УДК 004.056.53 Вестник УрФО № 1(59) / 2026, с. 48–57

Ощепков Н.В., Южаков А.А., Кротова Е.Л. DOI: 10.14529/secur260105

ВРЕМЕННЫЕ ИНТЕРВАЛЫ СВЯЗИ КАК ФАКТОР ДОВЕРИЯ В ПРОТОКОЛЕ MODBUS RTU

В работе предлагается метод повышения безопасности протокола Modbus RTU путём введения механизма временных контрактов и HMAC-аутентификации. Отсутствие встроенных средств защиты в протоколе Modbus делает системы АСУ ТП уязвимыми к атакам типа «человек посередине» и подмены узлов. Предложенный метод основан на предварительном согласовании временных интервалов обмена данными между ведущим и ведомыми устройствами с криптографической верификацией на основе HMAC. Метод обеспечивает совместимость с существующей инфраструктурой и не требует значительных вычислительных ресурсов, что делает его применимым для промышленных контроллеров с ограниченными возможностями. Экспериментальная проверка подтвердила эффективность метода в предотвращении несанкционированного доступа при минимальных накладных расходах.

Ключевые слова: *Modbus RTU, информационная безопасность АСУ ТП, HMAC, временные контракты, аутентификация устройств.*

Oshchepkov N.V., Yuzhakov A. A., Krotova E. L.

COMMUNICATION TIME INTERVALS AS A TRUST FACTOR IN THE MODBUS RTU PROTOCOL

This paper proposes a method for enhancing the security of the Modbus RTU protocol by introducing a time-based contract mechanism and HMAC authentication. The lack of built-in security features in the Modbus protocol makes ICS systems vulnerable to man-in-the-middle attacks and node spoofing. The proposed method is based on pre-negotiating data exchange time intervals between the master and slave devices with cryptographic verification based on HMAC. The method ensures compatibility with existing infrastructure and does not require significant computing resources, making it suitable for industrial controllers with limited capabilities. Experimental testing confirmed the method's effectiveness in preventing unauthorized access with minimal overhead.

Keywords: *Modbus RTU, ICS information security, HMAC, time-based contracts, device authentication.*

Введение

В связи с активным внедрением устройств, обеспечивающих автоматизацию различных технологических процессов, особую актуальность приобретает проблема доверия к сценариям функционирования вычислительных систем. Как известно, полностью защищённых информационных систем

не существует [1], что обусловливает необходимость реализации мер, направленных на снижение вероятности и последствий инцидентов информационной безопасности.

Протокол Modbus на протяжении десятилетий остаётся одним из наиболее распространённых средств обмена данными в промышленной автоматике. Простота его реализации, открытость спецификаций и совместимость с широким спектром оборудования способствуют его активному применению в различных отраслях промышленности [2]. Поэтому, согласно статистике, широко распространён и активно применяется в промышленной автоматизации в качестве основного или вспомогательного протокола связи [3].

Однако использование Modbus в режиме RTU сопровождается рядом существенных ограничений с точки зрения информационной безопасности. Отсутствие механизмов верификации узлов и защиты целостности сообщений создаёт предпосылки для реализации атак типа «человек посередине» и «подмена данных» [4, 5], что может представлять угрозу для надёжности функционирования критически важных систем. Даже при наличии физических ограничений доступа к инфраструктуре, использующей устройства автоматизации, сохраняется риск компрометации системы.

Протокол Modbus использует архитектуру «ведущий-ведомый» (master-slave) [6], в которой ведущее устройство инициирует запросы, а ведомые устройства отвечают на них. Базовая версия протокола Modbus определяет набор функций, обеспечивающих обмен данными между устройствами. При этом в спецификации протокола предусмотрены также зарезервированные коды функций, не имеющие программной реализации и предназначенные для пользовательского расширения. Такая архитектура делает протокол Modbus гибким и позволяет адаптировать его под специфические задачи, включая интеграцию дополнительных механизмов защиты и аутентификации.

Следует отметить, что спецификация протокола Modbus допускает использование широковещательных (broadcast) запросов, адресуемых всем узлам сети. В таких случаях в качестве идентификатора ведомого устройства используется значение 0 (нулевой slave ID). Особенность данного режима заключается в том, что он не позволяет однозначно идентифицировать конкретного получателя сообщения, что создаёт дополнительные требования к механизмам аутентификации.

Основные уязвимости

Ниже представлены основные уязвимости протокола, которые предполагается парировать на основании нового предложенного метода, а именно отсутствие:

• **аутентификации**. Протокол не содержит встроенных механизмов проверки подлинности отправителя и получателя сообщений. Любое устройство, подключённое к сети, может инициировать запросы или отвечать на них от имени легитимного узла

• **шифрования**. Все данные передаются в открытом виде, что позволяет злоумышленнику перехватывать и анализировать содержимое сообщений

• **контроля целостности**. Хотя протокол использует контрольную сумму CRC16 для обнаружения ошибок передачи, этот механизм не защищает от преднамеренной модификации данных злоумышленником, который может пересчитать CRC для изменённого пакета

• **защиты от повторного воспроизведения (replay attacks)**. Перехваченное легитимное сообщение может быть повторно отправлено в сеть для выполнения несанкционированных действий

Рассматриваемые типы атак на Modbus RTU

Рассмотрим в дальнейшем следующие виды возможных атак в промышленной системе:

Атака типа «подмена узла» (spoofing) [5]. Злоумышленник может выдавать себя за легитимное ведущее или ведомое устройство, отправляя команды или ложные ответы.

Атака типа «человек посередине» (MITM) [6]. Злоумышленник перехватывает коммуникацию между мастером и слейвом, имея возможность модифицировать, задерживать или подменять сообщения.

Атака повторного воспроизведения (replay attack). Перехваченные легитимные команды повторно отправляются для выполнения несанкционированных действий.

С учетом особенности функционирования протокола Modbus где в качестве двух взаимодействующих типов устройств действуют «ведущее устройство» (мастер), отправляющее команды, и «ведомое устройство» (слейв), принимающее команды, предлагается возложить инициативу на создание

контракта взаимодействия между устройствами на ведущее устройство.

Описание эксперимента

Для анализа возможностей реализации предлагаемого метода увеличения защищенности и получения оценок был проведен эксперимент с использованием программной реализации master на языке Python и библиотеки pyModbus и slave в виде запрограммированного микроконтроллера. В ходе эксперимента были произведены оценки задержек внутри сети. На рисунке 1 представлена схема подключения микроконтроллера на базе ESP32 к ПК на операционной системе Windows 10 через интерфейс RS-485 [7].

Оценка задержки производилось на основе запрограммированной пользовательской функции. Суть функции заключается в том, что мастер формирует кадр данных с длиной 4 байта: адрес slave, код функции и 2 байта CRC16 [8]. В момент отправки мастер фиксирует время отправки. Slave, получив фрейм, добавляет в кадр количество времени, затраченное на вычисления при обработ-

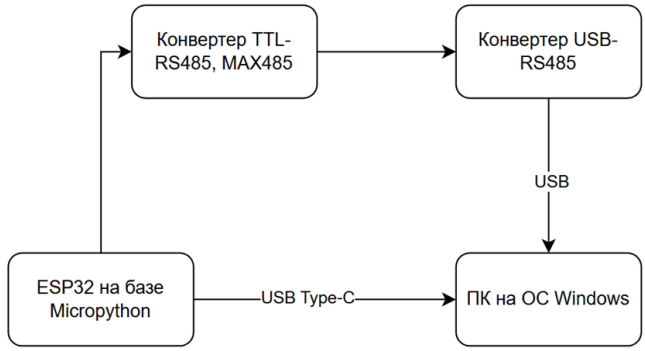

Рис. 1. Схема экспериментальной установки с подключением микроконтроллера к ПК по интерфейсу RS-485

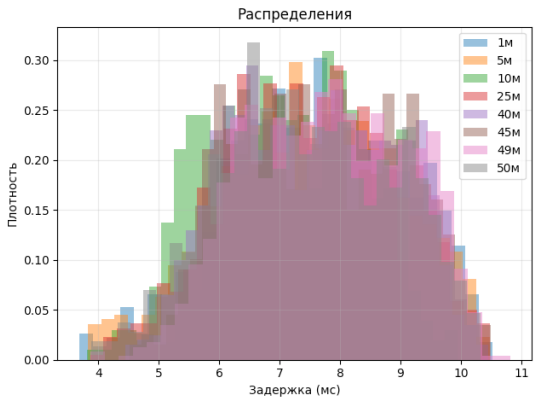

Рис. 2. Распределение задержек доставки кадра до слейва при скорости 9600 бит/с

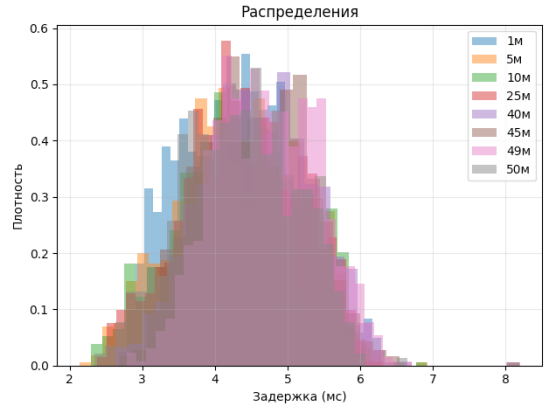

Рис. 3. Распределение задержек доставки кадра до слейва при скорости 19200 бит/с

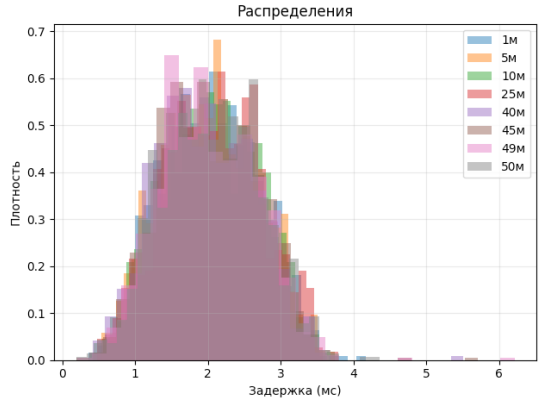

Рис. 4. Распределение задержек доставки кадра до слейва при скорости 115200 бит/с

Параметр	1м	5м	10м	25м	40м	45м	49м	50м
Среднее	7.59	7.47	7.08	7.50	7.55	7.63	7.78	7.43
Медиана	7.62	7.49	7.08	7.51	7.57	7.61	7.80	7.37
σ	1.42	1.43	1.25	1.34	1.32	1.32	1.30	1.40
Мин	3.68	3.83	3.82	3.87	4.45	4.52	3.88	3.89
Макс	10.53	10.48	9.95	10.48	10.46	10.50	10.81	10.32

Рис. 5. Сводная таблица значений для различных длин кабеля при скорости 9600 бит/с

Параметр	1м	5м	10м	25м	40м	45м	49м	50м
Среднее	4.24	4.34	4.38	4.38	4.58	4.54	4.71	4.32
Медиана	4.25	4.34	4.40	4.37	4.59	4.53	4.69	4.33
σ	0.76	0.83	0.84	0.79	0.75	0.73	0.70	0.78
Мин	2.67	2.13	2.29	2.38	2.62	2.69	2.79	2.34
Макс	6.29	6.93	6.92	6.33	8.20	8.20	6.71	6.64

Рис. 6. Сводная таблица значений для различных длин кабеля при скорости 19200 бит/с

Параметр	1м	5м	10м	25м	40м	45м	49м	50м
Среднее	2.05	2.02	2.04	2.08	1.96	2.01	2.01	2.07
Медиана	2.04	2.00	2.03	2.08	1.93	1.96	1.96	2.07
σ	0.63	0.64	0.64	0.66	0.66	0.66	0.66	0.66
Мин	0.36	0.54	0.43	0.47	0.41	0.19	0.20	0.20
Макс	4.17	3.62	3.75	3.79	5.51	5.71	6.21	4.36

Рис. 7. Сводная таблица значений для различных длин кабеля при скорости 115200 бит/с

ке кадра. После получения мастером ответа от slave, он рассчитывает полный период за исключением вычислений, которые поступили в кадре и делит полученное значение на 2. Это значение и является временем, через которое первый байт фрейма достигает целевого слейва. В качестве кабеля связи была использована витая пара типа CAT5e с длинами 1, 5, 10, 25, 40, 45, 49, 50 метров. Также стоит отметить, что оценка задержек производилась с различными значениями скоростей передачи (baudrate): 9600, 19200, 115200 бит/с. На рисунках 2, 3, 4 представлены графики распределения длительности полного цикла общения для 1000 измерений с различным интервалом отправки кадра.

Выше представлены сводные таблицы с характеристиками системы для каждого из битрейтов.

Для корректного применения параметрических статистических методов анализа данных необходимо было проверить гипотезу о нормальности распределения полученных измерений задержек с помощью статистического критерия.

Для проверки гипотезы о нормальности эмпирических распределений задержек применялся критерий Шапиро–Уилка [9], являю-

щийся наиболее мощным непараметрическим критерием согласия при объёме выборки $n \leq 5000$.

При объёме выборки $n = 1000$ критерий Шапиро–Уилка обладает повышенной чувствительностью и отвергает нулевую гипотезу даже при незначительных отклонениях, не имеющих практического значения. Поэтому наряду с критерием Шапиро–Уилка выполнялась оценка коэффициентов асимметрии и эксцесса, а также проверка по эмпирическому правилу трёх сигм [10]. Расчёты выполнены для трёх скоростей передачи данных (9600, 19200 и 115200 бод) и восьми длин кабеля (1, 5, 10, 25, 40, 45, 49 и 50 м), объём каждой выборки 1000 измерений.

Статистика Шапиро–Уилка для 9600 бод: $W = 0{,}9751–0{,}9913$. Коэффициент асимметрии составил от −0,203 до +0,030, что свидетельствует о практически симметричном распределении. Отрицательный эксцесс в диапазоне от −0,510 до −1,014 указывает на незначительное уплощение вершины по сравнению с нормальным законом, обусловленное дискретностью системных таймеров операционной системы и джиттером интерпретатора Python.

Результаты проверки правила трёх сигм:

• интервал ±1σ: 59,80–66,30% (теор. 68,27%, среднее $\Delta = 5{,}67\%$);

• интервал ±2σ: 96,60–98,60% (теор. 95,45%, среднее $\Delta = 2{,}17\%$);

• интервал ±3σ: 100,00% для всех выборок (теор. 99,73%, $\Delta < 0{,}27\%$).

Статистика Шапиро–Уилка для 19200 бод: $W = 0{,}9836–0{,}9935$. Асимметрия: от −0,124 до +0,280, эксцесс: от -0,682 до -0,111 - отклонения минимальны.

Результаты проверки правила трёх сигм:

• интервал ±1σ: 64,00–67,40% (теор. 68,27%, среднее $\Delta = 3{,}04\%$);

• интервал ±2σ: 96,30–97,70% (теор. 95,45%, среднее $\Delta = 1{,}64\%$);

• интервал ±3σ: 99,90–100,00% ($\Delta < 0{,}27\%$).

Статистика Шапиро–Уилка для 115200 бод: $W = 0{,}9809–0{,}9936$. Асимметрия: от −0,007 до +0,390, эксцесс: от −0,676 до +1,025. При длинах кабеля 45–49 м наблюдается незначительный положительный эксцесс, что может объясняться увеличением доли передискретизированных пакетов на высокой скорости передачи.

Результаты проверки правила трёх сигм:

• интервал ±1σ: 63,90–66,80% (теор. 68,27%, среднее $\Delta = 2{,}68\%$);

• интервал ±2σ: 96,50–97,40% (теор. 95,45%, среднее $\Delta = 1{,}54\%$);

• интервал ±3σ: 99,80–100,00% ($\Delta < 0{,}27\%$).

По результатам проверки критерием Шапиро–Уилка и правилом трёх сигм установлено, что для всех исследуемых скоростей передачи данных и длин кабеля эмпирические распределения задержек Modbus RTU демонстрируют достаточную близость к нормальному закону. Отклонения эмпирических частот от теоретических значений не превышают 8,5% для интервала ±1σ, 3,2% - для ±2σ и 0,3% - для ±3σ. Умеренный отрицательный эксцесс, наблюдаемый преимущественно при скорости 9600 бод ($\gamma_2 = −0{,}51...−1{,}01$), обусловлен системными артефактами измерительной среды и не препятствует применению параметрических статистических методов для дальнейшего анализа зависимости задержек от длины кабеля и скорости передачи данных.

Важным результатом проведённых экспериментов является отсутствие статистически значимой зависимости между длиной кабеля в исследуемом диапазоне от 1 до 50 метров и величиной задержки передачи данных. Это свидетельствует о том, что время распространения сигнала по физической среде передачи пренебрежимо мало по сравнению с другими составляющими общей задержки, такими как время обработки запроса в микроконтроллере, формирование ответа и программные задержки в мастере-устройстве.

В тоже время, анализ измерений показал, что среднее значение стандартного отклонения задержек зависит от скорости передачи данных.

Для скорости 9600 бит/с среднее стандартное отклонение составило 1,35 мс, что отражает наибольшую вариабельность времени отклика в системе. При увеличении скорости передачи до 19200 бит/с среднее стандартное отклонение снизилось до 0,77 мс, демонстрируя повышение стабильности обмена данными. Наименьшая вариабельность наблюдалась при скорости 115200 бит/с, где среднее стандартное отклонение составило 0,65 мс. Результаты представлены в таблице 1.

Снижение стандартного отклонения с ростом скорости передачи данных может быть объяснено сокращением времени передачи кадра по физическому каналу, что уменьшает влияние временных флуктуаций в системе на общую задержку. При более высоких скоро-

Таблица 1

Среднее значение задержки и стандартное отклонение при различных скоростях передачи данных

baudrate, бит/с	$\bar{x}$, мс	σ, мс
9600	7,50	1,35
19200	4,44	0,77
115200	2,03	0,65

стях передачи относительный вклад времени передачи в общую задержку уменьшается, что приводит к более предсказуемому и стабильному времени отклика системы.

Предлагаемый метод заключения контракта

Суть предлагаемого метода заключения контракта состоит в предварительной договорённости между мастером и slave о сеансах связи, в которые мастер устройство посылает команды, а slave отвечает. Таким образом, предполагается, что наличие у мастера сведений о том, в какой временной интервал отправлять посылку, может служить критерием легитимности устройства.

Ключевое ограничение: мастера, которые не договорились со slave о временном интервале связи, не отвечают ни на одну команду чтения или записи, посылаемую им от любого slave в сети.

В рамках предложенного подхода реализован метод криптографической верификации узлов, основанный на механизме HMAC (Hash-based Message Authentication Code), подробно описанном в стандарте RFC 4226 [11]. Данный механизм представляет собой метод проверки подлинности сообщений, основанный на использовании криптографической хэш-функции в сочетании с общим секретным ключом. Основная цель HMAC - обеспечение целостности передаваемых данных и подтверждение аутентичности сторон, участвующих в обмене.

В разрабатываемой реализации в качестве входных данных для HMAC используется временная метка (Unix timestamp), для которой вычисляется хэш с применением алгоритма семейства SHA-1 и индивидуального секретного ключа ведомого устройства. Результат вычисления преобразуется в массив байтов, после чего выполняется динамическое усечение. Итоговое значение форматируется в строку фиксированной длины, что обеспечивает корректное отображение возможных лидирующих нулей (например, «003451»). Для сохранения таких нулей результат кодируется в ASCII, что гарантирует точную передачу данных при обмене.

Используя широковещательные запросы, каждый из slave получит сообщение о заключении контракта и произведёт вычисление HMAC на основе своего индивидуального ключа. Данный механизм позволяет slave убедиться в том, что инициатор взаимодействия (мастер) действительно является легитимным участником сети.

В рамках системы безопасности мастер хранит полный набор секретных ключей, каждый из которых индивидуально соответствует конкретному slave. Мастера, в свою очередь, содержат свои уникальные ключи, сохранённые в энергонезависимой памяти. Формат ключей представлен в виде base32-кодированных последовательностей символов, что обеспечивает их совместимость с текстовыми интерфейсами, средствами диагностики и стандартными инструментами обработки данных.

Описание алгоритма

Изначально необходимо задать 2 базовые величины:

• T – период открытия окна (Step)

• D – длительность окна приёма сигнала (Duration)

Как было описано ранее, протокол обладает рядом кодов функций, зарезервированных для реализации пользователем. Благодаря этому была запрограммирована функция заключения контракта, кадр которой представлен на рисунке 8.

Поскольку запрос является широковещательным, ID slave – всегда 0. Благодаря включаемым в фрейм компонентам timestamp – временная метка отправки мастером сообщения и HMACm, вычисленной на основе временной метки и ключа slave, с которым мастер заключает контракт, лишь единственный slave, вычислив на своей стороне

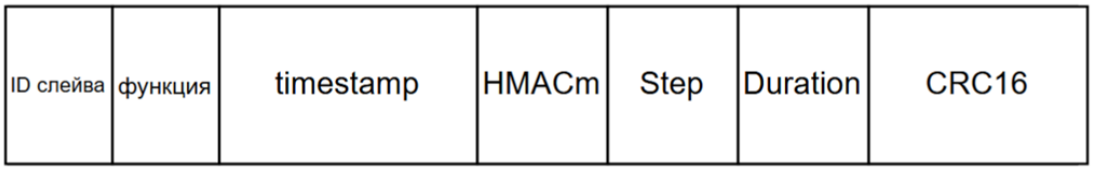

Рис. 8. Формат кадра для заключения контракта об окнах связи

HMAC(timestamp) и сравнив его с HMACm, полученным в фрейме, установит соответствие. Именно это и будет являться критерием, по которому данный slave поймет, что сообщение адресовано ему.

Получив запрос и проверив одноразовый код, slave сохраняет в памяти временную метку, в которую поступил запрос, чтобы в дальнейшем производить расчет попадания запросов мастера в окно длительности D через каждые T мс.

Оценка момента времени

Как было отмечено выше, время доставки пакета от мастера к слейву описывается нормальным распределением:

$$\tau \sim N(\mu,\sigma^2)$$

В момент времени $t_0^{(M)}$ (по локальным часам мастера) мастер отправляет сообщение синхронизации slave. Slave в свою очередь принимает фрейм в момент времени:

$$t_0^{(S)} = t_0^{(M)} + \tau_0,$$

где τ_0 – реализация случайной величины задержки для данной передачи.

Поскольку $\tau_0 \in [\mu - 3\sigma; \mu + 3\sigma]$, с высокой вероятностью оценка момента прибытия с точки зрения мастера:

$$t_0^{(S)} = t_0^{(M)} + \mu,$$

что соответствует математическому ожиданию интервала неопределенности.

Для определения момента отправки очередного запроса мастер вычисляет время, прошедшее с момента установления синхронизации:

$$\Delta t = t_{текущее}^{(M)} - t_0^{(M)}.$$

Полный период цикла рассчитывается по формуле (1):

$$C = T + D.$$

Тогда для определения момента отправки необходимо рассчитать фазу цикла (2):

$$\phi(t) = (\Delta t - \mu) \bmod C, \quad (2)$$

где вычитание μ компенсирует среднюю задержку при установлении контракта.

В свою очередь, границы, которым должна удовлетворять величина фазы соответствуют формулам (3) и (4):

$$\phi_{left} = k\sigma, \quad (3)$$

$$\phi_{right} = D - k\sigma, \quad (4)$$

где k – коэффициент безопасности.

Учитывая вышеизложенное передача разрешена при выполнении условия:

$$\phi_{left} \le \phi(t) \le \phi_{right}.$$

В условиях нестабильного сигнала и наличия в системе факторов, влияющих на задержку при обмене данными, в качестве коэффициента в (3) и (4) было выбрано число 4, а не 3, как это присуще для нормального распределения. Данный коэффициент исключает возможность выхода за границы доверительного интервала и ограничивает момент отправки на мастере. Таким образом (5), формируется ограничение для длительности окна D:

$$D > 8\sigma. \quad (5)$$

В ходе проведения эксперимента были получены уточняющие положения, которые необходимо учитывать. Поскольку два устройства являются независимыми, то каждое из них обладает своим индивидуальным таймером, по которому определяется текущая временная метка. Как известно, существует понятие дрейфа таймеров, что со временем приводит к рассинхронизации. Именно поэтому необходимо производить ресинхронизацию с устройствами каждые 2-3 мин. Это позволит избежать проблемы оценки времени на мастере и slave.

С учетом представленных ранее видов атак, в алгоритме верификации на основе гаммирования с использованием секретного ключа, будут рассмотрены следующие сценарии атаки:

1. Злоумышленник отправляет кадр на slave, которое не заключило контракт.

2. Злоумышленник не знает об алгоритме гаммирования кадра и отправляет запрос на устройство, которое заключило контракт.

3. Злоумышленник знает алгоритм гаммирования кадра, но не знает ключ и временные интервалы.

В первом случае, поскольку злоумышленник отправляет кадр до того, как легитимный мастер заключил его, то ведомое устройство не отправляет ответ на запрос.

При реализации 2 сценария, необходимо предпринять следующие действия:

1. Изначально заключить контракт с использованием легитимного мастера.

2. Произвести подмену легитимного мастера на мастер злоумышленника.

3. Осуществить попытку отправки кадра.

Поскольку злоумышленник не знает об алгоритме гамирования, отправленная им команда не будет исполнена.

Реализация третьего сценария атаки может быть выполнена аналогично второму сценарию. Поскольку в данном этапе происходит двухфакторная верификация мастера как по гамирующему ключу, который может принимать значения [32768;65535], так и по временному окну, то можно вывести формулу расчета вероятности попадания кадра злоумышленника с угаданным ключом:

$$P = \frac{D}{(T+D)*32768}.$$

На рисунке 9 представлен график функции f(T,D) = D / (T+D). Из графика видно, что уменьшение величины D, и увеличение величины T приводит к уменьшению значения функции. Из этого следует, что чем реже возникает окно и чем оно короче, тем меньше вероятность реализации атаки подмены.

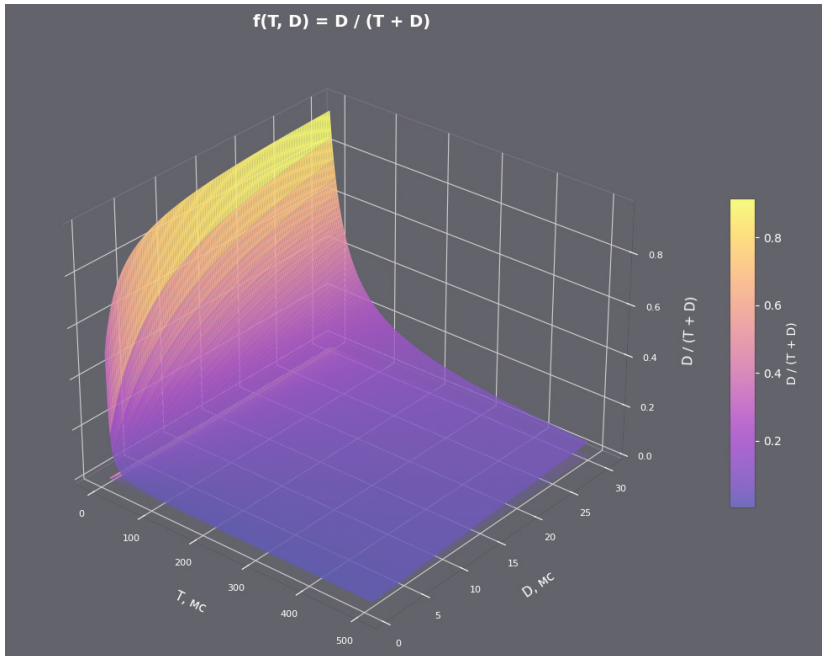

Рис. 9. График функции f(T,D) = D / (T+D)

Данная составляющая многоуровневой защиты slave от исполнения команд, полученных от мастера злоумышленника, сокращает вероятность успешной реализации атаки в 32768 раз, где 32768 – количество возможных ключей, используемых slave.

В разработанной модификации протокола первым этапом происходит применение функции гаммирования, а затем определение, попал ли кадр в заданный интервал. Благодаря этому, злоумышленник не определит, где он допустил ошибку: либо допущена ошибка при выборе ключа, либо некорректный момент отправки кадра. Именно это является двухэтапным методом проверки, что отправитель – легитимное устройство.

В качестве эксперимента для скорости 9600 бод/с и длины кабеля 10м были выбраны T = 250мс, D = 30мс. В случае, если верификация происходит только при гаммировании, то вероятность подбора ключа и исполнения команды равна 2^{15}. Та же атака, но при введении дополнительного этапа верификации по окнам связи снижает вероятность в $\frac{250+30}{30} = 9{,}33$ раза.

Выводы

Введение двухэтапной верификации, комбинирующей криптографическую аутентификацию через гаммирование и проверку посредством временных окон связи, фундаментально трансформирует модель угрозы в распределенных системах на базе протокола Modbus. При одноэтапной проверке злоумышленник решает задачу поиска в одномерном пространстве ключей мощностью N равной 32768. Двухэтапная схема переводит задачу атаки в двумерное пространство параметров, где необходимо одновременно подобрать корректный ключ и обеспечить попадание во временное окно длительностью D

внутри цикла C равного T+D. Статистическая независимость этих параметров даёт вероятность успешной атаки равную $\frac{D}{(T+D)*32768}$. Для экспериментальных параметров T равного 250 миллисекунд и D равного 30 миллисекунд достигается снижение вероятности атаки в 9.33 раза. Критическое свойство схемы заключается в принципе неразличимости ошибок. Система не предоставляет информации о причине отказа, формируя идентичный ответ при провале любой из проверок. Это лишает злоумышленника возможности декомпозиции задачи на последовательные этапы поиска временного окна и подбора ключа. Атакующий вынужден решать совместную задачу оптимизации без возможности адаптивной корректировки стратегии, что сохраняет мультипликативный характер защиты вместо аддитивного. Оптимальный выбор параметров обеспечивает коэффициент усиления защиты для промышленных сетей, что соответствует снижению вероятности успешной атаки на порядок величины при сохранении приемлемой пропускной способности канала.

Литература

1. Защита АСУ ТП: от теории к практике: сайт InformationSecurity. [Электронный ресурс]. – Режим доступа : https://lib.itsec.ru/articles2/Oborandteh/zaschita-asu-tp-ot-teorii-k-praktike (Дата обращения 01.03.2026)

2. Claudio Urrea, Claudio Morales, Enhancing Modbus-RTU Communications for Smart Metering in Building Energy Management Systems. Wiley. Online Library. [Электронный ресурс]. – Режим доступа : https://onlinelibrary.wiley.com/doi/full/10.1155/2019/7010717 (дата обращения 25.02.2026)

3. Industrial Network Market Analysis: сайт EBYTE. [Электронный ресурс]. – Режим доступа : https://www.cdebyte.com/news/1101 (дата обращения 01.03.2026)

4. Alakbarov R.G. Application and Security Issues of Internet of Things in Oil-Gas Industry // Alakbarov R.G., Hashimov M.A. // International Journal of Education and Management Engineering. – 2018. – №6, с. 24–36. https://doi.org/10.5815/ijeme.2018.06.03

5. Арзуманян Э.А. // МИТМ-атака. Угроза информационной безопасности в РФ // Арзуманян Э.А., Чумаков А.А. // Znanstvena Misel. – 2019. – № 8-1(33), с. 37-40.

6. Ведущий — ведомый: Википедия. Свободная энциклопедия. [Электронный ресурс]. – Режим доступа: https://ru.wikipedia.org/wiki/Ведущий_—_ведомый (дата обращения 01.03.2026)

7. RS-485: Википедия. Свободная энциклопедия. [Электронный ресурс]. – Режим доступа: https://ru.wikipedia.org/wiki/RS-485 (дата обращения 01.03.2026)

8. Циклический избыточный код: Википедия. Свободная энциклопедия. [Электронный ресурс]. – Режим доступа: https://ru.wikipedia.org/wiki/Циклический_избыточный_код (дата обращения 02.03.2026)

9. Shapiro, S. S. An analysis of variance test for normality (complete samples) // Shapiro, S. S. Wilk, M. B. // Biometrika. – 1965, T. 52, с. 591–611.

10. Gauss, C. F. Theoria motus corporum coelestium in sectionibus conicis solem ambientium // Cambridge University Press. – 2012.

11. RFC 4226: RFC Editor. [Электронный ресурс]. – Режим доступа: https://www.rfc-editor.org/info/rfc4226 (дата обращения 02.03.2026)

References

1. Zashchita ASU TP: ot teorii k praktike: sayt InformationSecurity. [Elektronnyy resurs]. – Rezhim dostupa: https://lib.itsec.ru/articles2/Oborandteh/zaschita-asu-tp-ot-teorii-k-praktike (Data obrashcheniya 01.03.2026)

2. Claudio Urrea, Claudio Morales, Enhancing Modbus-RTU Communications for Smart Metering in Building Energy Management Systems. Wiley. Online Library. [Электронный ресурс]. – Режим доступа : https://onlinelibrary.wiley.com/doi/full/10.1155/2019/7010717 (дата обращения 25.02.2026)

3. Industrial Network Market Analysis: сайт EBYTE. [Электронный ресурс]. – Режим доступа : https://www.cdebyte.com/news/1101 (дата обращения 01.03.2026)

4. Alakbarov R.G. Application and Security Issues of Internet of Things in Oil-Gas Industry // Alakbarov R.G., Hashimov M.A. // International Journal of Education and Management Engineering. – 2018. – №6, с. 24–36. https://doi.org/10.5815/ijeme.2018.06.03

5. Arzumanyan E.A. // MITM-ataka. Ugroza informatsionnoy bezopasnosti v RF // Arzumanyan E.A., Chumakov A.A. // Znanstvena Misel. – 2019. – № 8-1(33), s. 37-40.

6. Vedushchiy — vedomyy: Vikipediya. Svobodnaya entsiklopediya. [Elektronnyy resurs]. – Rezhim dostupa: https://ru.wikipedia.org/wiki/Vedushchiy_—_vedomyy (data obrashcheniya 01.03.2026)

7. RS-485: Vikipediya. Svobodnaya entsiklopediya. [Elektronnyy resurs]. – Rezhim dostupa: https://ru.wikipedia.org/wiki/RS-485 (data obrashcheniya 01.03.2026)

8. Tsiklicheskiy izbytochnyy kod: Vikipediya. Svobodnaya entsiklopediya. [Elektronnyy resurs]. – Rezhim dostupa: https://ru.wikipedia.org/wiki/Tsiklicheskiy_izbytochnyy_kod (data obrashcheniya 02.03.2026)

9. Shapiro, S. S. An analysis of variance test for normality (complete samples) // Shapiro, S. S. Wilk, M. B. // Biometrika. – 1965, T. 52, c. 591–611.

10. Gauss, C. F. Theoria motus corporum coelestium in sectionibus conicis solem ambientium // Cambridge University Press. – 2012.

11. RFC 4226: RFC Editor. [Электронный ресурс]. – Режим доступа: https://www.rfc-editor.org/info/rfc4226 (дата обращения 02.03.2026)

ЮЖАКОВ Александр Анатольевич, доктор технических наук, профессор, заведующий кафедрой Автоматика и телемеханика, Федеральное государственное автономное образовательное учреждение высшего образования «Пермский национальный исследовательский политехнический университет». 614990, Пермский край, г. Пермь, Комсомольский проспект, д. 29. E-mail: uz@at.pstu.ru

КРОТОВА Елена Львовна, кандидат физико-математических наук, доцент кафедры Высшей математики, Федеральное государственное автономное образовательное учреждение высшего образования «Пермский национальный исследовательский политехнический университет». 614990, Пермский край, г. Пермь, Комсомольский проспект, д. 29. E-mail: lenkakrotova@yandex.ru

ОЩЕПКОВ Никита Владимирович, аспирант кафедры Автоматика и телемеханика, Федеральное государственное автономное образовательное учреждение высшего образования «Пермский национальный исследовательский политехнический университет». 614990, Пермский край, г. Пермь, Комсомольский проспект, д. 29. E-mail: maserati_2000@mail.ru

YUZHAKOV Alexander Anatolyevich, Doctor of Technical Sciences, Professor, Head of the Department of Automation and Telemechanics, Federal State Autonomous Educational Institution of Higher Education “Perm National Research Polytechnic University”. 614990, Perm Region, Perm, Komsomolsky Prospekt, 29. E-mail: uz@at.pstu.ru

KROTOVA Elena Lvovna, Candidate of Physical and Mathematical Sciences, Associate Professor of the Department of Higher Mathematics, Federal State Autonomous Educational Institution of Higher Education “Perm National Research Polytechnic University”. 614990, Perm Region, Perm, Komsomolsky Prospekt, 29. E-mail: lenkakrotova@yandex.ru

OSHCHEPKOV Nikita Vladimirovich, postgraduate student of the Department of Automation and Telemechanics, Federal State Autonomous Educational Institution of Higher Education “Perm National Research Polytechnic University”. 614990, Perm Region, Perm, Komsomolsky Prospekt, 29. E-mail: maserati_2000@mail.ru