подтверждения» блока «отчета» ПТС ОРИ должны записать информацию об ошибочно принятом ПУ блоке и ошибочных записях в блоке в журнал, при этом передача последующих блоков по задаче на ПУ не должна прерываться. ПТС ОРИ должны предоставить техническому персоналу ОРИ доступ к журналу с записями об ошибочно принятых на ПУ блоках отчетов и средства исправления ошибочных данных в отчетах. Подтвержденные блоки должны исключаться из «окна канала передачи данных» (в «окне канала передачи данных» должны остаться только неподтвержденные блоки);

6) в случае разрыва TCP/IP соединения кпд2 при существующем соединении кпд1, по кпд1 должно быть передано сообщение «Незначительная ошибка ПО, данные не потеряны, дальнейшая работа возможна» с соответствующим описанием проблемы. В данном случае должно продолжаться выполнение установления соединения по кпд2 согласно пункту 10 приложения № 2 к настоящим требованиям;

7) передача блоков данных должна прерываться в случае получения ПТС ОРИ запроса ПУ «Запрос прерывания загрузки данных».

8) если по кпд2 не производится передача блоков отчетов в течение «максимального времени неактивности» (session-timeout при создании сессии ConnectRequest), ПТС ОРИ должна направить на ПУ «сигнал» (HeartBeat) и ожидать его подтверждения согласно подпункту 9 пункта 1 приложения № 5 к настоящим требованиям.

3. Диаграмма состояний переходов ПУ по кпд2 приведена на схеме 2:

1) ПУ с задаваемым интервалом должен выполнять попытки установления TCP-соединения к ПТС ОРИ по заданному порту кпд2. После установления соединения должна выполняться взаимная SSL/TLS-аутентификация;

2) при поступлении запроса ПУ «Запрос загрузки данных» (DataLoadRequest) ПУ должен ожидать начала передачи данных в течение времени «таймаут начала передачи блоков отчетов» (data-load-timeout в ConnectRequest). Если данные не поступают в течение вышеописанного периода, то ПУ должен разорвать соединения по каналам кпд1 и кпд2 и перевести соединения в начальное состояние согласно подпункту 1 пункта 1 приложения № 5 к настоящим требованиям и подпункту 1 настоящего пункта;

3) при поступлении блока отчета ПУ должен произвести декодирование полученного блока и сохранение декодированных данных;

4) в ответ на переданный блок данных ПУ должен послать сообщение «подтверждение» получения блока отчета. Количество последовательно переданных ПТС ОРИ блоков данных без подтверждения со стороны ПУ определяется параметром «размер окна канала передачи данных», который согласовывается при создании сессии. При подтверждении блока отчета ПУ должен сигнализировать об ошибке декодирования блока. В этом случае в сообщении «подтверждение» приема для ошибочно декодированного блока ПУ, в случае возможности, следует указать номер записи в блоке, начиная с которой декодирование не удалось;



Схема 2

5) если при ожидании поступления в ПУ блоков отчетов ПТС ОРИ не посылали «сигнал» HeartBeat в течение трех интервалов «максимального времени неактивности» (session-timeout, задается в ConnectRequest), ПУ должен разорвать соединения к ПТС ОРИ по кпд1 и кпд2 и перевести их в начальное состояние согласно подпункту 1 пункта 1 приложения № 5 к настоящим требованиям и подпункту 1 пункта 2 настоящего приложения.

Приложение № 7
к Требованиям к оборудованию и программно-техническим средствам, используемым организатором распространения информации в сети «Интернет» в эксплуатируемых им информационных системах, для проведения уполномоченными государственными органами, осуществляющими оперативно-розыскную деятельность или обеспечение безопасности Российской Федерации, мероприятий в целях реализации возложенных на них задач, утвержденным приказом Министерства цифрового развития, связи и массовых коммуникаций Российской Федерации от 29.10.2018 2018 № 571

Требования, предъявляемые к функционированию канала мониторинга кпд3

1. ПТС ОРИ должны обеспечивать подключение ПУ и обработку следующих поступающих от ПУ запросов по каналу кпд3 (канал мониторинга):

1) запрос на получение структуры ПТС ОРИ и списка модулей ПТС ОРИ («GetStructureRequest»);

2) запрос на получение конфигурации модуля ПТС ОРИ («GetModuleConfigRequest»);

3) запрос на изменение конфигурации модуля ПТС ОРИ («SetModuleConfigRequest»);

4) запрос на получение состояния модуля ПТС ОРИ («CheckModuleRequest»).

2. На каждый запрос по кпд3 ПТС ОРИ должны посылать ответ, содержащий результат обработки соответствующего запроса – «ManagementResponse».

3. Диаграммы состояний перехода ПТС ОРИ и ПУ по кпд3 соответствуют диаграммам для кпд1, представленным на схеме 1 и схеме 2 приложения № 5 к настоящим требованиям:

1) ПТС ОРИ по TCP-порту кпд3 должны находиться в состоянии ожидания входящих соединений. После установления соединения с ПУ должна выполняться взаимная SSL/TLS-аутентификация;

2) если SSL/TLS-соединение по кпд3 установлено, а сессия не открыта, ПТС ОРИ должны реагировать только на сообщение «Запрос на открытие сессии». Создание сессии должно быть аналогично приведенному в подпунктах

3 – 5 пункта 1 приложения № 5 к настоящим требованиям. При попытке отправки каких-либо других сообщений со стороны ПУ ПТС ОРИ должно быть разорвано TCP-соединение по кпд3 и канальный уровень подключения должен быть возвращен в исходное состояние;

3) после создания сессии ПТС ОРИ должны быть переведены в режим ожидания команд от ПУ. Обработка поступающих команд и отправка сигналов должна производиться в соответствии с подпунктами 6 - 10 пункта 1 приложения № 5 к настоящим требованиям;

4) ПУ с заданным интервалом должен попытаться установить TCP-соединения с ПТС ОРИ по заданному порту. После установления соединения должна выполняться взаимная SSL/TLS-аутентификация;

5) ПУ по каналу кпд3 должен направить команду для создания сессии (ConnectRequest);

6) ожидание подтверждения создания сессии, согласование списка поддерживаемых типов, отправка команд, ожидание ответов и обработка полученных от ПТС ОРИ сигналов по кпд3 должно производиться ПУ согласно подпунктам 3 – 9 пункта 2 приложения № 5 к настоящим требованиям.

4. ПТС ОРИ должны обеспечивать получение ПУ следующей информации о структуре и функционировании ПТС ОРИ по запросу ПУ:

1) о структуре и составе ПТС ОРИ, составе и состоянии интерфейсов взаимодействия ПТС ОРИ с ПУ;

2) об установленном в ПТС ОРИ общесистемном ПО ПТС ОРИ, перечне и состоянии программных модулей в составе ПО ПТС ОРИ;

3) о точках подключения ПТС ОРИ к ИС ОРИ и интерфейсах ввода информации в ПТС ОРИ.

5. ПТС ОРИ по запросу ПУ должна предоставить следующую информацию в части структуры и состава ПТС ОРИ, состава и состоянии интерфейсов взаимодействия ПТС ОРИ с ПУ:

1) перечень коммутационного и серверного оборудования, средств хранения данных с его идентификацией;

2) идентификацию интерфейсов подключения оборудования ПТС ОРИ друг к другу;

3) параметры для серверного оборудования (на момент формирования запроса ПУ):

общий и занятый объем оперативной памяти;

количество сетевых интерфейсов с их идентификацией, текущую нагрузку;

общее количество процессоров, текущую загрузку;

общий объем дискового пространства, объем свободного пространства;

4) параметры технических средств хранения данных:

перечень модулей, составляющих средства хранения данных с их идентификацией;

для каждого входящего в состав средств хранения данных модуля: общий объем дискового пространства, объем свободного дискового пространства

и текущее состояние модуля (штатное функционирование, сбой, не функционирует), текстовую расшифровку сбоя.

6. ПТС ОРИ по запросу ПУ должна предоставить текущую информацию на момент формирования запроса в части точек подключения ПТС ОРИ к ИС ОРИ, интерфейсов ввода информации в ПТС ОРИ, содержащую:

1) перечень точек подключения к сети связи и точек ввода информации в ПТС ОРИ с их идентификацией;

2) для каждой точки подключения предоставляет информацию о: виде поступающих в ПТС ОРИ сведений (о соединениях абонентов, о платежах, об абонентах, иные сведения);

состоянии точки подключения (ввода информации) (штатное функционирование, сбой, не функционирует), текстовую расшифровку сбоя;

объеме информации, поступающей в секунду, в том числе количестве записей, объеме (байт);

периоде времени, в течение которого на точку подключения/ввода информации в ПТС ОРИ не поступала информация.

7. ПТС ОРИ по запросу ПУ должна предоставить следующую информацию в части состава общесистемного ПО ПТС ОРИ, его текущего состояния:

1) перечень установленного общесистемного ПО и его идентификации;

2) предоставление для общесистемного ПО информации:

идентификатора ПТС ОРИ, на котором установлено ПО;

наименование общесистемного ПО;

текущее состояние ПО (штатное функционирование, сбой, не функционирует), текстовую расшифровку сбоя;

3) перечень установленного ПО ПТС ОРИ и его идентификации;

4) предоставление для ПО ПТС ОРИ информации:

идентификатора ПТС ОРИ, на котором установлено ПО;

назначение (определяется разработчиком ПС ОРИ);

текущее состояние (штатное функционирование, сбой, не функционирует), текстовую расшифровку сбоя;

список контролируемых параметров (определяется разработчиком ПТС ОРИ).

Приложение № 8
к Требованиям к оборудованию и программно-техническим средствам, используемым организатором распространения информации в сети «Интернет» в эксплуатируемых им информационных системах, для проведения уполномоченными государственными органами, осуществляющими оперативно-розыскную деятельность или обеспечение безопасности Российской Федерации, мероприятий в целях реализации возложенных на них задач, утвержденным приказом Министерства цифрового развития, связи и массовых коммуникаций Российской Федерации от 19.10.2018 2018 № 571

Требования, предъявляемые к функционированию канала неформатированных данных кпд4

1. ПТС ОРИ должны обеспечивать подключение ПУ и обработку поступающих запросов по каналу кпд4 (канал неформатированных данных). Кпд4 является необязательным и должен реализовываться ПТС ОРИ по согласованию с ПУ.

2. ПТС ОРИ должны обеспечивать прием и обработку от ПУ по каналу кпд4 следующих запросов:

1) «запрос проверки наличия вида неформатированных данных в ПТС ОРИ» (DataTypesRequest);

2) «запрос на начало передачи неформатированных данных» (DataStartRequest);

3) «запрос на остановку передачи неформатированных данных» (DataStopRequest).

3. На каждый запрос по кпд4 ПТС ОРИ должны посылать ответ на ПУ, содержащий результат обработки соответствующего запроса.

4. ПТС ОРИ и (или) ИС ОРИ должны накапливать информацию с неформатированными данными в буфере. Данные в буфере должны упорядочиваться в соответствии с временем их поступления.

5. Диаграмма состояний переходов ПТС ОРИ по кпд4 приведена на схеме 1 настоящего приложения:

1) ПТС ОРИ должны ожидать и устанавливать соединение согласно подпунктам 1 – 2 пункта 3 приложения № 7 к настоящим требованиям;

неформатированных данных», «запрос на остановку передачи неформатированных данных» (DataStartRequest/ DataStopRequest);

3) при приеме команды «запрос на начало передачи неформатированных данных» ПТС ОРИ, находясь в режиме ожидания команд, должны направлять «ответ на запрос начала передачи неформатированных данных» и перевести канал кпд4 в режим передачи данных, при этом должна производиться передача данных того типа, который указан в команде запроса;

4) при приеме команды «запрос на остановку передачи неформатированных данных» ПТС ОРИ, находясь в режиме ожидания команд, должны направлять «ответ на запрос остановки передачи неформатированных данных» с отрицательным результатом;

5) в режиме передачи данных посылка блоков отчетов и их подтверждение должны производиться согласно подпунктам 3, 5 и 8 пункта 2 приложения № 6 к настоящим требованиям;

6) максимальная задержка подтверждения приема блока данных со стороны ПУ не должна превышать параметр «таймаут подтверждения приема блока данных отчета» (data-packet-response-timeout), указываемый при создании сессии. Если задержка подтверждения превысила заданное значение, то оставшиеся для передачи блоки данных не должны отправляться, при этом ПТС ОРИ должны разорвать соединение по кпд4 и перевести кпд4 в изначальное состояние;

7) передача неформатированных данных соответствующего типа должна производиться из буфера кпд4 согласно подпункту 3 пункта 1 приложения № 5 к настоящим требованиям;

8) при приеме команды «запрос на остановку передачи неформатированных данных» ПТС ОРИ, находясь в режиме передачи данных, должны направлять «ответ на запрос остановки передачи неформатированных данных», завершить посылку блоков данных и перейти в режим ожидания команд;

9) при приеме команды «запрос на начало передачи неформатированных данных» ПТС ОРИ, находясь в режиме передачи данных, должны направлять «ответ на запрос начала передачи неформатированных данных» с отрицательным результатом. При этом передача неформатированных данных не должна прекращаться;

10) исходные данные о соединениях в виде неформатированных данных должны записываться в буфер независимо от текущего режима работы канала кпд4 ПТС ОРИ;

11) если команда «запрос на остановку передачи неформатированных данных» поступила в момент передачи из буфера файловых данных, то передаваемый файл должен сохраняться в буфере и быть доступным для передачи после посылки команды «запрос на начало передачи неформатированных данных» с учетом ограничений по длительности хранения, согласно подпункту 3 пункта 1 приложения № 5 к настоящим требованиям.

6. Диаграмма состояний переходов ПУ по кпд4 представлена на схеме 2:

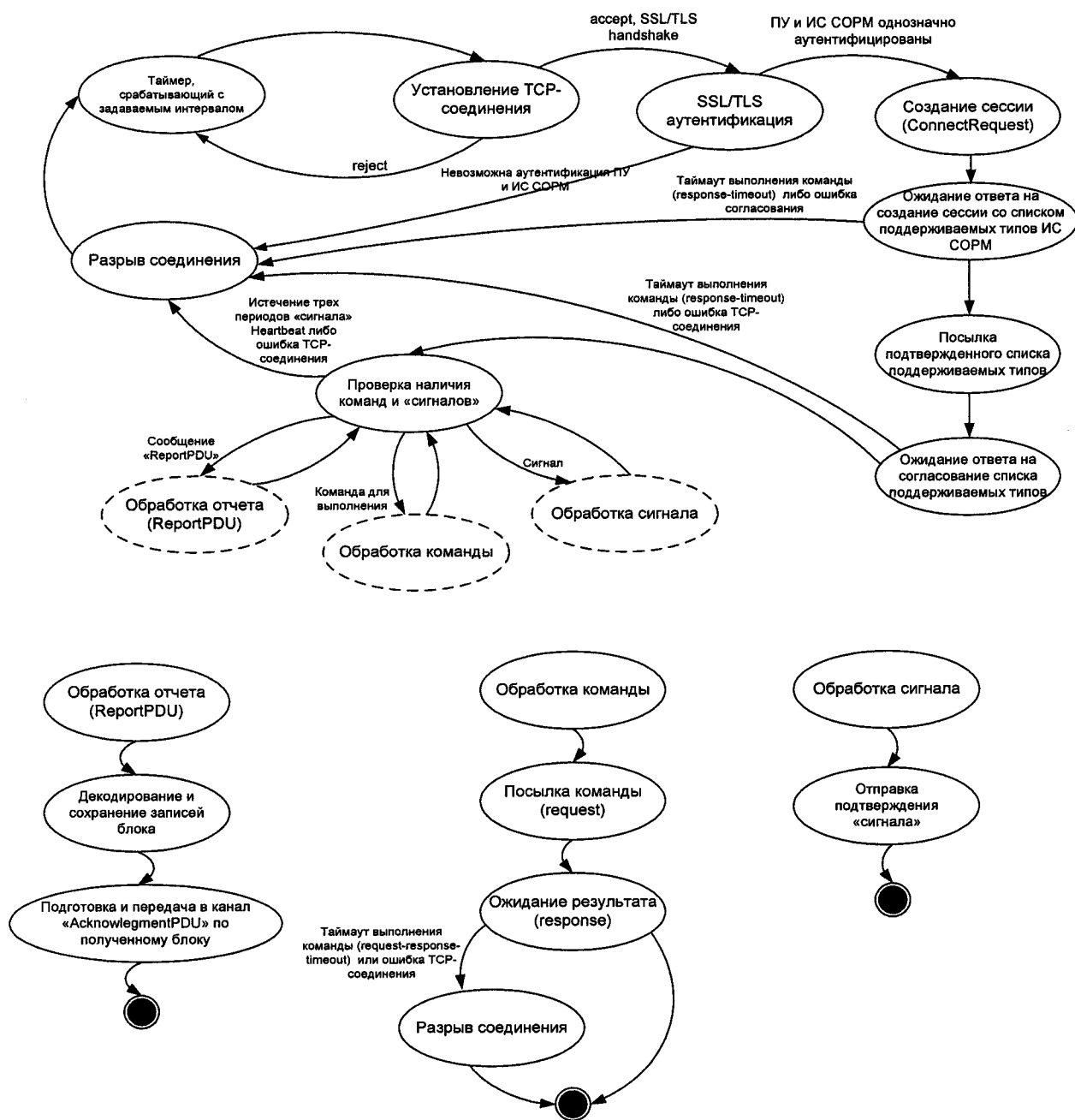


Схема 2

1) ПУ с задаваемым интервалом должен выполнять попытки установления TCP-соединения с ПТС ОРИ по заданному порту. После установления соединения должна выполняться взаимная SSL/TLS-аутентификация;

2) ПУ по кпд4 направляет команду создания сессии (ConnectRequest);

3) ожидание подтверждения создания сессии, согласование списка поддерживаемых типов, отправка команд, ожидание ответов и обработка полученных от ПТС ОРИ сигналов по кпд4 производится ПУ согласно подпунктам 3 - 9 пункта 2 приложения № 5 к настоящим требованиям, за исключением команд «запрос на начало передачи неформатированных данных» и «запрос на остановку передачи неформатированных данных» (DataStartRequest/ DataStopRequest);

4) при посылке команд «запрос на начало передачи неформатированных данных» и «запрос на остановку передачи неформатированных данных» ПУ должен ожидать результата (DataStartResponse) согласно подпунктам 3 – 4 пункта 2 приложения № 5 к настоящим требованиям, после чего должен переводить канал кпд4 в режим передачи данных;

5) в режиме передачи данных ПУ должен производить прием, декодирование и подтверждение приема данных согласно подпунктам 2 – 4 пункта 2 приложения № 5 к настоящим требованиям;

6) при приеме команды «запрос на остановку передачи неформатированных данных» (DataStopRequest) ПУ, находясь в режиме передачи данных, должен завершить прием блоков данных и перейти в режим передачи команд.
