ФЕДЕРАЛЬНОЕ АГЕНТСТВО

ПО ТЕХНИЧЕСКОМУ РЕГУЛИРОВАНИЮ И МЕТРОЛОГИИ

НАЦИОНАЛЬНЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

ГОСТ Р 71807— 2024

Цифровая промышленность

УНИФИЦИРОВАННАЯ АРХИТЕКТУРА OPC

Часть 2

Модель безопасности

(IEC/TR 62541-2:2020, NEQ)

Издание официальное

Москва
Российский институт стандартизации
2025

Предисловие

1 РАЗРАБОТАН Ассоциацией «Цифровые инновации в машиностроении» и Федеральным государственным бюджетным учреждением «Российский институт стандартизации» (ФГБУ «Институт стандартизации»)

2 ВНЕСЕН Техническим комитетом по стандартизации ТК 306 «Измерения, управление и автоматизация в промышленных процессах»

3 УТВЕРЖДЕН И ВВЕДЕН В ДЕЙСТВИЕ Приказом Федерального агентства по техническому регулированию и метрологии от 29 ноября 2024 г. № 1803-ст

4 Настоящий стандарт разработан с учетом основных нормативных положений международного документа МЭК/ТР 62541-2:2020 «Унифицированная архитектура OPC. Часть 2. Модель безопасности» (IEC/TR 62541-2:2020 «OPC unified architecture — Part 2: Security model», NEQ)

5 ВВЕДЕН ВПЕРВЫЕ

Правила применения настоящего стандарта установлены в статье 26 Федерального закона от 29 июня 2015 г. № 162-ФЗ «О стандартизации в Российской Федерации». Информация об изменениях к настоящему стандарту публикуется в ежегодном (по состоянию на 1 января текущего года) информационном указателе «Национальные стандарты», а официальный текст изменений и поправок — в ежемесячном информационном указателе «Национальные стандарты». В случае пересмотра (замены) или отмены настоящего стандарта соответствующее уведомление будет опубликовано в ближайшем выпуске ежемесячного информационного указателя «Национальные стандарты». Соответствующая информация, уведомление и тексты размещаются также в информационной системе общего пользования — на официальном сайте Федерального агентства по техническому регулированию и метрологии в сети Интернет (www.rst.gov.ru)

Содержание

Введение

Настоящий стандарт является второй частью серии стандартов «Цифровая промышленность. Унифицированная архитектура OPC». Он является руководством по применению модели безопасности открытой платформы (OPC), обеспечивающей унифицированную архитектуру (UA) механизма обмена данными в промышленных системах контроля и управления. Модель включает в себя характеристику угроз безопасности физических, аппаратных и программных сред, где предусмотрена работа OPC UA.

Настоящий стандарт дает определение общих терминов безопасности, которые используются в этой и других частях серии стандартов OPC UA. В нем дается обзор функций безопасности, а также содержатся ссылки на сервисы, сопоставления и профили, которые нормативно определены в других частях серии стандартов OPC UA. Настоящий стандарт содержит рекомендации по наилучшей практике обеспечения безопасности. Большое внимание уделяется защите данных, которыми обмениваются приложения, определяющие протокол связи OPC UA. Настоящий стандарт предназначен для разработчиков клиентских или серверных приложений OPC UA, а также для пользователей, использующих функции безопасности, предоставляемые OPC UA.

НАЦИОНАЛЬНЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

Цифровая промышленность

УНИФИЦИРОВАННАЯ АРХИТЕКТУРА OPC

Часть 2

Модель безопасности

Digital industry.
OPC unified architecture.
Part 2. Security model

Дата введения — 2025—02—01

1 Область применения

В настоящем стандарте представлены концепции и обзор унифицированной архитектуры OPC (OPC UA). Настоящий стандарт разработан в развитие остальных частей многокомпонентного набора документов по архитектуре OPC. Настоящий стандарт содержит объяснение остальных частей серии стандартов OPC UA.

В настоящем стандарте описана модель безопасности OPC Unified Architecture (OPC UA); описаны угрозы безопасности физических, аппаратных и программных сред, в которых, как ожидается, будет работать OPC UA; описано, как OPC UA полагается на другие стандарты безопасности.

В настоящем стандарте дано определение общих терминов безопасности, которые используются в этой и других частях серии стандартов OPC UA. В нем дается обзор функций безопасности, которые указаны в других частях серии стандартов OPC UA. В нем содержатся ссылки на сервисы, сопоставления и профили, которые нормативно определены в других частях серии стандартов OPC UA. В нем содержатся предложения или рекомендации по наилучшей практике обеспечения безопасности. Любая кажущаяся двусмысленность между этой частью и одной из других нормативных частей не отменяет и не уменьшает требования, указанные в другой нормативной части.

Поскольку OPC UA определяет протокол связи, основные нормы относятся к защите данных, которыми обмениваются приложения. Настоящий стандарт предназначен для разработчиков клиентских или серверных приложений OPC UA, а также для конечных пользователей, которым требуется использовать нормы различных функций безопасности, предоставляемых OPC UA. В нем содержатся положения, применимые при развертывании систем. Эти положения носят общий характер, поскольку детали будут зависеть от фактической реализации приложений OPC UA и решений, принятых для обеспечения безопасности сайта.

2 Нормативные ссылки

В настоящем стандарте использованы нормативные ссылки на следующие стандарты:

ГОСТ Р 70988 Система стандартов в цифровой промышленности. Основные положения. Общие требования к системе

ГОСТ Р 70992 Цифровая промышленность. Интеграция и интероперабельность систем. Термины и определения

Издание официальное

ГОСТ Р 71806 Цифровая промышленность. Унифицированная архитектура OPC. Часть 1. Обзор и концепции

ГОСТ Р 71809 Цифровая промышленность. Унифицированная архитектура OPC. Часть 4. Сервисы

ГОСТ Р 71810 Цифровая промышленность. Унифицированная архитектура OPC. Часть 5. Информационная модель

ГОСТ Р 71811 Цифровая промышленность. Унифицированная архитектура OPC. Часть 6. Сопоставления

Примечание — При пользовании настоящим стандартом целесообразно проверить действие ссылочных стандартов в информационной системе общего пользования — на официальном сайте Федерального агентства по техническому регулированию и метрологии в сети Интернет или по ежегодному информационному указателю «Национальные стандарты», который опубликован по состоянию на 1 января текущего года, и по выпускам ежемесячного информационного указателя «Национальные стандарты» за текущий год. Если заменен ссылочный стандарт, на который дана недатированная ссылка, то рекомендуется использовать версию этого стандарта с учетом всех внесенных в данную версию изменений. Если заменен ссылочный стандарт, на который дана датированная ссылка, то рекомендуется использовать версию этого стандарта с указанным выше годом утверждения (принятия). Если после утверждения настоящего стандарта в ссылочный стандарт, на который дана датированная ссылка, внесено изменение, затрагивающее положение, на которое дана ссылка, то это положение рекомендуется применять без учета данного изменения. Если ссылочный стандарт отменен без замены, то положение, в котором дана ссылка на него, рекомендуется применять в части, не затрагивающей эту ссылку.

3 Термины, определения и сокращения

3.1 Термины и определения

В настоящем стандарте применены следующие термины с соответствующими определениями:

3.1.1 **авторизация** (authorization): Возможность предоставления доступа к системному ресурсу.

Примечание — Авторизация доступа к ресурсам должна основываться на принципе «нужно знать». Важно, чтобы доступ в системе был ограничен.

3.1.2 **алгоритм обмена ключами** (key exchange algorithm): Протокол, используемый для установления безопасного канала связи между двумя сущностями в незащищенной среде, при котором обе сущности применяют определенный алгоритм для безопасного обмена секретными ключами, которые используются для защиты связи между ними.

Примечание — Типичным примером алгоритма обмена ключами является протокол SSL Handshake Protocol, указанный в SSL/TLS.

3.1.3 **асимметричная криптография** (asymmetric cryptography): Криптографический метод, использующий пару ключей, один из которых называется закрытым ключом и хранится в секрете, а другой называется открытым ключом, который становится доступным.

Примечание — Асимметричная криптография известна как «криптография с открытым ключом». В алгоритме асимметричного шифрования, когда объект «А» требует конфиденциальности данных, отправляемых объекту «В», объект «А» шифрует данные с помощью открытого ключа, предоставленного объектом «В». Только объект «В» имеет соответствующий закрытый ключ, необходимый для расшифровки данных.

В алгоритме асимметричной цифровой подписи, когда объекту «А» требуется целостность сообщения или для обеспечения аутентификации данных, отправляемых объекту «В», объект «А» использует свой закрытый ключ для подписи данных. Для проверки подписи объект «В» использует соответствующий открытый ключ, предоставленный объектом «А». В алгоритме асимметричного согласования ключей объекты «А» и «В» отправляют друг другу свой собственный открытый ключ. Затем, согласно ГОСТ Р 71806, каждый использует свой собственный закрытый ключ и открытый ключ другого для вычисления нового значения ключа.

3.1.4 **асимметричная подпись** (asymmetric signature): Механизм, используемый в асимметричной криптографии для подписания данных закрытым ключом субъекта и проверки подписи данных связанным с ним открытым ключом.

3.1.5 **асимметричное шифрование** (asymmetric encryption): Механизм, используемый в асимметричной криптографии для шифрования данных с помощью открытого ключа субъекта и для расшифровки данных с помощью соответствующего закрытого ключа.

3.1.6 **аудируемость** (auditability): Цель безопасности, которая гарантирует, что любые действия или активность в системе могут быть зарегистрированы.

3.1.7 **аудит** (auditing): Отслеживание действий и мероприятий в системе, включая мероприятия, связанные с безопасностью, где записи аудита использованы для анализа и проверки работы системы.

3.1.8 **аутентификация** (authetification): Цель безопасности, которая гарантирует, что личность субъекта, такого как клиент, сервер или пользователь, проверена.

3.1.9 **безопасность транспортного уровня** (transport layer security): Стандартный протокол для создания защищенных каналов в сетях на базе IP.

3.1.10 **группа безопасности** (security group): Издатель и подписчики, использующие общий контекст безопасности.

3.1.11 **доступность** (aviability): Цель безопасности, которая гарантирует, что система работает нормально, то есть никакие сервисы не были скомпрометированы таким образом, чтобы стать недоступными.

3.1.12 **закрытый ключ** (privat key): Секретный компонент пары криптографических ключей, используемых для асимметричной криптографии.

Примечание — Открытый ключ и закрытый ключ всегда генерируются как пара, если один из них обновляется, то и другой также должен быть обновлен.

3.1.13 **защищенный канал** (secure channel): В OPC UA канал связи, установленный между клиентом и сервером OPC UA, которые аутентифицировали друг друга с помощью определенных сервисов OPC UA и для которых были согласованы и применены параметры безопасности.

3.1.14 **инфраструктура открытого ключа** (public key infrastructure): Набор аппаратных средств, программного обеспечения, людей, политик и процедур, необходимых для создания, управления, хранения, распространения и отзыва сертификатов, основанных на асимметричной криптографии.

Примечание — Основными функциями PKI являются регистрация пользователей и выдача их сертификатов с открытым ключом, отзыв сертификатов при необходимости и архивирование данных, необходимых для проверки сертификатов в более позднее время. Пары ключей для обеспечения конфиденциальности данных могут генерироваться центром сертификации (ЦС); согласно ГОСТ Р 70988, от владельца закрытого ключа требуется генерировать свою собственную пару ключей, так как это повышает безопасность, поскольку закрытый ключ никогда не будет передан.

3.1.15 **код аутентификации сообщения** (message authentification code; MAC): Короткий фрагмент данных, полученный в результате работы алгоритма, использующего секретный ключ (см. симметричная криптография) для хэширования сообщения, с помощью которого получатель сообщения может проверить его на предмет изменения путем вычисления MAC, который должен быть идентичным при использовании того же сообщения и секретного ключа.

3.1.16 **конфиденциальность** (confidentiality): Цель безопасности, обеспечивающая защиту данных от прочтения непреднамеренными лицами.

3.1.17 **криптография** (cryptogrphy): Преобразование ясной, осмысленной информации в зашифрованную, неразборчивую форму с помощью алгоритма и ключа.

3.1.18 **неотрицание** (non-repudiation): Убедительное и существенное доказательство личности подписавшего сообщение и целостности сообщения, достаточное для того, чтобы сторона не смогла успешно опровергнуть первоначальную отправку или доставку сообщения и целостность его содержимого.

3.1.19 **область** (scope): Утверждение, представляющее подмножество ресурса.

Примечание — Область, обозначающая набор узлов, управляемых сервером.

3.1.20 **ограничение доступа** (access restriction): Ограничение условий, при которых на узле выполнена операция, такая как чтение, запись или вызов.

Примечание — Операции выполняются на узле только в том случае, если клиент имеет необходимые разрешения и соблюдает все ограничения доступа.

3.1.21 **одноразовый номер** (nonce): Случайное число, которое используется один раз, как правило, алгоритмами, генерирующими ключи безопасности.

3.1.22 **орган сертификации** (certificate authority): Организация, которая может выпускать сертификаты, также известная как ЦС.

Примечание — Сертификат удостоверяет принадлежность открытого ключа названному субъекту сертификата. Это позволяет другим (доверяющим сторонам) полагаться на подписи или утверждения, сделанные с помощью закрытого ключа, соответствующего сертифицированному открытому ключу. В этой модели доверительных отношений ЦС является доверенной третьей стороной, которой доверяют как субъект (владелец) сертификата, так и сторона, полагающаяся на сертификат. ЦС характерны для многих схем инфраструктуры открытых ключей (PKI).

3.1.23 **открытый ключ** (public key): Публично раскрываемый компонент пары криптографических ключей, используемых для асимметричной криптографии.

Примечания
1 См. [1].
2 Открытый ключ и закрытый ключ всегда генерируются как пара, если один из них обновляется, другой также должен быть обновлен.

3.1.24 **подпись сообщения** (message signature): Цифровая подпись, используемая для обеспечения целостности сообщений, отправляемых между двумя сущностями.

Примечание — Существует несколько способов создания и проверки подписей сообщений; они разделяются на симметричные и асимметричные подходы.

3.1.25 **поставщик идентификационных данных** (integrity provider): Сервер, который проверяет учетные данные, предоставленные принципом безопасности, и возвращает маркер, который передан связанным с сервисом авторизации.

3.1.26 **разрешение** (permison): Право на выполнение операции, такой как чтение, запись или вызов на узле.

3.1.27 **ресурс** (resource): Защищенный объект, к которому приложение должно получить доступ.

Примечание — Ресурс — это, как правило, сервер.

3.1.28 **Ривест-Шамир-Адлеман** (Rivest-Shamir-Adleman): Алгоритм асимметричной криптографии, изобретенный в 1977 году Роном Ривестом, Ади Шамиром и Леонардом Адлеманом.

Примечание — См. [1].

3.1.29 **роль** (role): Функция, которую принимает на себя клиент при обращении к серверу.

Примечание — Роль может означать конкретную рабочую функцию, например оператора или инженера.

3.1.30 **сертификат X.509** (X.509 sertificate): Сертификат в одном из форматов, определенных X.509 v1, 2 или 3.

Примечание — Сертификат X.509 содержит последовательность элементов данных и имеет цифровую подпись, вычисленную на этой последовательности. OPC UA использует только V3.

3.1.31 **сертификат экземпляра приложения** (application instance certificate): Сертификат отдельного экземпляра приложения, который был установлен на отдельном хосте.

Примечание — Различные установки одного программного продукта будут иметь разные сертификаты экземпляра приложения [2]. Использование сертификата см. примечание 1 к 3.1.23. Возможно существование нескольких экземпляров одного и того же приложения, запущенных одновременно на нескольких компьютерах. Не рекомендуется использование на одном компьютере экземпляра приложения для целей, выходящих за рамки того, что описано в спецификации. Это может значительно снизить безопасность, обеспечиваемую сертификатом экземпляра.

3.1.32 **симметричная криптография** (symmetric cryptography): Раздел криптографии, включающий алгоритмы, которые используют один и тот же ключ для двух различных шагов алгоритма (например, шифрование и дешифрование, создание подписи и проверка подписи).

Примечание — См. ГОСТ Р 71809.

3.1.33 **симметричная подпись** (simmetrie signature): Механизм, используемый в симметричной криптографии для подписания данных с помощью криптографического ключа, совместно используемого двумя сущностями.

Примечание — Подпись затем проверяется путем повторной генерации подписи для данных и сравнения этих двух подписей. Если они одинаковы, то подпись действительна, в противном случае ключ либо данные отличаются.

3.1.34 **симметричное шифрование** (symmetric encryption): Механизм, используемый в симметричной криптографии для шифрования и дешифрования данных с помощью криптографического ключа, общего для двух субъектов.

3.1.35 **система управления кибербезопасностью** (cyber security management System): Программа, разработанная организацией для поддержания безопасности всех активов организации на установленном уровне конфиденциальности, целостности и доступности, независимо от того, находятся ли они на стороне бизнеса или на стороне промышленной автоматизации и систем управления организации.

3.1.36 **сервис авторизации** (authorization service): Сервер, который проверяет запрос на доступ к ресурсу. Сервер имеет право вернуть токен доступа, предоставляющий доступ к ресурсу.

Примечание — В других стандартах сервис авторизации также называется STS [Сервис токенов безопасности (Security Token Service)].

3.1.37 **сервис ключей безопасности** (security key service): Сервер, принимающий токены доступа, выданные сервисом авторизации, и возвращающий ключи безопасности, которые используются для доступа к указанному ресурсу.

Примечание — Ключи используются для операций криптографии, таких как шифрование или расшифровка сообщений, отправляемых в потоке PubSub.

3.1.38 **список доверия** (trust list): Список сертификатов, которым приложение OPC UA доверяет.

3.1.39 **токен доступа** (access token): Документ с цифровой подписью, подтверждающий, что субъект имеет право на доступ к ресурсу.

Примечание — В документе указаны имя субъекта и ресурс, к которому осуществляется доступ.

3.1.40 **утверждение** (claim): Утверждение в маркере доступа, подтверждающее информацию о субъекте, которую сервис авторизации считает истинной.

Примечание — Утверждения, включающие имя пользователя, электронную почту и роли, предоставленные субъекту.

3.1.41 **функция хэширования** (hash function): Алгоритм, такой как SHA-1, для которого вычислительно невозможно найти объект данных, который сопоставляется с заданным результатом хэширования (свойство «односторонности»), либо два объекта данных, которые сопоставляются с одним и тем же результатом хэширования (свойство «отсутствия коллизий»).

Примечание — См. [1].

3.1.42 **хешированный код аутентификации сообщения** (hash message authentication code): MAC, сгенерированный с помощью итеративной хэш-функции.

3.1.43 **хранилище сертификатов** (certificate store): Постоянное место, где хранятся сертификаты и списки отзыва сертификатов (CRL).

Примечание — Это резидентная файловая структура на диске или на платформах Windows, или это местоположение реестра Windows.

3.1.44 **целостность** (Integrity): Цель безопасности, которая гарантирует, что информация не была изменена или уничтожена несанкционированным образом.

Примечание — См. [1].

3.1.45 **цифровая подпись** (digital signature): Значение, вычисляемое с помощью криптографического алгоритма и добавляемое к данным таким образом, что любой получатель данных может использовать подпись для проверки происхождения и целостности данных.

3.1.46 **экземпляр приложения** (application instance): Индивидуальная установка программы, запущенной на одном компьютере.

3.2 Сокращения

В настоящем стандарте применены следующие сокращения:

AES — расширенный стандарт шифрования (Advanced Encryption Standard);
CA — центр сертификации (Certificate Authority);
CRL — список отзыва сертификатов (Certificate Revocation List);
CSMS — система управления кибербезопасностью (Cyber Security Management System);
DNS — система доменных имен (Domain Name System);
DSA — алгоритм цифровой подписи (Digital Signature Algorithm);
ECDH — эллиптическая кривая Диффи-Хеллмана (Elliptic Curve Diffie-Hellman);
ECDSA — алгоритм цифровой подписи с эллиптической кривой (Elliptic Curve Digital Signature Algorithm);
HMAC — код аутентификации сообщений на основе хэша (Hash-based Message Authentication Code);
JSON — объектная нотация JavaScript (JavaScript Object Notation):
JWT — веб-токен JSON (JSON Web Token);
NIST — Национальный институт стандартов и технологий (National Institute of Standard and Technology);
PKI — инфраструктура открытых ключей (Public Key Infrastructure);
RSA — алгоритм с открытым ключом для подписи или шифрования, Ривест, Шамир, Адлеман (Public key algorithm for signing or encryption, Rivest, Shamir, Adleman);
SHA — алгоритм безопасного хэширования (существует несколько версий SHA1, SHA256,...) (Secure Hash Algorithm (Multiple versions exist SHA1, SHA256,…);
SKS — сервер ключей безопасности (Security Key Server);
SOAP — простой протокол доступа к объектам (Simple Object Access Protocol);
SSL — уровень защищенных сокетов (Secure Sockets Layer);
TLS — безопасность транспортного уровня (Transport Layer Security);
UA — унифицированная архитектура (Unified Architecture);
UACP — протокол соединения унифицированной архитектуры (Unified Architecture Connection Protocol);
UADP — протокол датаграмм унифицированной архитектуры (Unified Architecture Datagram Protocol);
URI — единый идентификатор ресурса (Uniform Resource Identifier);
XML — расширяемый язык разметки (Extensible Mark-up Language).

4 Архитектура безопасности OPC UA

4.1 Среда безопасности OPC UA

OPC UA — это протокол, используемый между компонентами при эксплуатации промышленного объекта на нескольких уровнях: от управления предприятием на высоком уровне до непосредственного управления технологическим процессом устройства на низком уровне. Использование OPC UA для управления предприятием предполагает взаимодействие с клиентами и поставщиками. Нарушение связи в системе управления технологическим процессом приводит к финансовым потерям, влияя на безопасность сотрудников и общества, или наносит ущерб окружающей среде согласно ГОСТ Р 70988 и ГОСТ Р 70992.

OPC UA назначается для использования в различных операционных средах с различными предположениями об угрозах и доступности, а также с различными политиками безопасности и режимами обеспечения соблюдения. OPC UA предоставляет гибкий набор механизмов безопасности, см. ГОСТ Р 71809.

На рисунке 1 представлена сводная таблица, показывающая комбинацию таких сред. Часть OPC UA приложений находится на одном хосте и может быть легко защищена от внешних атак. Часть OPC приложений общего доступа находится на разных хостах в одной операционной сети и может быть защищена с помощью средств защиты, которые отделяют операционную сеть от внешних подключений.

Часть приложений OPC UA работает в относительно открытых средах, где управление пользователями и приложениями может быть затруднено. Другие приложения OPC UA встроены в системы управления, которые не имеют прямого электронного подключения к внешним системам.

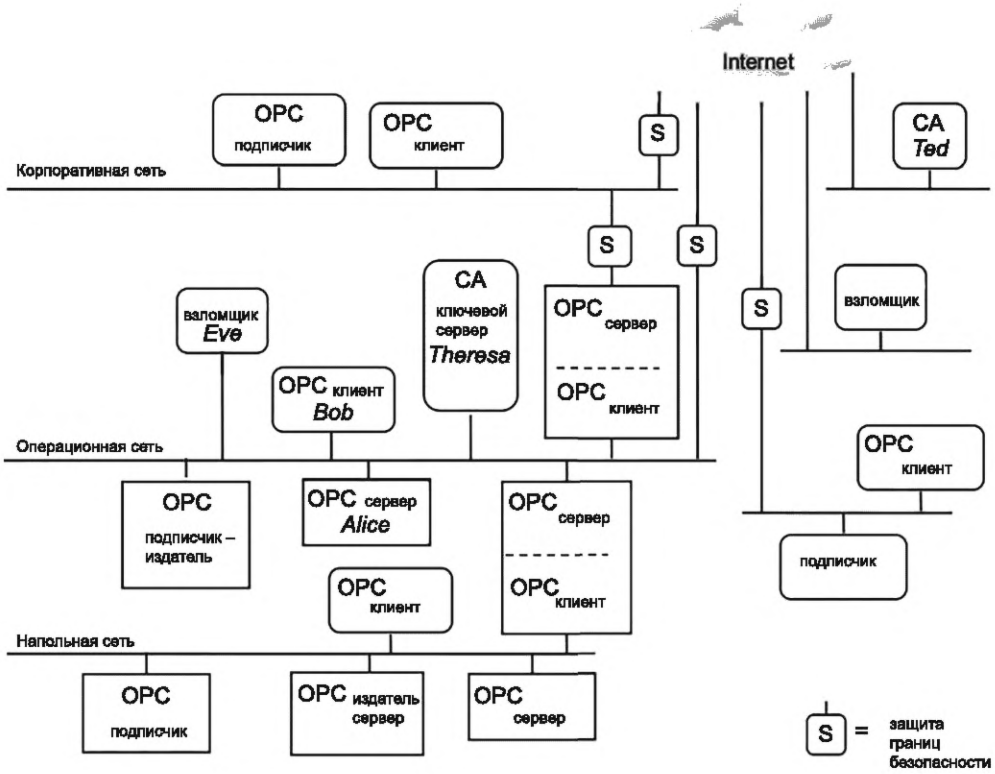

Рисунок 1 — Пример сети OPC UA

OPC UA поддерживает множество протоколов и коммуникационных технологий, для которых требуются различные уровни безопасности и различная инфраструктура безопасности. Например, оба клиента — взаимодействие сервера и издателя с подписчиком показано на рисунке 1.

4.2 Цели обеспечения безопасности

4.2.1 Обзор

Безопасность информационной системы снижает риск ущерба от атак с помощью выявления угроз для системы, выявления уязвимостей системы к этим угрозам и принятия контрмер. Эти контрмеры непосредственно уменьшают уязвимости, противодействуют угрозам или восстанавливаются после успешных атак.

С целью обеспечения безопасности систем промышленной автоматизации требуется решение задач, постоянных несмотря на меняющийся набор угроз для систем. Они описаны в 5.1 и 5.2, чтобы согласовать решение этих задач с функциями OPC UA. В разделе 6 содержатся дополнительные рекомендации по наилучшей практике для разработчиков клиентов и серверов или тех, кто развертывает приложения OPC UA.

4.2.2 Аутентификация

Клиенты, серверы и пользователи должны подтверждать свою личность. Аутентификация основывается на характеристиках того, чем является объект, что он имеет или что ему известно.

4.2.3 Авторизация

Доступ к ресурсам для чтения, записи или выполнения должен быть разрешен только для тех объектов, которые нуждаются в таком доступе в соответствии с требованиями системы. Авторизация может быть грубой, например разрешать или запрещать клиенту доступ к серверу, так и тонкой, напри-

мер разрешать ограниченные действия с определенными элементами информации конкретным пользователям. Степень детализации системы частично зависит от функциональности, поддерживаемой сервером, но в целом авторизация должна предоставляться на основе принципа «необходимо знать», т. е. пользователю должен быть предоставлен доступ только к информации, необходимой ему для выполнения функции, которую он выполняет.

4.2.4 Конфиденциальность

Данные защищены от пассивных атак, таких как подслушивание, независимо от того, передаются ли они, хранятся ли они в памяти. Для обеспечения конфиденциальности используются алгоритмы шифрования данных, использующие специальные секреты для защиты данных, а также механизмы аутентификации и авторизации для доступа к этому секрету.

4.2.5 Целостность

Получатели получают ту же информацию, которую отправил первоначальный отправитель, без изменения данных во время передачи.

4.2.6 Неотказуемость

Отречение — это отказ или отрицание чего-либо как действительного или истинного. Отказ от ответственности — это гарантия того, что то, что произошло на самом деле, не может быть заявлено как несуществующее. Сервис безопасности, предоставляющий такую защиту, может быть одного из двух типов:

- тот, в котором получатель данных получает и хранит информацию, подтверждающую, что данные поступили от отправителя. Это не позволяет отправителю утверждать, что он никогда не отправлял данные;
- тот, в котором отправитель данных получает подтверждение того, что данные были получены получателем по назначению.

4.2.7 Проверяемость

Действия, предпринимаемые системой, должны регистрироваться, чтобы предоставить доказательства заинтересованным сторонам:

- что эта система работает должным образом (отслеживаются успешные действия);
- которые идентифицируют инициатора определенных действий (отслеживается активность пользователя);
- что попытки взлома системы были отклонены (отслеживаются неудачные действия).

4.2.8 Доступность

Доступность снижается, когда отключается выполнение программного обеспечения, которое необходимо запустить, или когда программное обеспечение или система связи перегружены обработкой входных данных. Нарушение доступности в OPC UA может проявляться, например в снижении производительности подписки или невозможности добавления сеансов.

4.3 Угрозы безопасности систем общего доступа OPC

4.3.1 Обзор

OPC UA предоставляет контрмеры для противодействия угрозам безопасности передаваемой информации согласно ГОСТ Р 71806. В 4.3 перечислены известные на данный момент угрозы для сред, в которых будет развернут OPC UA, а в 5.1 приведено соответствие этих угроз функциям OPC UA.

4.3.2 Отказ в обслуживании

4.3.2.1 Обзор

Отказ в обслуживании — это предотвращение авторизованного доступа к системному ресурсу или задержка системных операций и функций. Это может быть вызвано различными направлениями атак, включая переполнение сообщений, истощение ресурсов и сбои в работе приложений. Каждый из этих факторов описан отдельно. Отказ в обслуживании влияет на доступность. См. 5.1.2 для согласования этой угрозы.

4.3.2.2 Поток сообщений

Для взаимодействия клиент-сервер злоумышленник может отправить большой объем сообщений или одно сообщение, содержащее большое количество запросов, с целью подавления работы сервера OPC UA или зависимых компонентов, таких как центральный процессор, стек TCP/IP, операционная система или файловая система.

Атаки с использованием флуда могут проводиться на нескольких уровнях, включая OPC UA, SOAP [HTTP] или TCP. Атаки с использованием флуда сообщений могут использовать как правильно сфор-

мированные, так и искаженные сообщения. В первом случае злоумышленником может быть злоумышленник, использующий законный клиент для заполнения сервера запросами. Существуют два случая, в одном из которых клиент не имеет сеанса связи с сервером, а в другом — имеет. Поток сообщений может нарушить возможность установления сеансов OPC UA или прервать существующий сеанс. Во втором случае злоумышленник может использовать вредоносный клиент, который загружает сервер OPC UA ошибочными сообщениями, чтобы истощить ресурсы сервера.

Для PubSub злоумышленник может отправлять большой объем сообщений с наборами данных с целью взлома подписчика, промежуточного программного обеспечения или зависимых компонентов, таких как процессор, стек TCP/IP, операционная система или файловая система. Атаки с использованием потока сообщений могут проводиться на нескольких уровнях, включая OPC UA, UDP, AMQP и MQTT.

Как и в случае клиент-серверных атак, атаки с использованием потока сообщений PubSub могут использовать как правильно сформированные, так и неправильно сформированные сообщения. Для правильно сформированных сообщений злоумышленником может быть тот, в котором издатель не является членом SecurityGroup, и тот, в котором он является членом. Для получения некорректных сообщений злоумышленник может использовать вредоносного издателя, который заполняет сеть некорректными сообщениями, чтобы истощить ресурсы системы. В целом переполнение сообщениями может нарушить возможность взаимодействия с объектом OPC UA и привести к отказу в обслуживании.

4.3.2.3 Исчерпание ресурсов

Злоумышленник может отправить ограниченное количество сообщений, которые позволяют получить доступ к системному ресурсу. Команды обычно действительны, но каждая из них использует определенный ресурс, в результате чего один клиент получает все ресурсы, блокируя доступ действительных клиентов к серверу. Например, на сервере, на котором доступно только 10 сеансов, злоумышленник, использующий законный клиент, может получить доступ ко всем 10 сеансам. Или вредоносный клиент может попытаться открыть 10 защищенных каналов, фактически не завершив процесс.

Атаки, связанные с исчерпанием ресурсов, не выполняются аналогичным образом для сообщений PubSub, поскольку ни сеанс, ни ресурсы не выделяются. Для сообщений PubSub издатель не уязвим. В режиме PubSub без посредников подписчик может с помощью фильтров обойти любые проблемы, связанные с исчерпанием ресурсов. В случае с брокером как издатель, так и подписчик должны подключиться к брокеру. Хотя издатель и подписчик не являются напрямую зависимыми (как в случае без брокера), брокер является зависимым. Информация о взаимодействии с брокером не является частью OPC UA, а определяется протоколом брокера.

4.3.2.4 Сбой приложения

Злоумышленник может отправить специальное сообщение, которое приведет к сбою приложения. Обычно это происходит из-за известной проблемы в стеке или приложении. Эти системные ошибки могут позволить клиенту выдать команду, которая приведет к сбою сервера. В качестве альтернативы сервер должен иметь возможность ответить на сообщение ответом, который приведет к сбою клиента. Злоумышленником также может быть издатель, который отправляет сообщение, которое может привести к сбою работы подписчиков.

4.3.3 Подслушивание

Подслушивание — это несанкционированное разглашение конфиденциальной информации, которое может непосредственно привести к критическому нарушению безопасности или быть использовано в последующих атаках. Если злоумышленник скомпрометировал базовую операционную систему или сетевую инфраструктуру, то он может иметь возможность записывать и перехватывать сообщения. Характеристики клиента и сервера должны позволять восстанавливать работоспособность скомпрометированной операционной системы.

Прослушивание напрямую влияет на конфиденциальность, и, если не обеспечена безопасность установления сеанса связи, на аутентификацию и авторизацию. Это также косвенно угрожает всем другим целям безопасности. См. 5.1.3 для устранения этой угрозы.

4.3.4 Подделка сообщений

Это действие включает в себя подделку идентификационных данных (пользователя, приложения, процесса и т. д.). Злоумышленник может подделать сообщения от клиента, сервера или издателя, в которых сообщения подделываются таким образом, чтобы они могли выглядеть как исходящие от приложения, отличного от отправляющего приложения или процесса. Подмена может происходить на нескольких уровнях стека протоколов.

Подделывая сообщения от клиента, сервера или издателя, злоумышленники могут выполнять несанкционированные операции и избегать обнаружения своих действий.

Подделка сообщений влияет на целостность и авторизацию. См. 5.1.4 для устранения этой угрозы.

4.3.5 Изменение сообщения

Сетевой трафик и сообщения прикладного уровня могут быть перехвачены или изменены и перенаправлены на серверы, клиентам и подписчикам OPC UA. Изменение сообщения может привести к несанкционированному доступу к системе.

Изменение сообщения влияет на целостность, авторизацию, возможность проверки, неотзывчивость и аутентификацию при установлении сеанса связи/безопасного канала. См. 5.1.5 для устранения этой угрозы.

4.3.6 Воспроизведение сообщения

Сетевой трафик и действительные сообщения прикладного уровня могут быть перехвачены и повторно отправлены OPC UA на серверы, клиентам и подписчикам на более позднем этапе без изменений. Злоумышленник может дезинформировать пользователя или отправить правильную команду, например открыть клапан, но в неподходящее время, что может привести к повреждению или потере имущества. Злоумышленник может попытаться установить сеанс, используя записанный сеанс.

Воспроизведение сообщения влияет на авторизацию и во время сеанса/ установления защищенного канала. См. 5.1.6 для устранения этой угрозы.

4.3.7 Искаженные сообщения

Злоумышленник может создавать различные сообщения с неправильной структурой (искаженный XML, SOAP, двоичный код UA и т. д.) или значения данных и отправлять их клиентам, на серверы или абонентам.

Клиент, сервер или подписчик OPC UA могут некорректно обрабатывать сообщения с искаженным форматом, выполняя несанкционированные операции или обрабатывая ненужную информацию. Это может привести к отказу в обслуживании или ухудшению качества обслуживания, включая завершение работы приложения, или, в случае встроенных устройств, к полному сбою. В худшем случае злоумышленник может использовать искаженные сообщения в качестве предварительного шага для многоуровневой атаки, чтобы получить доступ к базовой системе приложения OPC UA. Искаженные сообщения влияют на целостность и доступность. См. 5.1.7 для устранения этой угрозы.

4.3.8 Профилирование сервера

Злоумышленник пытается установить личность, тип, версию программного обеспечения или поставщика сервера или клиента, чтобы использовать знания о конкретных уязвимостях этого продукта для проведения более навязчивой или разрушительной атаки. Злоумышленник может создать профиль объекта, отправив ему корректные или недопустимые сообщения в форматированном виде, и попытаться определить тип объекта по шаблону его обычных ответов и ответов об ошибках.

Профилирование сервера косвенно влияет на все цели безопасности. См. 5.1.8 для устранения этой угрозы.

4.3.9 Перехват сеанса

Злоумышленник может использовать информацию (полученную путем прослушивания сообщения или путем угадывания) о запущенном сеансе, установленном между двумя приложениями, для передачи манипулируемых сообщений (с действительной информацией о сеансе), которые позволяют ему или ей перехватить сеанс у авторизованного пользователя.

Злоумышленник может получить несанкционированный доступ к данным или выполнить несанкционированные операции. Перехват сеанса влияет на все цели безопасности. Для устранения этой угрозы см. 5.1.9.

4.3.10 Мошеннический сервер

Злоумышленник создает вредоносный сервер OPC UA или устанавливает несанкционированный экземпляр подлинного сервера OPC UA в системе. Несанкционированный сервер может попытаться выдать себя за законный пользовательский сервер или он может просто появиться как новый сервер в системе.

Клиент OPC может раскрыть необходимую информацию. Сервер-мошенник влияет на все цели безопасности, за исключением целостности и неотзывчивости. Для устранения этой угрозы см. 5.1.10.

4.3.11 Сторонний издатель

Сторонний издатель — это злоумышленник, который создает вредоносный OPC UA издатель или устанавливает несанкционированный экземпляр подлинного OPC UA издателя в системе. Сторонний издатель может попытаться выдать себя за законного издателя общего доступа или он может просто появиться в системе как новый издатель.

Мошеннический издатель влияет на все цели безопасности, за исключением целостности и неотзывчивости. Для устранения этой угрозы см. 5.1.10.

4.3.12 Компрометация учетных данных пользователя

Злоумышленник получает учетные данные пользователя, такие как имена пользователей, пароли, сертификаты или ключи, просматривая их на бумаге, на экранах или в электронных сообщениях, или взламывая их с помощью угадывания или использования автоматизированных средств, таких как средства взлома паролей.

Неавторизованный пользователь может запустить систему и получить к ней доступ, чтобы получить всю информацию и внести изменения в управление и данные, которые могут нанести ущерб работе предприятия или информации. После использования скомпрометированных учетных данных все последующие действия могут казаться законными.

Скомпрометированные учетные данные пользователя влияют на аутентификацию, авторизацию и конфиденциальность. Для устранения этой угрозы см. 5.1.11.

4.3.13 Отказ от ответственности

Возможна не прямая атака на коммуникации, а влияние на доверие, которое изменяется после коммуникации. Отказ от предоставления данных вызывает проблемы с доверием как у отправителя, так и у получателя данных.

Отказ от предоставления данных влияет на отказ от ответственности. Для устранения этой угрозы см. 5.1.12.

4.4 Связь OPC UA с безопасностью сайта

OPC UA security работает в рамках общей системы управления кибербезопасностью (CSMS) сайта. На сайтах часто есть CSMS, которая отвечает за политику и процедуры безопасности, персонал, обязанности, аудит и физическую безопасность. CSMS устраняет угрозы, которые включают в себя те, которые были описаны в 4.3. Они также анализируют риски для безопасности и определяют, какие средства контроля безопасности необходимы сайту.

В результате средства контроля безопасности обычно реализуют стратегию «глубокой защиты», которая обеспечивает несколько уровней защиты и признает, что ни один уровень не может защитить от всех атак. Средства защиты границ, показанные в качестве абстрактных примеров на рисунке 1, включают брандмауэры, системы обнаружения и предотвращения вторжений, средства управления подключениями удаленного доступа, а также средства управления мультимедийными устройствами и компьютерами, подключенными к системе. Меры защиты в компонентах системы должны включать в себя усиленную настройку операционных систем, управление исправлениями безопасности, антивирусные программы и запрет доступа к электронной почте в сети управления. Стандарты, которым должен соответствовать сайт, включают NERC CIP и систему стандартов OPC UA (все части), ссылки на которые приведены в разделе 2.

Требования к безопасности CSMS сайта распространяются на его интерфейсы OPC UA. То есть требования к безопасности интерфейсов OPC UA, которые развертываются на сайте, определяются сайтом, а не стандартами OPC UA. OPC UA определяет функции, которые предназначены для обеспечения соответствия. Приложения OPC UA должны соответствовать требованиям безопасности, которые будут предъявляться к сайтам, на которых они будут развернуты. Те специалисты, которые отвечают за безопасность на сайте, должны определить, как обеспечить соответствие требованиям сайта с помощью продуктов, соответствующих стандарту OPC UA.

Владелец системы, устанавливающий приложения OPC UA, должен проанализировать риски для своей безопасности и предоставить соответствующие механизмы для снижения этих рисков для достижения приемлемого уровня безопасности. OPC UA удовлетворяет широкому спектру потребностей в области безопасности, которые могут возникнуть в результате анализов. Приложения OPC UA должны быть оснащены функциями безопасности, которые доступны для дополнительного использования владельцем системы. Каждый владелец системы должен иметь возможность адаптировать решение для обеспечения безопасности, соответствующее его требованиям безопасности и экономичности, используя комбинацию механизмов, доступных в рамках серии стандартов OPC UA и внешних по отношению к OPC.

Требования к безопасности, предъявляемые к приложениям OPC UA, развернутым на сайте, определяются CSMS сайта, а не серией стандартов OPC UA. Стандарты безопасности OPC UA содержат требования, предъявляемые к приложениям OPC UA, и рекомендации о том, как OPC UA должен быть развернут на сайте, чтобы соответствовать требованиям безопасности, которые, как ожидается, будут указаны на сайте.

OPC UA устраняет угрозы, как описано в 4.3. Как описано в разделе 6, базовая версия OPC рекомендует, чтобы разработчики приложений OPC UA устраняли оставшиеся угрозы компонентам инфраструктуры, которые могут привести к компрометации операционных систем, где приложения OPC UA запущены, но OPC UA к ним не обращается.

4.5 Архитектура безопасности общего доступа OPC

4.5.1 Обзор

Архитектура безопасности OPC UA — это универсальное решение, которое позволяет реализовать необходимые функции безопасности в различных местах архитектуры приложений OPC UA. В зависимости от различных сопоставлений, описанных в ГОСТ Р 71811, задачи обеспечения безопасности решаются на разных уровнях. Архитектура безопасности OPC UA для взаимодействия клиент/сервер структурирована на прикладном уровне и коммуникационном уровне поверх транспортного уровня, как показано на рисунке 2.

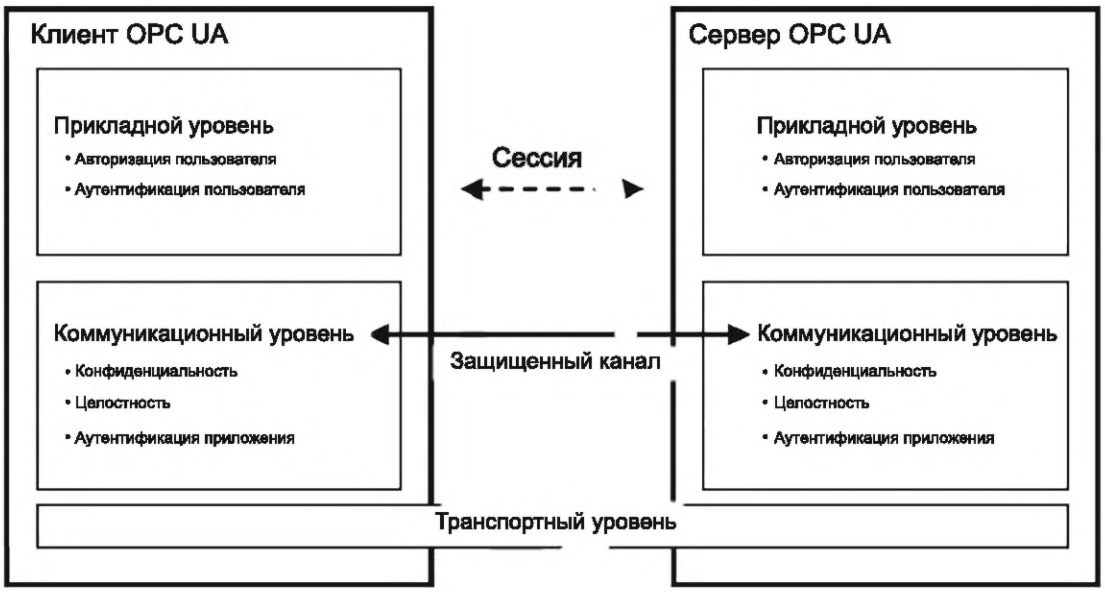

Рисунок 2 — Архитектура безопасности OPC UA — клиент/сервер

OPC UA также поддерживает архитектуру обмена данными с возможностью публикации и подписки, и архитектура безопасности для этой связи показана на рисунке 3.

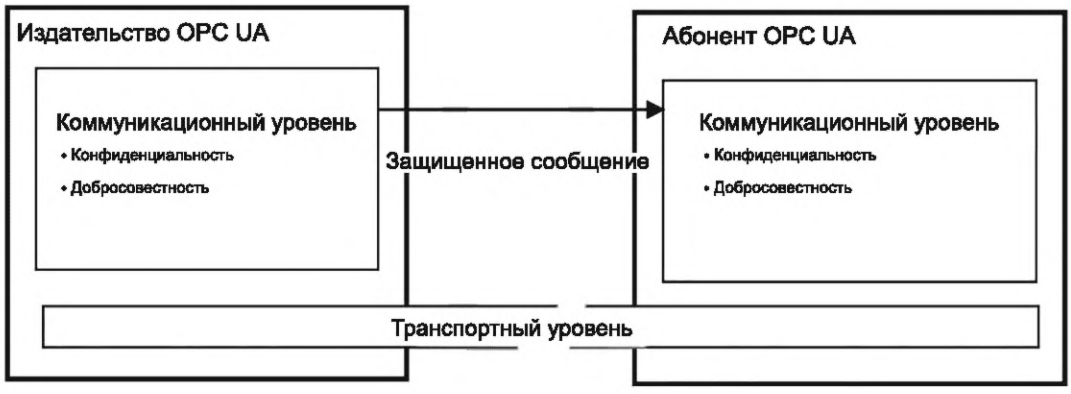

Рисунок 3 — Архитектура безопасности OPC UA — издатель-подписчик

4.5.2 Клиент-сервер

Взаимодействие клиент-сервер включает в себя возможность как сеансового, так и бессессионного взаимодействия. Рутинная работа клиентского приложения и серверного приложения по передаче информации, настроек и команд выполняется в сеансе на прикладном уровне. Прикладной уровень также управляет задачами безопасности, связанными с аутентификацией и авторизацией пользователя. Задачи безопасности, которые решаются на прикладном уровне, решаются с помощью сервиса сеанса, указанных в ГОСТ Р 71809. Сеанс на прикладном уровне взаимодействует по сети.

Защищенный канал, который создается на коммуникационном уровне, используется для обеспечения безопасной связи. Все данные сеанса передаются на коммуникационный уровень для дальнейшей обработки.

Несмотря на то, что сеанс взаимодействует по защищенному каналу и должен быть активирован перед его использованием, привязка пользователей, сеансов и защищенных каналов является гибкой. Олицетворение позволяет пользователю стать владельцем существующего сеанса.

Если защищенный канал прерывается, сеанс остается действительным в течение определенного периода времени, что позволяет клиенту восстановить подключение к сеансу по новому защищенному каналу. В противном случае сеанс закрывается по истечении срока его действия.

Коммуникационный уровень обеспечивает механизмы безопасности для обеспечения конфиденциальности, целостности и аутентификации приложений в качестве целей безопасности. Одним из важных механизмов для достижения этих целей является создание защищенного канала (см. 4.13), который используется для обеспечения безопасности связи между клиентом и сервером. Защищенный канал обеспечивает шифрование для сохранения конфиденциальности, подписи сообщений для поддержания целостности и сертификаты для аутентификации приложений. Данные, поступающие с прикладного уровня, защищены и передают «защищенные» данные на транспортный уровень. Механизмы безопасности, управляемые коммуникационным уровнем, предоставляются сервисами защищенных каналов, которые указаны в ГОСТ Р 71809.

Механизмы безопасности, предоставляемые сервисами <защищенный канал>, реализуются с помощью стека протоколов, который выбран для реализации. Сопоставления сервиса с некоторыми параметрами стека протоколов указаны в ГОСТ Р 71811, который определяет, как функции стека протоколов используются для достижения целей безопасности OPC UA.

Уровень связи представляет собой стек протоколов подключения OPC UA. OPC UA определяет альтернативные сопоставления стека, которые используются в качестве коммуникационного уровня. Эти сопоставления описаны в ГОСТ Р 71811. Если используется протокол подключения OPC UA (UACP), то обеспечена функциональность для конфиденциальности.

Целостность, аутентификация приложения и защищенный канал аналогичны спецификациям SSL/TLS, как описано в ГОСТ Р 71811.

Транспортный уровень обрабатывает передачу, прием и транспортировку данных, предоставляемых коммуникационным уровнем, чтобы пережить потерю соединений транспортного уровня (например, TCP-соединений) и возобновить их с помощью нового соединения. Коммуникационный уровень отвечает за восстановление транспортного уровня подключением на другом уровне без прерывания логического защищенного канала.

Транспортный уровень также может использоваться для обеспечения конфиденциальности и целостности с помощью протокола HTTPS, согласно ГОСТ Р 71811. HTTPS-сертификаты могут совместно использоваться (и часто используются) несколькими приложениями на платформе. Они могут быть скомпрометированы за пределами использования OPC UA. Все приложения на платформе, использующие один и тот же общий сертификат, имеют одинаковые настройки, такие как отключение SSLv2.

OPC UA обеспечивает бессессионный вызов службы (см. ГОСТ Р 71806 и ГОСТ Р 71811). Бессессионная связь обеспечивает аутентификацию пользователя. Канал связи обеспечивает конфиденциальность и целостность. Канал связи может быть защищенным каналом OPC UA (без сеанса связи). Это может быть канал связи, такой как HTTPS, который использует транспортные протоколы для обеспечения безопасности. Аутентификация и/или проверка подлинности приложения также устанавливаются с помощью токена доступа, который получается из сервиса авторизации (подробности см. в ГОСТ Р 71806).

Дополнительные сопоставления данных представлены в ГОСТ Р 71811. При обеспечении конфиденциальности и целостности данных эти сопоставления зависят от протоколов транспортного уровня. Одним из примеров являются использующие защиту веб-сокеты транспортного уровня HTTPS.

4.5.3 Публикация-подписка

4.5.3.1 Обзор

Служба <публикация-подписка> разворачивается в двух средах: в одной из которых брокер существует, а в другой — брокера нет. Подробное описание этой модели приведено в [3].

С этими двумя средами связаны вопросы безопасности, и каждая из них описана отдельно.

4.5.3.2 Без посредников

Коммуникационная модель <публикация-подписка> без посредников обеспечивает конфиденциальность и целостность. Это достигается с помощью симметричного шифрования и алгоритмов подписи. Необходимые симметричные ключи распределяются сервером ключей безопасности (SKS) (дополнительные сведения см. в [1]). SKS использует стандартную систему защиты клиент-сервер, описанную в предыдущем разделе, для установления аутентификации приложений, а также аутентификации пользователей. Такой подход позволяет всем приложениям (издателям и/или подписчикам) в группе безопасности обмениваться информацией.

Преимуществом использования общих симметричных ключей является высокая производительность, которую они обеспечивают, но недостатком является то, что для группы приложений, использующих общий симметричный ключ, все приложения в группе имеют одинаковые права. Все приложения должны доверять всем другим приложениям в группе. Любое приложение (издатель или подписчик) в группе может опубликовать сообщение, и любое приложение (издатель или подписчик) в группе может расшифровать сообщение.

Например, система может состоять из совместно используемой симметричной группы, состоящей из контроллера (издателя) и трех подписчиков (например, HMI). Контроллер публикует, а HMI-интерфейсы получают сообщения. Если один из HMI скомпрометирован, он также может начать публиковать сообщения. Два других HMI не смогут определить, что сообщение было отправлено не с контроллера. Одним из возможных решений этой ситуации могло бы быть, если бы общая симметричная группа состояла только из контроллера и одного HMI. Дополнительные группы были бы созданы для каждого HMI, тогда бы ни один HMI не смог повлиять на другие HMI. Другие возможные решения могут также включать сетевую архитектуру и сервисы, такие как одноадресная передача данных по сети с ограниченным доступом, но они выходят за рамки серии стандартов OPC UA. Конфигурация групп защиты требует тщательного рассмотрения при развертывании систем для обеспечения безопасности.

4.5.3.3 Посредник

При использовании брокера в модели <публикация-подписка> для обеспечения конфиденциальности и целостности используются те же общие симметричные ключи, что и в 4.5.3.2. Кроме того, связь с брокером может быть защищена в соответствии с правилами, установленными для брокера. Эти правила не определены в спецификации базовой модели OPC, но определены промежуточным программным обеспечением. Во многих случаях промежуточное программное обеспечение требует авторизации как издателей, так и подписчиков, прежде чем они смогут взаимодействовать с брокером. Взаимодействие с брокером может обеспечить механизмы безопасности для обеспечения конфиденциальности, целостности и аутентификации приложений или пользователей в качестве целей безопасности. Если опубликованное сообщение не защищено с помощью концепции общего симметричного ключа, содержимое сообщения становится видимым для брокера, что создает риск атак типа «человек посередине». Использование общих симметричных ключей устраняет этот риск.

4.6 Политика безопасности

Политика безопасности определяет, какие механизмы безопасности должны использоваться, и является производной от профиля безопасности (подробности см. в 4.7). Политики безопасности используются сервером для объявления того, какие механизмы он поддерживает, а клиентом — для выбора одного из них для использования с защищенным каналом, который он хочет открыть, или для подключения без использования сеанса связи, которое он хочет установить. Политики безопасности также используются при общении через публикацию-подписку. Политика безопасности включает в себя следующую информацию:

- алгоритмы подписи и шифрования,
- алгоритм получения ключа.

Выбор разрешенных политик безопасности осуществляется администратором, когда установлены приложения OPC UA. Доступные политики безопасности указаны в [2]. Позднее администратор может

также изменить или модифицировать выбор разрешенных политик безопасности в зависимости от обстоятельств.

Объявление политик безопасности обрабатывается специальными службами обнаружения, указанными в ГОСТ Р 71809. Более подробная информация о механизмах обнаружения и стратегиях объявления политик содержится в [4].

В схеме взаимодействия клиент-сервер каждый клиент должен иметь возможность выбрать политику, независимую от политики, выбранной другими клиентами.

Для шаблона связи с публикацией и подпиской политика безопасности связана с опубликованным набором данных, и все подписчики должны использовать одну и ту же политику безопасности.

Поскольку вычислительная мощность растет, алгоритмы, которые сегодня считаются безопасными, в будущем могут стать небезопасными; поэтому в приложении OPC UA требуется поддержка различных политик безопасности и создавать возможность внедрять новые по мере их появления. Список поддерживаемых политик безопасности будет обновляться на основе рекомендаций, опубликованных, например NIST. С точки зрения развертывания, требуется, чтобы периодический обзор сайта проверял, что выбранный в данный момент список профилей безопасности соответствует требуемым целям безопасности, и если это не так, то выбирается более новый набор профилей безопасности.

Существует возможность разработки новых политик безопасности для поддержки новых алгоритмов, которые повышают уровень безопасности продуктов OPC UA. Приложение «Архитектура приложений OPC UA» должно быть разработано таким образом, чтобы можно было обновлять или добавлять дополнительные криптографические алгоритмы в приложение практически без изменений кодировки.

В [2] указано несколько политик, которые идентифицируются с помощью определенного уникального URI. Для улучшения взаимодействия продуктов поставщиков серверные и издательские продукты должны реализовывать эти политики. Клиенты и подписчики должны поддерживать одни и те же политики.

4.7 Профили безопасности

Клиентские и серверные продукты OPC UA сертифицированы на соответствие профилям, определенным в [2]. Некоторые профили определяют функции безопасности, а остальные — другие функциональные возможности, не связанные с безопасностью. Профили предъявляют требования к сертифицированным продуктам, но не к способам их использования. Различные профили требуют одинакового минимального уровня безопасности. Однако в разных профилях указаны разные детали, например определено, какие алгоритмы шифрования требуются для функционирования OPC UA. Если в одном алгоритме шифрования обнаружена проблема, то на основании базовой модели OPC должна быть возможность определить новый профиль, который аналогичен, но в котором указан другой алгоритм шифрования, о котором не известно о проблеме. [2] является нормативной спецификацией профилей, но профили поддерживаются в онлайновом приложении (https://apps.opcfoundation.org/profilereporting/), что позволяет обновлять профили, особенно профили, связанные с безопасностью, в более сжатые сроки, чем это предусмотрено циклами публикации документации.

Политики относятся ко многим из тех же параметров безопасности, что и профили; однако политика определяет, какие из этих параметров следует использовать в сеансе. Политика не определяет диапазон параметров, предлагаемых продуктом, они описаны в профилях, которые он поддерживает. Эти правила включены в сертификационные испытания, связанные с приложениями OPC UA. Сертификационные испытания должны гарантировать соблюдение стандарта и поддержку соответствующих алгоритмов безопасности.

Каждый механизм безопасности в OPC UA предусмотрен в приложениях OPC UA в соответствии с профилями, которым соответствует приложение OPC UA. Механизмы безопасности могут быть развернуты на сайте. Таким образом, каждый сайт имеет все доступные функции безопасности OPC UA и выбирает, какие из них использовать для достижения своих целей в области безопасности.

Профили безопасности описывают профиль <Нет>, который используется для тестирования, но при наличии других, более безопасных доступны профили, если по умолчанию этот профиль отключен.

4.8 Настройки режима безопасности

OPC UA поддерживает выбор нескольких режимов безопасности: <Нет>, <Подписать>, <Подписать и зашифровать>. Режим безопасности <Нет> можно использовать только с профилем безопасности <Нет>. Он отключен для всех остальных профилей безопасности. Выбор <Подписать> или

<Подписать и зашифровать> зависит от CSMS; в некоторых приложениях, где конфиденциальность данных не требуется, <Подписать> может быть достаточно.

4.9 Аутентификация пользователя

Аутентификация пользователя выполняется, когда клиент передает серверу учетные данные пользователя, как указано в сервисах сеанса согласно ГОСТ Р 71809. Сервер может аутентифицировать пользователя с помощью этих учетных данных. Владелец (пользователь) сеанса изменяется с помощью службы активизации сеанса в соответствии с потребностями приложения.

Аутентификация пользователя напрямую не является частью шаблона обмена данными между публикацией и подпиской, используется как часть SKS, связанного с этим шаблоном обмена данными.

4.10 Аутентификация приложений

В OPC UA используется концепция аутентификации приложений, позволяющая приложениям, которые намереваются взаимодействовать, идентифицировать друг друга. Каждому экземпляру приложения OPC UA присваивается сертификат использования приложения, который предъявляется учреждением по защищенному каналу. Получатель сертификата проверяет, доверяет ли он сертификату, и на основании этой проверки принимает или отклоняет запрос или ответное сообщение от отправителя. Эта проверка доверия выполняется с использованием концепции списков доверия. Списки доверия реализованы в виде хранилища сертификатов, назначенном администратором. Прежде чем поместить сертификат в список доверия администратор определяет, подписан ли он, подтвержден ли и заслуживает ли доверия. В списке доверия также находятся центры сертификации (CA). Списки доверия, включающие центры сертификации, включают списки отзыва сертификатов (CRLS). OPC UA использует отраслевые стандарты, определенные другими организациями.

В OPC UA HTTPS используется для создания защищенных каналов; однако эти каналы не обеспечивают аутентификацию приложений. Если требуется аутентификация, то она должна быть основана на учетных данных пользователя (см. 4.9). Более подробная информация об аутентификации приложения содержится в ГОСТ Р 71809.

4.11 Авторизация пользователя

OPC UA обеспечивает авторизацию пользователя на основе аутентифицированного пользователя (см. 4.9). OPC UA могут самостоятельно определять, какие данные доступны и какие операции разрешены, или они могут использовать роли (см. 4.12). Профили используются для указания поддержки учетных данных пользователя, а также для ограничения или контроля доступа к адресному пространству.

4.12 Роли

OPC UA предоставляет стандартный подход для реализации безопасности на основе ролей. Серверы могут выбрать не реализовывать часть или все механизмы, определенные в ГОСТ Р 71810. Подход OPC UA назначает разрешения ролям. Затем клиентам присваиваются роли на основе информации о подключении. Роли могут быть ограничены на основании проверки подлинности пользователя приложений, режимов безопасности или средств коммуникации. Назначение ролей и ограничений зависит от конкретного приложения. Взаимодействия показаны на рисунке 4.

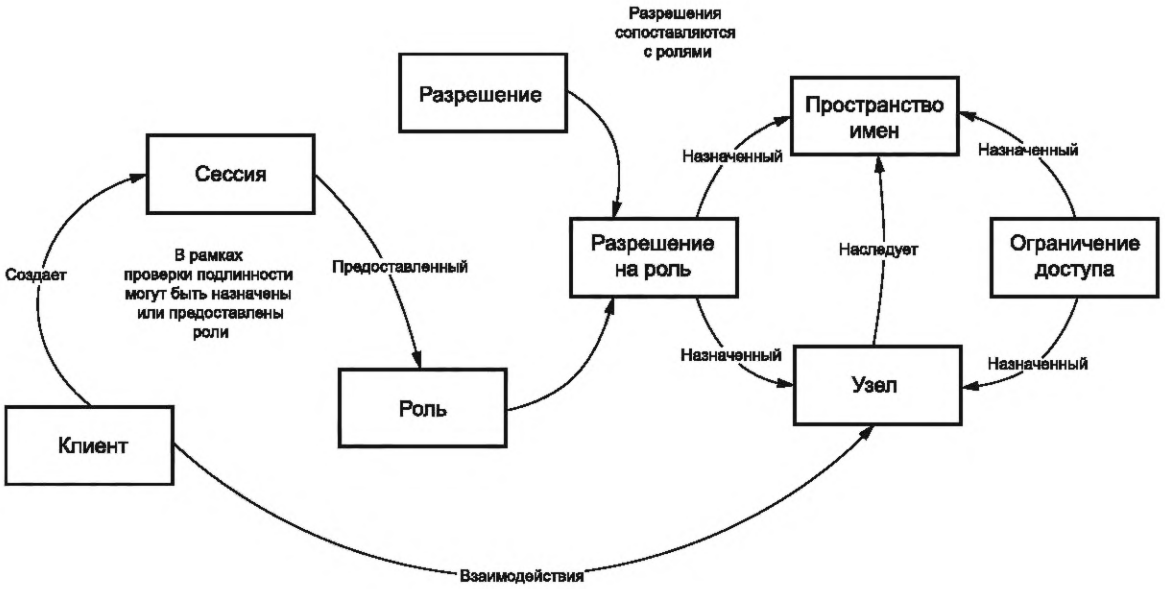

Рисунок 4 — Обзор ролей

Дополнительное описание ролей приведено в ГОСТ Р 71810.

4.13 Службы, связанные с безопасностью OPC UA

Службы безопасности OPC UA — это группа абстрактных определений служб, указанных в ГОСТ Р 71806. ГОСТ Р 71809 применяют для использования различных механизмов безопасности при связи между клиентами OPC и сервера в UA. Набор служб обнаружения согласно ГОСТ Р 71809 определяет службы, используемые клиентом OPC UA для получения информации о политиках безопасности (см. 4.6) и сертификатах, определенных серверов OPC UA.

Сервисы набора служб защищенного канала в соответствии с ГОСТ Р 71809 используются для создания защищенного канала, который отвечает за защиту сообщений, отправляемых между клиентом и сервером. Сложность создания защищенного канала заключается в том, что для этого требуется, чтобы клиент и сервер безопасно обменивались криптографическими ключами и секретной информацией в небезопасной среде, поэтому участники коммуникации применяют специальный алгоритм обмена ключами (аналогичный протоколу SSL Handshake, определяемому в SSL/TLS).

Клиент OPC UA получает политики без опасности и сертификаты сервера OPC UA с помощью вышеупомянутых сервисов обнаружения. Эти сертификаты содержат открытые ключи сервера OPC UA. Клиент OPC UA отправляет свой открытый ключ в виде сертификата и секретной информации вместе с сервисным сообщением по открытому безопасному каналу. Это сообщение защищается путем применения асимметричного шифрования с использованием открытого ключа сервера и путем создания асимметричных подписей с использованием закрытого ключа клиента. Однако сертификат отправляется в незашифрованном виде, чтобы получатель мог использовать его для проверки асимметричной подписи.

Сервер расшифровывает сообщение с помощью своего закрытого ключа и проверяет асимметричную подпись с помощью открытого ключа клиента. Секретная информация клиента OPC UA вместе с секретной информацией сервера OPC UA используется для получения набора криптографических ключей, которые используются для защиты всех последующих сообщений. Кроме того, все остальные служебные сообщения защищены симметричным шифрованием и симметричными подписями вместо асимметричных эквивалентов.

Сервер отправляет клиенту свою секретную информацию в ответ, чтобы клиент мог получить тот же набор криптографических ключей.

Поскольку клиенты и серверы имеют одинаковый набор криптографических ключей, они могут безопасно взаимодействовать друг с другом. Эти производные криптографические ключи периодически меняются, чтобы злоумышленники не имели времени и последовательности сообщений, достаточных для определения этих ключей.

Для сообщений <публикация-подписка> определения, связанные с безопасностью, приведены в ГОСТ Р 71809 и содержат описание того, как защитить сообщения, а также как получить ключи безопасности, необходимые для защиты сообщений.

Издатель должен использовать предоставленные ключи для защиты сообщения. Он зашифрует текст сообщения и подпишет все сообщение целиком. Подписчики должны использовать ключи для расшифровки и проверки подписи сообщений. Чтобы получить необходимые ключи, издатель или подписчик используют связь клиент—сервер. Ключи также должны быть получены с помощью методов сеансового вызова.

4.14 Аудит

4.14.1 Общие положения

Клиенты и серверы генерируют записи аудита успешных и неуспешных попыток подключения, результатов согласования параметров безопасности, изменений конфигурации, системных изменений, взаимодействий с пользователем и отклонений сеансов.

OPC UA обеспечивает поддержку журналов аудита безопасности с помощью механизма, обеспечивающего прослеживаемость между журналами аудита клиента и сервера. Клиент создает запись в журнале аудита для операции, которая включает запрос. Когда клиент отправляет запрос на обслуживание, он создает запись в журнале аудита и включает локальный идентификатор записи в запросе, отправляемом на сервер. Сервер регистрирует полученные запросы и включает идентификатор клиента в свой журнал аудита. Если проблема, связанная с безопасностью, обнаруживается, то на сервере идентифицируется и проверяется соответствующая запись в журнале аудита клиента. OPC UA не требует, чтобы записи аудита записывались на диск, но требует, чтобы они были доступны. ОПК UA предоставляет серверам возможность генерировать уведомления о событиях, которые сообщают о проверяемых событиях. Более подробная информация об алгоритме проведения аудита сервиса в OPC UA содержится в ГОСТ Р 71809.

OPC UA определяет параметры аудита, которые должны быть включены в записи аудита. Это обеспечивает согласованность между журналами аудита и событиями аудита. В ГОСТ Р 71810 определены типы данных для этих параметров. Другие информационные модели могут расширять определения аудита. В [2] определены профили, которые включают возможность генерировать события аудита и использовать эти параметры, включая идентификатор записи аудита клиента.

Поскольку журналы аудита используются для подтверждения безопасной работы системы, сами журналы аудита также должны быть защищены от несанкционированного вмешательства. Механизмы защиты журналов аудита выходят за рамки данной спецификации. Кроме того, информация в отчете аудита может содержать конфиденциальную информацию, поэтому возможность подписаться на события аудита предоставляется только соответствующим пользователям и/или приложениям. В качестве альтернативы поля с конфиденциальной или приватной информацией могут содержать код ошибки, указывающий на то, что пользователям, не имеющим соответствующих прав, отказано в доступе.

4.14.2 Единый клиент и сервер

В простом случае взаимодействия клиента с сервером OPC-клиент «A» выполняет некоторую проверяемую операцию, которая включает в себя вызов службы общего доступа OPC на сервере «D». Он записывает свою собственную запись в журнал аудита и включает идентификатор этой записи в запрос на обслуживание, который он отправляет на сервер.

Сервер получает запрос и создает для него свою собственную запись в журнале аудита. Эта запись идентифицируется собственным идентификатором аудита и содержит свою собственную информацию аудита. Он также включает в себя название клиента, отправившего запрос на обслуживание, и идентификатор записи аудита клиента, получившего запрос. Используя эту информацию, аудитор должен проверить совокупность записей журнала сервера и соотнести их с соответствующими записями клиента.

4.14.3 Сервер-агрегатор

Существует случай, когда клиент получает доступ к сервисам с сервера-агрегатора. Сервер-агрегатор — это сервер, который предоставляет свои услуги, получая доступ к сервисам других OPC-серверов общего доступа, называемых серверами нижнего уровня. Каждый из серверов получает запросы и создает для них свою собственную запись в журнале аудита. Каждая запись идентифицируется своим собственным идентификатором аудита и содержит свою собственную информацию аудита. Запись также включает имя клиента, отправившего запрос на обслуживание, и идентификатор записи

аудита клиента, полученный в запросе. Затем сервер передает идентификатор аудита только что созданной записи следующему серверу в цепочке. Используя эту информацию, аудитор проверяет записи журнала сервера и связывает их с соответствующими записями клиента.

4.14.4 Агрегирование данных с помощью сервера, не проводящего аудит

В случае, когда клиент получает доступ к сервисам с сервера агрегирования, который не поддерживает аудит, каждый из серверов получает запросы и создает для них свою собственную запись в журнале аудита, за исключением сервера, который не поддерживает аудит. Этот сервер передает идентификатор аудита, который он получает от своего клиента, на следующий сервер. Так создается требуемая цепочка аудита. В случае, когда сервер не поддерживает запись аудита, вся система может считаться не поддерживающей аудит.

4.14.5 Сервер агрегирования с распределением услуг

В случае, когда клиент отправляет запрос на обслуживание на сервер агрегирования, а служба агрегирования поддерживает эту услугу, отправляя несколько запросов на обслуживание на свои базовые серверы, серверу агрегирования потребуется подписаться на события аудита с серверов, которые он агрегирует. Таким образом он сможет предоставлять все необходимые данные.

5 Проверка безопасности

5.1 Проверка угроз с помощью механизмов безопасности OPC UA

5.1.1 Обзор

В 5.1.2—5.1.12 приведены угрозы, описанные в 4.3, для функций OPC UA. По сравнению с согласованием решения задач, которые будут приведены в 5.2, это согласование является более конкретным, так как оно связывает функции безопасности OPC UA с конкретными угрозами. Важно то, что профилирование сервера косвенно может повлиять на все атаки.

5.1.2 Отказ в обслуживании

5.1.2.1 Обзор

Для регламентации обеспечения безопасности систем промышленной автоматизации «отказ в обслуживании» требуется выделять три основные угрозы: переполнение сообщениями, исчерпание ресурсов и сбои в работе приложений.

5.1.2.2 Утечка сообщений

OPC UA должна сводить к минимуму потерю доступности, вызванную утечкой сообщений, за счет минимизации объема обработки, выполняемой с сообщением до его аутентификации. Это не позволяет злоумышленнику заставить законное приложение OPC UA тратить время на ответ, тем самым отвлекая ресурсы обработки от законных действий.

Точки доступа и характеристики открытого канала безопасности в соответствии с ГОСТ Р 71809 являются единственными информационными сущностями, которые сервер обрабатывает до аутентификации клиента. Ответ на запрос получения полных данных представляет собой всего лишь набор статической информации, поэтому серверу не требуется выполнять большую обработку. Ответ на запрос характеристик открытого канала безопасности потребляет значительные ресурсы сервера. С помощью обработки подписи и шифрования OPC UA минимизирует эту обработку, но устранить ее невозможно.

Серверная реализация могла бы защитить себя от потока сообщений, передаваемых по открытому каналу безопасности двумя способами:

- сервер может намеренно задерживать обработку запросов открытого канала безопасности. Как только он получит количество неверных запросов больше некоторого минимального порога этот канал должен выдать сигнал тревоги, предупреждающий персонал предприятия о том, что происходит атака, которая может блокировать новые законные вызовы открытого канала безопасности;
- когда запрос открытого канала безопасности пытается превысить указанное сервером максимальное количество одновременных каналов, сервер выдает сообщение об ошибке без выполнения обработки подписи и шифрования. Сертифицированные серверы OPC UA должны указывать максимальное количество одновременных каналов в документации к своему продукту в соответствии с [2].

Аутентификация пользователей и клиентов OPC UA снижает риск использования законного клиента для организации атаки с использованием флуда. См. проверку подлинности в 5.2.3.

В PubSub подписчик фильтрует сообщения, которые он обрабатывает, на основе информации заголовка, что позволяет ему быстро отбрасывать любые сообщения, которые не соответствуют требуемому фильтру. Кроме того, проверяется подпись сообщения, чтобы исключить любое сообщение, которое правильно сформировано, но не из нужной группы безопасности. PubSub также может быть настроен на одноадресную передачу вместо многоадресной, что позволяет сетевой инфраструктуре блокировать атаки многоадресного флуда.

Функциональность аудита OPC UA предоставляет сайту доказательства, которые могут помочь сайту обнаружить, что на него совершаются атаки с использованием флуда, и найти способы предотвращения подобных атак в будущем (см. 4.14). В качестве практики требуется отслеживать события аудита на предмет чрезмерных запросов на подключение.

OPC UA полагается на CSM сайта для предотвращения атак, таких как переполнение сообщений, на уровнях протоколов и в системах, поддерживающих OPC UA.

5.1.2.3 Исчерпание ресурсов

Аутентификация пользователя и клиента по протоколу OPC UA снижает риск использования законного клиента для проведения атаки на истощение ресурсов. Кроме того, аудит сервера позволяет обнаружить клиента, если атака на истощение ресурсов была проведена законным клиентом. Серверы также должны повторно обрабатывать запросы открытого защищенного канала, которые не были выполнены в соответствии с ГОСТ Р 71809; это позволит избежать атак со стороны нелегальных клиентов. Атаки на исчерпание ресурсов не применяются к системам PubSub, поскольку никакие сеансы или ресурсы не доступны.

5.1.2.4 Сбои в работе приложений

OPC UA обеспечивает сертификацию приложений OPC UA. Лабораторное тестирование и сертификация включают в себя тестирование путем ввода команд «error» и «junk», которые могут выявить распространенные ошибки. Основные стеки OPC также проходят фаз-тестирование, чтобы убедиться в их устойчивости к ошибкам. Сертифицированное приложение OPC UA не гарантирует безотказной работы, но с большей вероятностью устойчиво к сбоям в работе приложений, вызванным атаками типа «отказ в обслуживании».

5.1.3 Подслушивание

Описание этой угрозы приведено в 4.3.3. OPC UA обеспечивает шифрование для защиты от подслушивания, как описано в 5.2.5.

5.1.4 Подделка сообщений

Описание этой угрозы приведено в 4.3.4. Согласно ГОСТ Р 71809 и ГОСТ Р 71811 OPC UA противодействует угрозам подмены сообщений, предоставляя возможность подписывать сообщения. Сообщения всегда будут содержать действительные идентификаторы сеанса, защищенного канала, запроса и временную метку, а также правильный порядковый номер. OPC UA, работая в рамках сеанса, таким же образом ограничивает подмену пользователей, поскольку информация о пользователе предоставляется в рамках установления сеанса. Требуется, чтобы при запуске устройства идентификатор сеанса, который изначально присваивается первому сеансу, был случайным числом или продолжением номера последнего использованного сеанса и не всегда сбрасывался на 0 или предсказуемое число.

Согласно [3] OPC UA PubSub противодействует угрозам подделки сообщений, предоставляя возможность подписывать сообщения. Сообщения также могут содержать действительные идентификаторы издателя и класса набора данных, информацию о временной метке, номер сетевого сообщения и порядковый номер, что еще больше ограничивает подделку сообщений.

5.1.5 Изменение сообщения

В 4.3.5 представлено описание этой угрозы. Если сообщения изменены, проверка подписи выявит любые изменения и позволит получателю отказаться от данного сообщения. Эта проверка является средством предотвращения непреднамеренного изменения сообщения из-за ошибок транспорта связи.

5.1.6 Воспроизведение сообщения

В 4.3.6 содержится описание этой угрозы. OPC UA использует идентификаторы сеансов и защищенных каналов, временные метки, порядковые номера и идентификаторы запросов.

Сообщения подписаны и не могут быть изменены без обнаружения, поэтому очень сложно воспроизвести сообщение, например, чтобы оно имело действительный идентификатор сеанса, идентификатор защищенного канала, метку времени, порядковые номера и идентификатор запроса в соответствии с ГОСТ Р 71809 и ГОСТ Р 71811. Установление безопасного канала/сеанса включает ту же подпись, временные метки и порядковый номер, которые являются частью всех сообщений и, следовательно, не могут быть воспроизведены.

OPC UA <публикация-подписка> должна использовать публикацию, идентификатор набора данных, а также временные метки номера сетевых сообщений и порядковые номера для опубликованных сообщений. Сообщения должны быть подписаны и не могут быть изменены без обнаружения. <Публикация-подписка> позволяет отключать поля в сообщении.

5.1.7 Неверные сообщения

В 4.3.7 дано описание этой угрозы. Реализации приложений OPC UA должны противодействовать угрозам искаженных сообщений, проверяя, что сообщения имеют правильную форму и что параметры сообщений находятся в пределах допустимого диапазона. Недействительные сообщения отбрасываются. Это указано в ГОСТ Р 71809, ГОСТ Р 71811 и [3].

5.1.8 Профилирование сервера

В 4.3.6 дано описание этой угрозы. OPC UA должна ограничивать объем информации, которую серверы предоставляют клиентам, которые еще не идентифицированы. Эта информация является ответом на службу получения полных данных, указанную в ГОСТ Р 71809.

5.1.9 Перехват сеанса

В 4.3.9 дано описание этой угрозы. OPC UA должна противодействовать перехвату сеанса путем назначения контекста безопасности (т. е. защищенного канала) для каждого сеанса, как указано в службе создания сеанса в ГОСТ Р 71809.

Таким образом, для захвата сеанса сначала потребуется скомпрометировать контекст безопасности.

5.1.10 Мошеннический сервер или издатель

В 4.3.10 и 4.3.11 даны описания этой угрозы. Клиентские приложения OPC UA должны противодействовать использованию мошеннических серверов, проверяя сертификаты экземпляров серверных приложений. Существует вероятность того, что мошеннический сервер предоставит сертификат от сертифицированного OPC UA-сервера, но, поскольку он не обладает соответствующим закрытым ключом для расшифровки сообщений, защищенных правильным открытым ключом, мошеннический сервер никогда не сможет читать и злоупотреблять защищенными данными, отправленными клиентом. Без закрытого ключа сервер никогда не сможет подписать ответное сообщение клиенту.

Приложения подписчика OPC UA противодействуют эффекту мошеннического издателя, проверяя подпись опубликованных сообщений.

5.1.11 Компрометация учетных данных пользователя

В 4.3.11 дано описание этой угрозы. OPC UA должна защищать учетные данные пользователя, передаваемые по сети, с помощью шифрования, как описано в 5.2.5. OPC UA зависит от CSMS сайта для защиты от других атак с целью получения учетных данных пользователя, таких как подбор пароля или социальная инженерия.

5.1.12 Отказ от участия

В 4.3.13 дано описания этой угрозы. Клиентские и серверные приложения OPC UA должны противодействовать отказу путем подписания сообщений, указанных в ГОСТ Р 71809. Подписанное сообщение указывает, что сообщение пришло от владельца закрытого ключа. Во время работы открытого канала безопасности и установления сеанса взаимодействующие стороны четко идентифицируются и подтверждаются. Аудит в соответствии с ГОСТ Р 71809 отслеживает информацию, связанную с сообщением.

5.2 Согласование задач с механизмами безопасности OPC UA

5.2.1 Обзор

Следующие подразделы согласовывают задачи, описанные в 4.2, с функциями OPC UA. По сравнению с согласованием угроз 5.1, это согласование оправдывает полноту архитектуры безопасности OPC UA.

5.2.2 Аутентификация приложения

Приложения OPC UA поддерживают аутентификацию объектов, с которыми они взаимодействуют.

Как указано в службах получения полных данных и открытого канала безопасности в соответствии с ГОСТ Р 71809, клиентские и серверные приложения OPC UA должны идентифицировать и аутентифицировать себя с помощью сертификатов X.509 v3 и связанных с ними закрытых ключей (см. X509). Некоторые варианты стека связи требуют, чтобы эти сертификаты представляли собственно конечный автомат или пользователя, а не приложение.

Для связи публикации и подписки требуется связь клиента с сервером для получения от сервиса общих ключей безопасности (SKS). Хотя аутентификация приложения не осуществляется напрямую между подписчиком и издателем, SKS гарантирует, что только прошедшие проверку подлинности приложения смогут получить ключи, используемые издателем и подписчиком.

5.2.3 Аутентификация пользователя

Приложения OPC UA поддерживают аутентификацию пользователей, обеспечивая необходимую аутентификацию полномочия другим субъектам. В службе активизации сеанса согласно ГОСТ Р 71809 клиент OPC UA должен принимать от пользователя несертифицированный токен и передавать его на сервер OPC UA. Сервер OPC UA должен аутентифицировать токен пользователя. Приложения OPC UA принимают токены в любой из следующих форм: имя пользователя/пароль, сертификат X.509 v3 (см. X509) или веб-токен JSON (JWT).

Согласно ГОСТ Р 71809 если несертифицированный токен является сертификатом, то этот токен проверяется с помощью процесса запроса-ответа. Сервер предоставляет данное время и алгоритм подписи в качестве запроса в своем ответе создания сеанса. Клиент отвечает на запрос, подписывая данное время сервера и предоставляя его в качестве аргумента при последующем вызове.

5.2.4 Авторизация

OPC UA не определяет, как должна предоставляться авторизация пользователя или клиента. Приложения OPC UA, являющиеся частью более крупного продукта промышленной автоматизации, должны иметь возможность управлять авторизациями, в соответствии с управлением авторизацией этого продукта. Идентификация и аутентификация количества пользователей указывается в OPC UA, чтобы клиентские и серверные приложения могли распознавать пользователя и определять уровень авторизации пользователя.

Серверы OPC UA отвечают кодом ошибки <плохому пользователю отказано в доступе>, указывая на ошибку авторизации или аутентификации, как указано в кодах состояния, определенных в ГОСТ Р 71809.

При взаимодействии публикации-подписки авторизация пользователя может использоваться как часть распределения ключей (SKS). Это позволяет издателю и SKS ограничивать доступ определенным пользователям.

5.2.5 Конфиденциальность

OPC UA использует симметричное и асимметричное шифрование для защиты конфиденциальности в качестве средства безопасности в выполнении задачи. Таким образом, асимметричное шифрование используется для согласования ключей и симметричного шифрования для защиты всех других сообщений, передаваемых между приложениями OPC UA. Механизмы шифрования определены в ГОСТ Р 71811 и [3].

OPC UA полагается на CSMS сайта для защиты конфиденциальности в сетевой и системной инфраструктуре. OPC UA полагается на PKI для управления ключами, используемыми для асимметричного шифрования, который затем используется для установления симметричных сеансовых ключей.

5.2.6 Целостность

OPC UA использует симметричные и асимметричные подписи для обеспечения целостности как цели безопасности. Асимметричные подписи используются на этапе согласования ключей во время установления безопасного канала.

Симметричные подписи применяются ко всем остальным сообщениям, включая сообщения публикации-подписки.

OPC UA полагается на CSMS сайта для защиты целостности сетевой и системной инфраструктуры. OPC UA полагается на PKI для управления ключами, используемыми для асимметричных подписей, которые затем используются для установления симметричных сеансовых ключей.

5.2.7 Проверяемость

Согласно ГОСТ Р 71809 OPC UA поддерживает ведение журнала аудита, обеспечивая отслеживание действий через записи журнала нескольких клиентов и серверов, которые инициируют, пересылают и обрабатывают действия. OPC UA зависит от продуктов приложений OPC UA, обеспечивающих эффективную схему регистрации аудита или эффективный способ сбора аудита.

Схема «события всех узлов» может быть частью более крупного продукта промышленной автоматизации, частью которого являются приложения OPC UA.

5.2.8 Доступность

OPC UA должна сводить к минимуму влияние лавинной рассылки сообщений, как описано в 5.1.2. Некоторые атаки на доступность включают открытие большего количества сеансов, чем может обрабо-

тать сервер, что приводит к сбою или плохой работе сервера. Серверы отклоняют сеансы, число которых превышает указанное максимальное количество. Другие аспекты OPC UA, такие как безопасный диалог OPC UA, также могут влиять на доступность и содержатся в ГОСТ Р 71811.

6 Вопросы реализации и развертывания

6.1 Обзор

В разделе 6 представлены положения, определяющие действия поставщиков, реализующих приложения OPC UA. Поскольку многие из контрмер, необходимых для устранения описанных выше угроз, выходят за рамки серии стандартов OPC UA, положения в разделе 6 показывают, как следует обеспечивать некоторые из этих контрмер.

Для каждой из следующих областей раздел 6 определяет область проблем, определяет последствия, если соответствующие контрмеры не будут реализованы, и рекомендует лучшие практики.

6.2 Соответствующие тайм-ауты

Тайм-ауты, время ожидания реализации (такого события, как получение сообщения) играют важную роль, влияя на безопасность реализации. Потенциальные последствия включают в себя:

- отказ в обслуживании. Условия отказа в обслуживании могут существовать, когда клиент не сбрасывает сеанс, если тайм-ауты очень велики;
- потребление ресурсов: когда клиент простаивает в течение длительного периода времени, сервер сохраняет буферизованное сообщение или информацию клиента в течение этого периода, что приводит к истощению ресурсов. Разработчик должен использовать разумные таймауты для каждого этапа подключения.

6.3 Строгая обработка сообщений

Спецификации определяют формат правильных сообщений. При наличии сообщений, которые отклоняются от спецификации, разработчик должен выполнить строгую проверку формата сообщения и отбросить пакеты либо отправить сообщение об ошибке, как описано ниже:

- при обработке ошибок используется код ошибки согласно ГОСТ Р 71809, который наиболее точно соответствует условию и подходит только при возврате кода ошибки. Коды ошибок используются в качестве вектора атаки, поэтому их использование должно быть ограничено, в соответствии с ГОСТ Р 71809. Данный стандарт определяет, что одна общая ошибка возвращается до и во время установления защищенного канала. После установления защищенного канала возвращаются соответствующие конкретные коды ошибок;
- вектором атаки являются изменения времени. Результаты такой атаки сводятся к минимуму согласно описанию в ГОСТ Р 71809, которое требует закрытия сокета в случае любых ошибок при установлении защищенного канала. Поставщикам следует проявлять осторожность при реализации, чтобы гарантировать, что все пути, которые приводят к закрытию сокета, не предоставляют подсказку по времени, указывающую, какой путь сбоя был обнаружен. Этого можно добиться, установив случайную задержку перед закрытием сокета или перед возвратом общего кода ошибки.

Все длины массивов, длины строк и глубина рекурсии должны строго соблюдаться.

6.4 Генерация случайных чисел

Случайные числа, отвечающие требованиям безопасности, генерируются с помощью подходящих функций, предоставляемых криптографическими библиотеками. Обычные случайные функции, такие как использование (rand), предоставляемые стандартной библиотекой «С», не генерируют достаточную энтропию. В качестве альтернативы разработчики должны использовать генераторы случайных чисел, предоставляемые библиотекой Microsoft Windows Crypto (библиотека WinCrypt) или OpenSSL. Даже случайные функции, представленные в криптографических библиотеках, требуют для инициализации источника энтропии, а необходимая энтропия не всегда доступна на встроенных устройствах. ПК могут использовать несколько отдельных фрагментов информации (идентификаторы оборудования, такие как процессор, MAC, адреса, USB-устройства, разрешение экрана, установленное программное обеспечение и т. д.) для генерации энтропии, но встроенные устройства устроены совершенно идентично.

Часто для энтропии остается только время и, возможно, MAC-адрес. Это делает встроенные устройства очень уязвимыми.

В целом, альтернативными решениями, которыми поставщик должен пользоваться, являются:

- при проектировании встраиваемых устройств добавление специального оборудования для генератора энтропии;
- отказ от создания сертификатов на встроенных устройствах. Рекомендация использовать внешний инструмент или GDS, чтобы сгенерировать сертификат и загрузить его на устройство;
- ожидание получения достаточного объема информации об энтропии;
- для встроенных систем без хорошего источника энтропии полезно сохранить состояние криптографического генератора псевдослучайных чисел (CPRNG), чтобы он не выдавал одни и те же случайные числа после каждой загрузки.

Поставщик должен гарантировать, что используемые им криптографические функции инициализируются с подходящей энтропией и что сгенерированные сертификаты не создаются предсказуемым образом.

6.5 Специальные и зарезервированные пакеты

Реализация считывает и правильно интерпретирует любые типы сообщений, которые зарезервированы как специальные (например, широковещательные и многочисленные адреса в спецификации IP). Неспособность понять и интерпретировать эти специальные пакеты может привести к уязвимостям.

6.6 Ограничение скорости и управление потоком

Требования к OPC UA не предоставляют механизмов управления скоростью; однако ее реализация может включать в себя управление скоростью.

6.7 Административный доступ

Согласно требованиям к OPC UA такие функции, как управление хранилищами сертификатов, должны быть доступны только администраторам. Система стандартов OPC UA (все части) не описывает детали, связанные с административным доступом. Характер административного доступа варьируется от платформы к платформе. На некоторых платформах есть только один администратор.

Другие платформы предоставляют несколько уровней административного доступа, например, администратор резервного копирования, сетевой администратор, администратор конфигурации и т. д. На сайте развертывания следует сделать соответствующие выборы для доступа администратора, а разработчик должен разрешить настройку соответствующего доступа к учетной записи администратора.

Ограничения административного доступа включают такие элементы, как файлы конфигурации для серверов и клиентов. Например, файлы конфигурации могут содержать пути к хранилищам сертификатов или открытым конечным точкам, изменение которых может привести к серьезным проблемам.

Административный доступ также следует использовать для управления событиями аудита, дополнительную информацию см. в 4.14.

6.8 Криптографические ключи

Профили безопасности, определенные в [2] и ГОСТ Р 71809, описывают необходимые алгоритмы и требуемую длину ключей. Требования к длине ключа могут быть указаны в диапазоне, например 1024—2048. Важно, чтобы приложение OPC UA поддерживало весь диапазон своего сертификата экземпляра приложения. Это позволяет конечному пользователю создать ключ (сертификат экземпляра приложения), отвечающий его требованиям безопасности. Это может продлить период времени, в течение которого можно использовать данный профиль безопасности. Например, ключи длиной менее 2048 уже считаются небезопасными, но, если пользователь генерирует сертификаты для верхнего предела диапазона (2048), то приложение по-прежнему может считаться безопасным (в зависимости от других алгоритмов).

6.9 Рекомендации, связанные с сигнализацией

OPC UA должна поддерживать надежную информационную модель сигналов тревоги и состояний, которая включает в себя возможность отключать сигналы тревоги, откладывать сигналы тревоги и в целом управлять сигналами тревоги. Обработка и управление сигналами тревоги являются важной частью поддержания эффективного управления предприятием. С точки зрения безопасности важно,

чтобы этот путь был адекватно защищен, чтобы гарантировать, что мошеннический агент не создаст опасную или финансовую ситуацию. OPC UA предоставляет инструменты, необходимые для этой защиты, но разработчику необходимо убедиться, что они используются правильно. Все функции, которые позволяют вносить изменения в рабочую среду, могут генерировать события аудита и должны быть доступны только соответствующим пользователям.

Отключение сигналов тревоги является одной из таких функций, доступ к которой должен иметь только персонал с соответствующими правами доступа. Более того, любое действие, которое отключает сигнал тревоги, независимо от того, инициировано ли оно персоналом или какой-либо автоматизированной системой, должно генерировать событие аудита, созданное этим действием.

Размещение сигналов тревоги должно следовать тем же правилам, что и отключение сигналов тревоги в отношении доступа и аудита, хотя оно может быть доступно более широкому кругу пользователей (операторов, инженеров). Кроме того, разработчик должен убедиться, что для хранения сигналов тревоги настроены соответствующие тайм-ауты. Эти тайм-ауты должны гарантировать, что сигнал тревоги не может быть отложен на период времени, который может вызвать проблемы с безопасностью.

События диалога могут использоваться для перегрузки клиента. Лучшей практикой для серверов, которые поддерживают диалоги, является ограничение количества одновременно активных диалогов. Диалоги должны включать период ожидания, чтобы гарантировать, что они не будут использоваться для создания DOS.

Разработчики клиента должны гарантировать, что любая обработка диалогов не может быть использована для перегрузки оператора. Максимальное количество открытых диалогов должно быть ограничено, а диалоги должны иметь возможность игнорироваться (т. е. должна быть доступна другая обработка).

6.10 Доступ к программе

Функциональные возможности OPC UA должны позволять запуск программы в виде части работы сервера OPC UA. Эти программы можно использовать для выполнения расширенных алгоритмов управления или других действий. Использование этих действий должно быть разрешено только персоналу с соответствующими правами доступа.

Кроме того, следует тщательно контролировать определение программ. Рекомендуется вести статистику количества рабочих программ, а также частоты их использования. Эта информация доступна административному персоналу. Нельзя допускать неограниченное количество запусков программы.

6.11 Управление событиями аудита

Серия стандартов OPC UA определяет события аудита в виде требований, которые должны быть сгенерированы, и информации, которую эти события аудита включают как минимум; однако на события аудита могут подписываться несколько систем отслеживания аудита или систем регистрации.

Серия стандартов OPC UA не описывает эти системы. Предполагается, что любое количество систем, предоставленных поставщиком, может обеспечить эту функциональность. Независимо от того, какая система используется для хранения и управления, события аудита должны обеспечивать следующее:

- характеристики событий аудита не изменяются после их получения;
- подписка на события аудита должна осуществляться через безопасный канал, чтобы гарантировать, что они не будут подделаны во время перехода;
- клиентам, которые регистрируют события аудита, рекомендуется сохранять зарегистрированные события аудита, чтобы эти события можно было аутентифицировать и связать с исходной транзакцией.

В зависимости от CSMS сайта система управления событиями аудита может иметь дополнительные требования.

6.12 OAuth2, JWT и роли пользователя

OAuth2 определяет стандарт для сервиса авторизации, которые создают веб-токены JSON (JWT), также известные как токены доступа. Эти JWT передаются как выданный токен на сервер OPC UA, который использует подпись, содержащуюся в JWT, для проверки токена. JWT также может предоставлять серверу информацию о ролях, связанных с пользователем, прошедшим проверку подлинности.

Ответственность за соблюдение ролей лежит на сервере. В ГОСТ Р 71809, ГОСТ Р 71810 и ГОСТ Р 71811 подробно описаны OAuth2 и JWT. Сайты должны гарантировать, что они следуют рекомендациям, определенным в CSMS сайта для OAuth2.

6.13 HTTP, SSL/TLS и веб-сокеты

HTTPS определяет положения, характеризующие стандартную транспортную безопасность. Такая транспортная безопасность не всегда обеспечивает сквозную безопасность. Могут существовать прокси-серверы или другие посредники. Если требуется сквозная безопасность, требуются дополнительные шаги, такие как VPN.

Если поддерживается связь SSL/TLS, ключи, используемые для TLS, должны отличаться от ключей для связи TCP. Повторное использование ключей приводит к проблемам безопасности. Должен быть включен только TLS 1.2; другие версии TLS имеют проблемы с безопасностью, и их не следует включать.

SLL второй версии имеет проблемы с безопасностью, и ее следует отключить. Важно, чтобы она была отключена для всех приложений на компьютерном комплексе, а не только для приложения UA.

Вебсокеты — это протокол, защищенный с помощью HTTPS. При использовании веб-сокетов необходимо соблюдать все рекомендации по безопасности для HTTP и TLS.

6.14 Обратное подключение

Обратное подключение позволяет серверу инициировать соединение с клиентом (открыть сокет, отправив сообщение HEL). Это приводит к дополнительной проблеме безопасности для клиента, поскольку клиенту необходимо проверить, что соединение происходит с соответствующего сервера, а не является атакой типа «отказ в обслуживании». Если сервер не отвечает своевременно на запрос открытия защищенного канала, клиенту следует закрыть канал.

7 Незащищенные услуги

7.1 Обзор

OPC UA включает в себя ряд служб, доступ к которым не требует обеспечения безопасности. Эти службы требуют особого внимания с точки зрения безопасности. Эти службы предоставляют возможности, позволяющие клиентам обнаруживать серверы и подключаться к ним. Службы обнаружения доступны как локальные или глобальные и могут быть многоадресными.

7.2 Обнаружение многоадресной рассылки

Для многоадресного обнаружения OPC UA требуется настраивать способом, при котором серверы объявляют о себе в подсети при запуске. Компьютеры приложений или само приложение должны прослушивать и создавать список доступных серверов.

Многоадресные DNS-операции небезопасны по самой своей природе; они позволяют мошенническим серверам транслировать свое присутствие или выдавать себя за другой хост или сервер. Риски от мошеннических серверов сводятся к минимуму, если включена безопасность OPC UA и все приложения используют списки доверия сертификатов для контроля доступа. Кроме того, клиенты должны кэшировать информацию о соединении, сводя к минимуму поиск информации о сервере. Однако даже если обеспечена безопасность UA, в средах, где злоумышленник может легко получить доступ к сети, многоадресную DNS требуется отключать.

Приложения (или серверы обнаружения) созданы так, чтобы гарантировать, что они не могут быть перегружены или отключены из-за высоких скоростей широковещательной передачи по каналу многоадресного обнаружения или из-за слишком большого списка серверных приложений.

7.3 Безопасность сервера «Глобальное открытие»

7.3.1 Обзор

Сервер регистрации стратегических открытий (GDS) — это специальный сервер OPC UA, который предоставляет услуги обнаружения инноваций для предприятия или для системы более высокого уровня. Он может обеспечивать функциональность управления сертификатами (см. [4]).

Существует несколько способов доступа к GDS:

1) серверы могут регистрироваться на сервере регистрации инноваций;

2) клиенты могут запрашивать в GDS доступные серверы;

3) клиенты могут получать сертификаты из GDS;

4) серверы могут получать сертификаты из GDS;

5) GDS может отправлять сертификаты на сервер;

6) GDS может получить доступ к другим серверам обнаружения для создания списка доступных серверов.

В зависимости от доступных методов доступа необходимо обозначить несколько типов угроз:

1) угрозы, связанные с наличием в системе мошеннической GDS;

2) угрозы GDS, включая наличие мошеннических клиентов или серверов;

3) угрозы функции управления сертификатами, предоставляемой GDS.

7.3.2 Несанкционированная GDS

При работе с GDS требуется:

- чтобы серверы регистрировались на сервере инноваций, на котором они настроены для регистрации, и чтобы серверы не регистрировались вслепую на GDS, для которой они не настроены. Серверы должны знать, что сервер регистрации инноваций; может быть мошенническим;

- сервер должен регистрировать все конечные точки, которые он предоставляет, гарантируя совпадение списков, предоставленных сервером обнаружения. Это гарантирует, что клиенты смогут определить, предоставил ли сервер регистрации инноваций; достоверную информацию;

- клиенты должны знать о мошеннических серверах обнаружения, которые могут перенаправить их на мошеннические серверы. Клиенты могут использовать сертификат сервера SSL/TLS (если доступен) для проверки того, что сервер обнаружения является сервером, которому они доверяют, и/или убедиться, что они доверяют любому серверу, предоставленному сервером обнаружения;

- согласно ГОСТ Р 71809 клиенты всегда проверяют, доверяют ли они сертификату сервера и что конечный узел URI соответствует именам хостов, указанным в сертификате, прежде чем создавать сеанс с сервером. Описание конечной точки, предоставляемой сервером, включает относительный уровень защиты для определения, использовалась ли наиболее безопасная конечная точка.

7.3.3 Угрозы в отношении GDS

Согласно ГОСТ Р 71809 служба поиска серверов в сети используется без обеспечения безопасности, и поэтому она уязвима для атак типа «отказ в обслуживании» (DOS). Для минимизации объема обработки, необходимой для отправки ответа для этого сервиса, сервер обнаружения должен готовить результат заранее.

GDS принимает регистрацию серверов только от серверов, которым доверяют или которые имеют соответствующие права административного доступа. Это поможет гарантировать, что мошеннический сервер не будет зарегистрирован в GDS.

7.3.4 Угрозы управления сертификатами

GDS, которая также обеспечивает управление сертификатами, поддерживает безопасность пользовательского доступа, как описано в системе стандартов OPC UA.

Это включает в себя предоставление администраторам всех функций управления сертификатами. Список клиентов, которым разрешен доступ к функциям управления, может быть ограничен.

Управление сертификатами включает этап подготовки и этап выполнения. Фаза подготовки — это когда GDS предоставляет первоначальные сертификаты клиентам или серверам, которые только входят в систему. Фаза выполнения представляет собой повседневную работу системы и включает предоставление обновленных списков отзыва сертификатов, продление сертификатов и обновленных списков доверия.

Предоставление систем по своей сути небезопасно, но может быть очень полезно для значительно упрощенного развертывания сложной системы. Инициализация в GDS не включена по умолчанию, но для ее включения требуется административное действие. Также рекомендуется, чтобы функция подготовки, если она включена, оставалась включенной только в течение ограниченного времени.

Фаза выполнения операций с сертификатами GDS может выполняться безопасно, поскольку все серверы и клиенты уже имеют сертификаты для обеспечения безопасного соединения. Для принудительной модели управления сертификатами GDS устанавливает безопасный канал, используя самый высокий уровень безопасности, доступный на центральном сервере. Он не предоставляет обновленные списки отзыва сертификатов, сертификаты или списки доверия через конечную точку, уровень безопасности которой ниже уровня безопасности обновлений.

Например, если необходимо обновить сертификат 4096, его нельзя обновить с помощью канала 2048, но сертификат 2048 можно обновить с помощью канала 4096. Если необходимо развернуть новый сертификат более высокого уровня, это обрабатывается так же, как и предоставление нового сервера.

8 Управление сертификатами

8.1 Обзор

Приложения OPC UA имеют сертификаты экземпляра приложения для обеспечения безопасности на уровне приложения. Они применяются для установления безопасного соединения с использованием асимметричной криптографии. Эти сертификаты экземпляра приложения представляют собой сертификаты X.509 v3, содержат список элементов данных, которые определены в ГОСТ Р 71809 и ГОСТ Р 718011. Эти элементы данных описывают экземпляр приложения, которому назначен сертификат.

Сертификаты включают цифровую подпись генератора сертификата. Эта цифровая подпись может быть самоподписанной (подпись создается с помощью закрытого ключа, связанного с сертификатом X.509 v3, который является сертификатом экземпляра приложения) или может быть подписана центром сертификации (подпись создается с помощью закрытого ключа, связанного с X.509 v3 Сертификат ЦС). Оба типа сертификатов обеспечивают одинаковый уровень безопасности и могут использоваться в асимметричной криптографии. Подписи могут быть сгенерированы с использованием различных алгоритмов, причем алгоритмы обеспечивают разные уровни безопасности (128 бит, 256 бит, 512 бит и т. д.). Алгоритм, необходимый для подписания сертификата, указан в политике безопасности. Серверы и клиенты должны иметь возможность поддерживать более одного сертификата, поскольку может потребоваться более одного сертификата в зависимости от поддерживаемых профилей безопасности.

Асимметричная криптография использует два ключа — закрытый ключ и открытый ключ. Приложение OPC UA будет иметь список доверенных открытых ключей, которые представляют приложения, которым оно доверяет. Этот список доверенных открытых ключей хранится в реестре Windows либо в папке с файлами. У него также будет закрытый ключ, соответствующий его сертификату экземпляра приложения. Приложение OPC UA может использовать открытый ключ из своего списка для проверки того, что подпись полученного запроса на соединение была сгенерирована соответствующим закрытым ключом. Приложение также может использовать открытый ключ целевого приложения для шифрования данных, которые можно расшифровать только с помощью закрытого ключа целевого приложения.

8.2 Управление самоподписанными сертификатами

Основное различие между подписанным CA и самозаверяющим сертификатом в установке OPC UA заключается в усилиях, необходимых для развертывания и обслуживания сертификатов. Выбор того, когда использовать сертификат, выданный центром сертификации, вместо самозаверяющего сертификата, зависит от требований к установке и месту установки. Администратору потребуется скопировать открытый ключ, связанный со всеми клиентскими приложениями, во все серверные приложения, с которыми им может потребоваться связь. Кроме того, администратору потребуется скопировать открытый ключ, связанный со всеми серверными приложениями, во все клиентские приложения. Приложения, которым может потребоваться взаимодействовать с ними, количество серверов и клиентов растет, усилия администрации могут стать слишком обременительными. Кроме того, сертификат имеет срок действия, и в какой-то момент его необходимо будет заменить обновленным сертификатом. Для этого потребуется генерировать новые закрытые и открытые ключи и снова копировать все открытые ключи. В очень небольших установках содержится явное указание клиентов, которым доверяет сервер, путем установки открытого ключа сертификата экземпляра клиентского приложения в доверенный сертификат. Хранилище сервера может быть приемлемым.

8.3 Управление сертификатами, подписанными ЦС

В системах с несколькими серверами и клиентами установка открытых ключей в списках доверия может очень быстро стать затруднительной. В таких случаях использование центра сертификации конкретной компании может значительно упростить вопросы установки/настройки. Центр сертификации также может предоставлять дополнительные преимущества, такие как управление сроком действия сертификатов и списками отзыва сертификатов (CRL).

Администратору необходимо в приложении создать экземпляр сертификата, подписанного центром сертификации, для всех клиентов и серверов, установленных в системе, но ему нужно будет только установить открытый ключ центра сертификации на всех компьютерах. Когда срок действия сертификата истекает, и он заменяется, администратору нужно будет только заменить сертификат с истекшим сроком действия (открытые ключи и закрытые ключи), нет необходимости копировать открытый ключ в какие-либо места.

Специальный центр сертификации компании позволяет компании контролировать выдачу сертификатов. Использование коммерческого центра сертификации (например, VeriSign) в большинстве случаев не рекомендуется. Приложение OPC UA должно быть настроено так, чтобы доверять только другим приложениям, определенным компанией как доверенные. Если бы всем сертификатам, выданным коммерческим центром сертификации, можно было доверять, то не компания, а коммерческий центр сертификации смог бы контролировать, каким приложениям следует доверять.

Управление сертификатами должно быть реализовано всеми разработчиками приложений. Некоторые приложения могут использовать управление сертификатами, которое предоставляется как часть общесистемной инфраструктуры, другие генерируют самозаверяющие сертификаты в рамках установки. См. [4] для получения дополнительной информации об общесистемных инфраструктурах для управления сертификатами.

8.4 Управление сертификатами GDS

8.4.1 Обзор

В некоторых системах допускается развертывание сервера регистрации стратегических инноваций с управлением сертификатами. Сервер регистрации стратегических инноваций отправит сертификаты клиентам и серверам либо позволит серверам и клиентам получать сертификаты. Сервер регистрации стратегических инноваций может управлять всеми развертываниями сертификатов; сюда входят списки доверия, центры сертификации и списки отзыва сертификатов.

8.4.2 Управление сертификатами разработчиков

Если приложение OPC UA поддерживает сертификаты, при установке рекомендуется автоматически предоставлять самозаверяющий сертификат экземпляра приложения. Допустимо, чтобы приложение OPC UA могло заменить самозаверяющий сертификат экземпляра приложения сертификатом экземпляра приложения, выданным центром сертификации, или получить самозаверяющий сертификат, подписанный центром сертификации. Настройка списка доверия также должна быть выполнена. Списки доверия для открытых ключей экземпляров приложений хранятся в списке, отдельном от списков ЦС. Приложение OPC UA должно иметь возможность обрабатывать списки отзыва сертификатов (CRL). Это списки открытых ключей, связанных с данным центром сертификации, которые были отозваны. Это позволяет центру сертификации удалять из обращения подписанный им сертификат. CRL предоставляются центром сертификации и обычно распространяются автоматически; дополнительную информацию см. в [4].

Для обеспечения безопасности важно, чтобы хранилища сертификатов, используемые для хранения закрытых ключей, были защищены и защищены только с возможностью доступа для чтения/записи соответствующему администратору и/или приложению OPC UA. Списки доверия, списки отзыва сертификатов и списки доверенных центров сертификации защищены, позволяя только доступ на запись соответствующему администратору, а в случае конфигурации по запросу-приложению. Доступ на чтение должен быть предоставлен другим действительным пользователям, но список пользователей, которым разрешен доступ на чтение, будет определяться решением сайта.

С точки зрения установки рекомендуется предоставить стандартный инструмент для создания сертификата экземпляра приложения. Этот инструмент может быть предоставлен поставщиком OPC UA SDK или базовой OPC. Стандартный инструмент гарантирует, что экземпляр приложения «Создаваемые сертификаты» включает все обязательные поля и настройки. Конкретное приложение OPC UA должно иметь возможность принимать и устанавливать любые действительные сертификаты экземпляра приложения, созданные внешними инструментами. Выбор конкретного инструмента зависит от конкретного места.

Библиография

[1] Глоссарий терминов информационной безопасности [www.infosystems.ru]

[2] МЭК 62541-7:2020 Унифицированная архитектура OPC. Часть 7. Профили (OPC unified architecture — Part 7: Profiles)

[3] МЭК 62541-14:2020 Унифицированная архитектура OPC. Часть 14. PubSub (OPC unified architecture — Part 14: PubSub)

[4] МЭК 62541-12:2020 Унифицированная архитектура OPC. Часть 12. Обнаружение и глобальные службы (OPC unified architecture — Part 12: Discovery and global services)

УДК 004.03:006.354

ОКС 35.240.99
03.100.30

Ключевые слова: цифровая промышленность, унифицированная архитектура, открытая распределенная система, модели безопасности, характеристика угроз безопасности, термины безопасности, функции безопасности, обеспечение безопасности, защита данных, протокол связи OPC UA

Редактор *Н.А. Аргунова*
Технический редактор *В.Н. Прусакова*
Корректор *С.И. Фирсова*
Компьютерная верстка *М.В. Малеевой*

Сдано в набор 24.12.2024. Подписано в печать 10.01.2025. Формат 60×84⅛. Гарнитура Ариал.
Усл. печ. л. 4,18. Уч.-изд. л. 3,35.

Подготовлено на основе электронной версии, предоставленной разработчиком стандарта

Создано в единичном исполнении в ФГБУ «Институт стандартизации»
для комплектования Федерального информационного фонда стандартов,
117418 Москва, Нахимовский пр-т, д. 31, к. 2.
www.gostinfo.ru info@gostinfo.ru