Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение высшего образования
«Московский государственный технический университет имени Н. Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н. Э. Баумана)

ФАКУЛЬТЕТ «Информатика и системы управления» (ИУ)
КАФЕДРА «Информационная безопасность» (ИУ8)

Постквантовая криптография: методы, сложности, стандартизация и направления дальнейших исследований

по дисциплине «Криптографические методы защиты информации»

Студент ИУ8-114 (Группа) Мильченко И. Д. (И. О. Фамилия) _____ (Подпись, дата)

Преподаватель Жуков А. Е. (И. О. Фамилия) _____ (Подпись, дата)

Оценка: _____

Москва, 2025 г.

СОДЕРЖАНИЕ

1 Введение

В современном мире огромную роль играют коммуникации. Сегодняшнее общество опирается на Интернет как на базовый механизм любой дистанционной взаимосвязи. Это порождает необходимость защищать передаваемые данные и обеспечивать их конфиденциальность. Криптография — это область, полностью посвящённая защите информации, где множество исследователей разрабатывают и анализируют алгоритмы, позволяющие сохранять приватность и целостность данных.

Постепенно формируется новая парадигма вычислений — квантовые компьютеры, которые способны радикально изменить возможности классических вычислительных систем, используемых сегодня. Квантовый компьютер сможет выполнять отдельные задачи, недоступные даже современным высокопроизводительным многопроцессорным системам. Концепция квантовых вычислений основана на квантовой физике — на понимании поведения систем на субатомном уровне, заложенном в начале XX века такими учёными, как Шрёдингер (Schrödinger), Бор (Bohr), Гейзенберг (Heisenberg), Эйнштейн (Einstein) и др. Позднее, в 1980-е годы, идеи и математический аппарат квантовой физики были применены для моделирования вычислительных устройств, способных выполнять определённые задачи значительно быстрее классических компьютеров. Квантовые компьютеры используют в качестве входных данных квантовые биты и формируют результат, опираясь на принципы квантовой механики. Такие компьютеры способны решать задачи, которые классические вычислительные системы не могут выполнить за разумное время. Их потенциал распространяется на вычисления, недостижимые даже для современных суперкомпьютеров. Области применения чрезвычайно широки: навигационные системы, сейсмология, физические исследования, фармацевтика и многое другое [1]. Квантовые компьютеры позволяют решать сложные научные задачи, которые остаются нерешёнными для самых мощных суперкомпьютеров. Они станут революционным фактором и для искусственного интеллекта благодаря огромному приросту вычислительной мощности. Таким образом, потенциальные области применения квантовых вычислений практически безграничны.

Однако у любой технологии есть две стороны. С одной стороны, квантовые компьютеры дают надежду быстро решать множество сложных задач из самых разных областей. С другой — их появление может создать серьёзные угрозы. Один из наиболее опасных сценариев связан с криптоанализом.

Криптоанализ изучает методы, позволяющие восстановить содержимое зашифрованных данных без знания секретной информации, используемой отправителем и получателем для зашифрования и расшифрования. Иначе говоря, криптоанализ — это искусство взлома криптосистем. К сожалению, многие математические задачи, на которых основана безопасность современных криптографических алгоритмов, окажутся решаемыми на квантовых компьютерах. Это резко упростит взлом криптографических схем, широко применяемых в цифровых коммуникациях.

К широко используемым симметричным алгоритмам относятся Advanced Encryption Standard (AES) и 3DES (Data Encryption Standard). Их стойкость будет фактически снижена вдвое благодаря алгоритму поиска Гровера [2]. Асимметричные алгоритмы — такие как RSA, схема Диффи–Хеллмана (Diffie–Hellman) и криптография на эллиптических кривых (ECC) — опираются на вычислительную сложность задач факторизации и дискретного логарифмирования. Квантовые компьютеры способны решать эти задачи практически мгновенно, что делает безопасность асимметричных криптосистем крайне уязвимой.

Постквантовая криптография — это направление, изучающее криптосистемы, устойчивые как к классическим, так и к квантовым атакам. Эти криптосистемы строятся на задачах, которые, как считается, неразрешимы ни для классических, ни для квантовых компьютеров. Основные семейства постквантовых криптосистем включают: криптографию на решётках, криптографию на изогениях, некоммутативную криптографию, кодовые криптосистемы, схемы на основе хэширования и многомерную криптографию [3].

Необходимость разработки более совершенных криптографических алгоритмов привела к появлению инициатив NIST по созданию квантовоустойчивых стандартов. В рамках программы стандартизации были собраны заявки и предложения со всего мира, после чего началось многоэтапное обсуждение и экспертиза для выбора наиболее надёжных схем.

В данной работе рассматриваются последствия появления квантовых компьютеров для существующих криптографических алгоритмов, анализируется их устойчивость перед квантовым противником. Также изучаются основные направления и наиболее исследованные постквантовые криптосистемы, потенциально устойчивые даже к квантовым атакам. Завершает обзор анализ наиболее перспективных кандидатов на включение в стандарт NIST и обсуждение направлений дальнейших исследований в области постквантовой криптографии.

В 2018 году Мавроэидис (Mavroeidis) и соавт. [4] опубликовали работу, посвящённую влиянию квантовых вычислений на современную криптографию. В ней рассматривались четыре из шести основных семейств постквантовой криптографии, а также подробно анализировались подписи, основанные на хэшировании. Настоящая работа расширяет этот обзор: в ней представлены математические трудные задачи, лежащие в основе всех семейств постквантовых схем, кратко описаны ключевые представители каждого семейства, перечислены финалисты NIST PQC, а также намечены перспективные направления дальнейших исследований.

2 Криптография

Криптография — это наука, изучающая методы сокрытия сообщений с использованием математических средств. Содержание сообщения скрывается посредством криптографического алгоритма шифрования, который преобразует исходные данные таким образом, чтобы сделать их непонятными для постороннего наблюдателя. Полученные скрытые данные называются шифртестом. Исходное сообщение называют открытым текстом или plaintext. Шифртест может быть вновь преобразован в исходное сообщение с помощью алгоритма расшифрования. Конкретная реализация криптографического метода называется криптосистемой. Криптоанализ — это наука, изучающая методы анализа и взлома криптосистем, которые считаются защищёнными.

Криптография предоставляет пользователям широкий набор сервисов информационной безопасности. Во-первых, она обеспечивает конфиденциальность данных: шифрование делает информацию недоступной для всех, кроме обладателей секретного ключа (предполагаемых получателей), которые могут восстановить исходный текст. Во-вторых, она обеспечивает целостность данных: криптографические механизмы позволяют удостовериться, что информация не была изменена третьей стороной. В-третьих, криптография предоставляет средства аутентификации — отправитель и получатель могут подтвердить личности друг друга и убедиться, что не взаимодействуют со злоумышленником. Кроме того, она позволяет доказать подлинность личности субъекта. Наконец, криптография обеспечивает невозможность отказа от авторства (non-repudiation), то есть предоставляет механизм подтверждения того, что сообщение действительно было отправлено конкретным участником.

Криптография имеет множество прикладных применений [5]. Жёсткие диски компьютеров содержат большое количество деловой, военной и личной конфиденциальной информации, утечка которой может привести к серьёзным последствиям. Криптографические методы используются на устройствах хранения данных для обеспечения их защищённости даже при физическом доступе злоумышленника к носителю.

Широко применяется криптография и в компьютерных сетях. Она служит для аутентификации, создания цифровых подписей пользователей и отметки времени важных документов. Цифровые платёжные системы — такие как Google Pay и Paytm — основываются на криптографических механизмах для защиты банковских данных пользователей.

Шифрование и расшифрование используются в электронной почте, в системах мгновенного обмена сообщениями, таких как WhatsApp, а также в социальных сетях — например, Instagram и Facebook.

2.1 Симметричная криптография

Симметричная криптография – это раздел криптографии, в рамках которого и шифрование, и расшифрование выполняются с использованием одного и того же ключа. При условии, что длина ключа достаточно велика, а сам ключ является случайным, перебор всех возможных ключей выходит за пределы возможностей современных вычислительных систем.

Однако из-за того, что вся безопасность схемы основана на одном секретном ключе, критически важным аспектом симметричной криптографии является безопасная передача ключа от отправителя к получателю.

2.1.1 Data Encryption Standard (DES)

Data Encryption Standard (DES) — один из наиболее известных криптографических алгоритмов, уязвимость которого к полному перебору ключа была доказана ещё в доквантовую эпоху. Тем не менее увеличение длины ключа позволяет частично компенсировать эту уязвимость, что привело к появлению модифицированных схем, таких как Triple-DES (3DES), который может работать с двумя или тремя ключами, экспоненциально увеличивая количество операций, необходимых для полного перебора.

DES основан на композиции преобразований Фейстеля, в которой сообщение делится на две половины, проходящие чередующиеся преобразования в каждом раунде. [6]. Каждый блок сообщения имеет длину 64 бита, при это в каждом раунде обрабатывается 32 бита. В следующем раунде необработанные биты меняются местами с обработанными, и процесс повторяется.

Алгоритм DES включает 16 таких раундов, которым предшествует начальная перестановка (Initial Permutation, IP) и завершает всё конечная перестановка (Final Permutation, FP) — обратная к IP.

Фейстелева функция DES включает следующие операции:

а) **Расширение.** Входной 32-битовый блок преобразуется в 48-битовый путём дублирования и добавления первых и последних бит соседних 4-битовых блоков к текущему блоку.

б) **Смешивание с ключом.** Каждый раунд использует собственный 48-битовый подключ, порождаемый из исходного ключа с помощью алгоритма выработки подключей. Результат операции расширения складывается по модулю два (XOR) с подключом.

в) **Подстановка.** Полученный 48-битовый блок разбивается на восемь 6-битовых блоков, каждый из которых проходит через нелинейные преобразования в S-блоках. Таблицы S-блоков определены стандартом NIST. Каждый S-блок отображает 6 бит во входе в 4 бита на выходе, формируя итоговый 32-битовый блок для следующего раунда.

г) **Перестановка.** Выходы S-блоков переставляются согласно P-таблице, что обеспечивает распределение выходов каждого S-блока по четырём различным S-блокам в следующем раунде.

Алгоритм DES обеспечивает необходимую степень рассеивания и перемешивания благодаря S- и P-блокам, что делает выходной шифртест псевдослучайным. В таких условиях единственным практическим способом взлома является полный перебор ключа.

На сегодняшний день DES считается небезопасным. Алгоритм был впервые взломан в 1997 году в рамках проекта DESCHALL, а рост вычислительных мощностей привёл к тому, что время перебора стало практически ничтожным. В 2017 году был продемонстрирован наиболее быстрый взлом DES в модели выбранного открытого текста с использованием радужных таблиц, позволивший получить ключ менее чем за 25 секунд.

Для увеличения эффективной длины ключа и усложнения перебора была разработана усиленная схема — алгоритм 3DES.

2.1.2 3-DES

Тройной DES (3DES) был предложен как решение уязвимости DES, связанной с малой длиной ключа (56 бит), путём «расширения» эффективного размера ключа за счёт использования трёх различных независимых ключей K_1, K_2, K_3, каждый длиной 56 бит.

Шифрование открытого текста выполняется по схеме «Шифрование–Расшифрование–Шифрование» (EDE). Сначала открытый текст шифруется DES с ключом K_1. Затем промежуточный результат расшифровывается DES с ключом K_2. Наконец, полученный текст снова шифруется DES с ключом K_3, формируя шифртест [6].

В зависимости от того, как выбираются ключи, можно увеличить сложность атаки полного перебора. Если использовать три независимых значения K_1, K_2, K_3, то достигается максимальная стойкость 3DES, эквивалентная эффективной длине ключа 168 бит.

Тем не менее и эта схема оказалась уязвимой из-за малого размера блока, что увеличивает вероятность коллизий. Эта слабость была использована в атаке *Sweet32* [7], в рамках которой коллизии могут быть найдены примерно за 2^{36*2} попыток.

2.1.3 Advanced Encryption Standard (AES)

AES [8] является одним из наиболее широко применяемых симметричных криптографических алгоритмов. Его стойкость основана на псевдослучайности шифртеста, формируемого в результате многократного применения стандартизованной сети подстановки–перестановки (SP-сети). На практике используются версии AES с длинами ключа 128, 192 и 256 бит, которым соответствуют 10, 12 и 14 раундов, каждый из которых состоит из четырёх преобразований. Подключи генерируются с помощью алгоритма выработки подключей. Необходимые уровни запутывания и рассеивания достигаются многократными итерациями подстановок и смешивания.

Каждый раунд AES включает четыре преобразования:

а) **SubByte.** Каждый байт состояния заменяется значением из заранее определённого нелинейного соответствия. Подстановка осуществляется через фиксированный S-блок, содержащий 256 элементов.

б) **ShiftRows.** Строки состояния циклически сдвигаются согласно правилам, указанным в стандарте NIST. При необходимости допускаются модифицированные варианты сдвига.

в) **MixColumns.** Каждая колонка интерпретируется как полином и умножается на кодирующий полином с последующим приведением результата по модулю неприводимого полинома над полем Галуа $\mathbb{GF}(2^8)$.

г) **AddRoundKey.** К массиву состояния применяется операция XOR с раундовым подключом, порождённым по расписанию ключей.

Все операции выполняются над полем Галуа $\mathbb{GF}(2^8)$. Алгоритм расшифрования повторяет те же шаги, что и шифрование, но в обратном порядке, используя инверсные операции.

Хотя выявлены уязвимости, связанные с повторяющимися структурами открытого текста или некорректным выбором ключа, доказано, что эти атаки не уменьшают сложность до уровня, позволяющего осуществить полный взлом. С учётом современных вычислительных возможностей AES считается стойким. На сегодняшний день AES является наиболее широко применяемым симметричным шифром в проводных и беспроводных протоколах безопасности.

2.2 Асимметричная криптография

Асимметричная криптография — это направление криптографии, в котором ключ разделён на две части: открытый ключ и закрытый ключ. Открытый ключ передаётся отправителю, который использует его для шифрования данных, тогда как закрытый ключ применяется получателем для расшифрования. Такой подход обеспечивает конфиденциальность и подлинность сообщений.

2.2.1 RSA

RSA — это публичная криптосистема, широко используемая для безопасной передачи данных. Она была предложена Рональдом Райвестом (Rivest), Ади Шамиром (Shamir) и Леонардом Адлеманом (Adleman) в 1977 году.

Алгоритм основан на задаче факторизации целого числа. Наиболее эффективный классический алгоритм факторизации — General Number Field Sieve — имеет субэкспоненциальную сложность. RSA использует этот факт, что делает алгоритм вычислительно стойким.

Алгоритм опирается на однонаправленные функции: их легко вычислять, но трудно обращать (например, перемножить два простых числа легко, а вот разложить произведение на множители — сложно). Теорема о распределении простых чисел и оценка Чебышёва гарантируют, что среди чисел длиной 200 десятичных цифр приблизительно 1/1000 являются простыми. Поэтому пользователи RSA могут сравнительно просто находить большие простые числа. Для проверки простоты используется тест Ферма, который даёт вероятностный результат.

Тест простоты Ферма утверждает, что если число n простое, то для случайного a выполняется равенство $a^{n-1} \equiv 1 \pmod{n}$. При повторении теста для большого количества различных значений a и получении результата 1 вероятность того, что n является простым, возрастает. Если число составное, тест Ферма проваливается как минимум для половины значений из $\mathbb{Z}_n$. Поэтому каждая успешная проверка уменьшает вероятность ошибки примерно вдвое.

Для организации защищённого обмена данными получатель генерирует пару ключей. Он выбирает два больших простых числа p, q и вычисляет $n = p \cdot q$. Затем вычисляется $L = (p-1)(q-1)$. Выбирается число e, взаимно простое с L. Пара (e, n) публикуется как открытый ключ. Закрытый ключ определяется как $d = e^{-1} \pmod{(p-1)(q-1)}$.

Зашифрование выполняется отправителем по формуле $C = m^e \bmod n$, а расшифрование — получателем по формуле $m = C^d \bmod n$.

Для больших размеров ключей RSA (1024 бита и выше) не существует эффективных методов решения задачи факторизации. Все практические атаки на RSA основаны на попытке разложения n на простые множители [9]. Из-за субэкспоненциальной сложности факторизации взлом RSA при длине ключа 2048 бит считается нереалистичным при современном уровне вычислительной мощности. По этой причине RSA широко используется в разных протоколах и системах, требующих асимметричной криптографии.

2.2.2 Протокол обмена ключами Диффи – Хеллмана

Протокол обмена ключами Диффи – Хеллмана (DHKE) основан на задаче дискретного логарифма. Эта задача представляет собой одностороннюю функцию: прямое вычисление легко, а обратное чрезвычайно сложно. Для дискретного логарифма выбирается большое простое число N. В группе $\mathbb{Z}_N$ определяется генератор g, степени которого пробегают через все элементы группы [10].

Возведение g в степень p в $\mathbb{Z}_N$ вычисляется легко, однако найти p, зная N, g и значение o, крайне трудно. Математически: $o = g^p \bmod N$. Зная g, p и N, легко вычислить o, но зная лишь o, g и N, трудно определить $p = \mathrm{dlog}_g(o)$. Здесь dlog означает дискретный логарифм.

Решение задачи дискретного логарифма также имеет субэкспоненциальную сложность. Поскольку безопасность DHKE опирается на трудность этой задачи, взлом данного протокола требует экспоненциальных ресурсов.

Протокол использует два общедоступных параметра: простое число p и генератор q группы $\mathbb{Z}_p$. Стороны выбирают секретные случайные значения a и b и вычисляют открытые ключи: $A = q^a \bmod p$ и $B = q^b \bmod p$.

Значения A и B публикуются. Каждая сторона затем вычисляет общий ключ: $X = B^a \bmod p$ и $X = A^b \bmod p$. Оба выражения дают одно и то же значение X, которое используется далее в качестве секретного ключа в симметричных криптографических алгоритмах.

Протокол DHKE не обеспечивает аутентификацию. Поэтому возможна атака «человек посередине» (Man-in-the-middle), при которой злоумышленник устанавливает два разных общих ключа — с Алисой и с Бобом — выдавая себя за другую сторону, перехватывая, изменяя и пересылая сообщения.

При неправильном выборе параметров возможна атака с использованием малых подгрупп. Существуют и другие виды атак, использующие особенности выбора простых чисел. Тем не менее при корректной модернизации инфраструктуры протокол Диффи – Хеллмана остаётся столь же стойким, как задача вычисления дискретного логарифма.

2.2.3 Криптография на эллиптических кривых

Криптография на эллиптических кривых (ECC) основана на свойствах эллиптических кривых. Эллиптическая кривая обладает свойством, что любая прямая пересекает её не более чем в трёх точках. Это свойство используется для построения однонаправленной функции. Определяется операция $A \bullet A$ [11]. Если выполнить её n раз, получается значение Y. Вычислить Y просто, но определить число применений операции крайне сложно. Если $A \bullet A \bullet \ldots$ (выполнено p раз) даёт $A^p = O$, то отсюда следует $p = \mathrm{dlog}_A(O)$, что является сложной задачей в группе точек эллиптической кривой.

ECC имеет максимальное количество точек, определяемое порядком группы. Это приводит к эффекту «оборачивания» линий, аналогичному модульной арифметике. Максимум точек зависит от размера ключа.

ECC получила значительный интерес в последние годы, поскольку требуемые для таких систем длины ключей значительно меньше RSA. Например, эллиптическая кривая с ключом 256 бит обеспечивает ту же стойкость, что и RSA-ключ 3072 бит. ECC применяется в Bitcoin, Tor, WhatsApp [12]. Её стойкость основана на трудности решения дискретного логарифма в группе точек эллиптических кривых. NIST стандартизовал множество кривых, рекомендуе-

мых для использования в ECC. При атаке только на основе шифртеста единственный путь взлома ECC — найти алгоритм для нахождения дискретного логарифма. На данный момент решений с реалистичным временем работы не существует.

ECC уязвима к атакам по сторонним каналам. Эти атаки основаны на измерении физических параметров реализации: атаке по времени, анализу потребляемой мощности, дифференциальной атаке по мощности и анализу отказов. Атакующий может анализировать вариации, амплитуды и форму пиков напряжения [13].

Другим видом атак являются twist-security атаки [13]. Они успешны при выполнении ряда условий и могут привести к раскрытию закрытого ключа. В twist-атаке злоумышленник выбирает специальный вариант публичного ключа и обменивается им с жертвой для вычисления общего ключа, а затем использует его для восстановления закрытого ключа. Тем не менее при корректном выборе кривых и параметров подобных атак легко избежать. Отмечается, что некоторые стандартные кривые NIST могут содержать потенциальные бэкдоры.

3 Квантовые компьютеры

Квантовая механика известна своей парадоксальностью и тем, что она противоречит интуитивным представлениям. Это связано с такими явлениями, как суперпозиция, квантовая запутанность и квантовая неопределённость. Суперпозиция позволяет частице находиться сразу в нескольких состояниях. Запутанность описывает корреляции между частицами, которые невозможны в классическом мире. Принцип квантовой неопределённости утверждает, что наблюдение одной характеристики частицы приводит к потере информации о другой [14].

Базовой единицей информации в классических вычислениях является бит [15], который может принимать два дискретных значения — 0 и 1. В квантовой информатике элементарной единицей является кубит [16]. Кубит — это нормированный вектор в двумерном комплексном пространстве. Его состояние записывается в виде $\rangle\psi\rangle = \alpha|0\rangle + \beta|1\rangle$, где $|0\rangle$ и $|1\rangle$ — базисные векторы, а α и β — комплексные амплитуды, удовлетворяющие условию $|\alpha|^2 + |\beta|^2 = 1$. Для описания систем из двух и более кубитов используется тензорное произведение.

Набор квантовых вентилей, объединённых в схему, образует квантовый алгоритм. К распространённым квантовым вентилям относятся управляемый NOT (CNOT), преобразование Адамара, фазовые и инверсионные вентили.

В квантовых вычислениях измерение преобразует кубит в классический бит (0 или 1). Если состояние кубита равно $\rangle\psi\rangle = \alpha|0\rangle + \beta|1\rangle$, то вероятность получения 0 равна $|\alpha|^2$, а вероятность получить 1 — $|\beta|^2$. После измерения суперпозиция исчезает, и кубит переходит в соответствующее базисное состояние.

Теорема о запрете клонирования [17] утверждает, что невозможно создать точную копию произвольного квантового состояния. Ни одна квантовая схема не может принять на вход кубит и вывести исходный кубит вместе с его точной копией.

Квантовая запутанность — это явление, при котором система из двух или более кубитов описывается только совместным состоянием, а не отдельно для каждого кубита. Например, состояние $(|00\rangle + |11\rangle)/\sqrt{2}$ является состоянием Белла, и состояния отдельных кубитов не могут быть определены независимо. Измерение одного кубита сразу определяет состояние второго.

Существует множество физических платформ для реализации кубитов, включая сверхпроводниковые кубиты, фотонные кубиты (основанные на поляризации света) и другие. Фотонные кубиты удобны для дальних квантовых коммуникаций, тогда как сверхпроводниковые платформы лучше подходят для реализации квантовых процессоров благодаря сильным взаимодействиям между кубитами.

3.1 Распределение квантового ключа (протокол BB84)

Протокол BB84 представляет собой защищённый метод распределения общего ключа между двумя участниками. Для его работы требуется два канала: квантовый канал и классический канал. Классический канал является двунаправленным, тогда как квантовый может быть односторонним (только от отправителя к получателю, без обратной передачи).

Бит и базис, используемые для кодирования и отправки кубита, выбираются случайным образом. В протоколе используются два различных набора ортогональных базисов [18] (например, горизонтальная/вертикальная поляризация фотонов: $0°, 90°$, и диагональная поляризация: $45°, -45°$). Алиса и Боб заранее договариваются о том, каким образом кодируются биты 0 и 1 в каждом из двух базисов. Важно отметить, что измерение состояния фотона в одном базисе уничтожает информацию, соответствующую другому базису.

Алиса генерирует случайную последовательность битов и кодирует каждый бит в случайно выбранном базисе. Сформированные кубиты передаются Бобу через квантовый канал. Боб, не зная выбор Алисы, измеряет каждый полученный кубит с помощью случайно выбранного базиса (диагонального или вертикального/ горизонтального).

После передачи квантовой последовательности Алиса и Боб переходят к классическому каналу, по которому они объявляют, какие базисы были использованы для каждого бита (но не сами значения битов). Все позиции, где их базисы различаются, отбрасываются. Таким образом формируется финальный общий ключ, известный только Алисе и Бобу.

Защищённость протокола обеспечивается квантовым принципом невозможности клонирования: Ева не может создать копию кубитов. Если она попытается измерять кубиты, выбранный ею неправильный базис приведёт к уничтожению части информации. Когда Боб верифицирует несколько случайных битов с Алисой через классический канал, любое вмешательство будет обнаружено.

Если Ева попытается перехватить все кубиты и измерять их в случайных базисах, она неизбежно выберет неверные базисы для части из них, что приведёт к потере информации. Из-за этого она не сможет повторно подготовить правильные кубиты для Боба — атака будет выявлена.

3.2 Квантовые алгоритмы

3.2.1 Алгоритм Шора

Алгоритм Шора был разработан для решения двух фундаментальных задач, обеспечивающих стойкость современных криптосистем: задачи разложения числа на простые множители и задачи вычисления дискретного логарифма. Для классических вычислений обе задачи имеют решения субэкспоненциальной сложности. Используя квантовый алгоритм поиска периода, основанный на квантовом преобразовании Фурье (QFT), Шор предложил вероятностные алгоритмы, работающие за полиномиальное время. Алгоритм Шора способен разложить число на множители за полиномиальное время, тогда как лучшая классическая процедура — General Number Field Sieve — требует субэкспоненциального времени [19].

Оба алгоритма Шора (для факторизации и для дискретного логарифма) состоят из двух основных частей. Первая часть — классическая: она сводит исходную задачу к задаче поиска периода определённой функции. Вторая часть — квантовая — использует квантовый параллелизм и квантовое преобразование Фурье для нахождения периода.

Алгоритм Шора является вероятностным: он не всегда находит делитель числа с первой попытки. Однако при многократном запуске вероятность успеха стремительно возрастает, и факторизация становится практически гарантированной.

3.2.2 Алгоритм Гровера

Алгоритм Гровера — это квантовый алгоритм поиска, позволяющий выполнять поиск в неупорядоченной базе данных за $O(\sqrt{n})$ [2]. В классической модели вычислений такой поиск требует $O(n)$ операций. Алгоритм Гровера также является вероятностным: его можно запускать несколько раз, увеличивая вероятность нахождения нужного элемента.

Работа алгоритма основана на повторении итераций Гровера (amplitude amplification). Количество итераций можно увеличивать, чтобы повысить вероятность успешного обнаружения элемента.

Алгоритм Гровера может применяться не только для поиска в базе данных: он используется для нахождения среднего и медианы выборки, для вычисления обратных значений функций и других задач. Эти возможности делают его полезным инструментом для квантового криптоанализа. В частности, алгоритм Гровера может использоваться для ускоренного перебора ключей в симметричных криптосистемах, уменьшая эффективную стойкость алгоритмов, таких как AES.

3.2.3 Алгоритм Саймона

Алгоритм Саймона представляет собой квантовую схему, позволяющую определить, является ли функция взаимно-однозначной (1–1) либо двухкратно-однозначной (2–1). Свойство такой функции формулируется следующим образом: если два различных входных значения отображаются в один и тот же выход, то их XOR равен некоторому постоянному вектору b. Если b состоит из всех нулей, функция является 1–1; в противном случае она является 2–1.

Классический алгоритм для поиска такого совпадения требует $O(2^{n-1} + 1)$ вычислений. Квантовый алгоритм Саймона обеспечивает экспоненциальное ускорение по сравнению с классическим подходом [20].

Алгоритм Саймона стал источником идей для последующего создания алгоритма Шора. Сам по себе он также имеет криптографические приложения, при условии, что доступен корректный оракульный (query) интерфейс к функции.

3.3 Современные достижения в области квантовых компьютеров

Теоретические основы квантовых вычислений были заложены Ричардом Фейнманом в 1982 году. Бернстайн и Вазиряни [21] доказали, что квантовые компьютеры способны нарушать расширенную гипотезу Чёрча–Тьюринга. Они представили алгоритм рекурсивного преобразования Фурье, показавший возможности квантовых схем. Позднее, в 1994 году, Питер Шор [19] предложил алгоритмы, способные решать отдельные вычислительные задачи существенно быстрее классических подходов. Несмотря на эти результаты, квантовые вычисления долгое время оставались преимущественно теоретической областью.

Существует два основных типа квантовых компьютеров: универсальные квантовые компьютеры и квантовые отжигатели (quantum annealers). Универсальные квантовые компьютеры опираются на кубиты и квантовые логические вентили, позволяющие выполнять широкий спектр вычислений через квантовое программирование. Квантовые отжигатели не используют логические вентили и предназначены главным образом для решения задач оптимизации.

Компании, такие как D-Wave, разрабатывают квантовые отжигатели. Эти устройства более устойчивы к шуму и содержат порядка 4000 кубитов. D-Wave предлагает свои системы коммерчески компаниям, решающим оптимизационные задачи. Квантовые отжигатели используют свойства кубитов для поиска состояния с минимальной энергией, которое соответствует оптимальному решению. В отличие от классических компьютеров, перебирающих варианты последовательно, квантовые машины могут исследовать множество комбинаций одновременно. Ряд отраслей уже применяет системы D-Wave [22] для задач планирования и получает быстрые оптимальные решения. Существенным недостатком квантовых отжигателей является невозможность выполнять универсальные вычисления: задачу необходимо предварительно свести к формулировке оптимизации. В криптографии основное внимание уделяется именно универсальным квантовым компьютерам.

Исследования и разработка универсальных квантовых компьютеров также активно развиваются. Закон Мура [23] постепенно перестаёт действовать, так как размеры транзисторов стремятся к физическому минимуму, и квантовые эффекты начинают мешать их корректной работе. Квантовые вычисления рассматриваются как возможный путь дальнейшего роста вычислительных мощностей при физических ограничениях классических технологий.

Количество стартапов, занимающихся квантовыми вычислениями, стремительно растёт [23]. Компания PsiQuantum, поддерживаемая Microsoft, объявила о разработке оптического квантового процессора на один миллион кубитов, способного произвести революцию в вычислениях. Многие стартапы работают над созданием квантового аппаратного и программного обеспечения.

Крупнейшие технологические компании также активно инвестируют в квантовые технологии. Google и IBM соревнуются в достижении квантового превосходства. Они создают низкошумные квантовые процессоры. В 2019 году Google представила квантовый чип из 53 кубитов, продемонстрировавший вычисление, недоступное классическому суперкомпьютеру. IBM также активно развивает квантовое направление и уже создала несколько машин, планируя построить систему на 1000 кубитов к 2023 году [24]. Amazon и Microsoft сотрудничают со стартапами для интеграции квантовых вычислений в AWS и Azure.

Квантовые вычисления приведут к парадигмальному сдвигу в IT-индустрии. Исследователям предстоит решить проблемы коррекции ошибок и повышения стабильности квантовых устройств. Квалифицированных специалистов в области крайне мало, но в ближайшие десятилетия квантовые вычисления могут радикально преобразить промышленность благодаря новым вычислительным возможностям. Квантовые компьютеры создаются для решения задач, которые классически не поддаются решению за полиномиальное время. Первым практическим применением станет запуск алгоритма Шора, что сделает асимметричные криптосистемы уязвимыми. Поэтому необходимо разрабатывать новые криптографические задачи, устойчивые к квантовым атакам. Это привело к развитию постквантовой криптографии и запуску процесса стандартизации NIST в этой области.

4 Безопасность классической криптографии

Появление квантовых компьютеров поставило под сомнение безопасность классических криптографических алгоритмов. В симметричной криптографии эффективная стойкость (в битах) уменьшается вдвое при наличии квантового доступа. Асимметричная криптография с появлением квантовых вычислительных ресурсов фактически утратит свою безопасность.

4.1 Квантовая стойкость DES

Алгоритм DES достигает требуемой степени случайности за счёт множественных раундов преобразований. Однако сравнительно небольшой размер ключа (56 бит) делает DES уязвимым для полного перебора при современных вычислительных возможностях. Алгоритм 3DES увеличивает эффективную длину ключа до 168 бит, жертвуя при этом скоростью шифрования. Увеличение ключевого пространства до 2^{168} обеспечивает стойкость против классического перебора.

Тем не менее, как и для AES, алгоритм Гровера может быть использован для квантового поиска по ключу, что снижает необходимое количество операций до примерно 2^{84}, что уже не является достаточным уровнем безопасности. Кроме того, 3DES подвержен коллизиям, что делает его уязвимым к алгоритму Саймона. 3DES также чрезвычайно медленный и ресурсоёмкий, поэтому не рассматривается как вариант для широкого использования.

Модели упрощённого DES (SDES) [25] в настоящее время реализуются в квантовых схемах для анализа схем Фейстеля. Реализация S-блоков требует сложных компоновок квантовых вентилей, и их нельзя упростить, поскольку именно они обеспечивают нелинейность алгоритма. Сложность реализации 3DES в виде квантовых схем могла бы предположить некоторый уровень стойкости, однако из-за слабости базового DES он практически не рассматривается.

В целом симметричные криптографические алгоритмы считаются плохо приспособленными для реализации на квантовых моделях, но при этом они существенно более устойчивы к квантовому поиску, чем асимметричные криптосистемы.

4.2 Квантовая стойкость AES

AES зарекомендовал себя как один из наиболее надёжных криптографических алгоритмов, оставаясь устойчивым к атакам перебора, которые доступны современным вычислительным системам.

Однако недавний прогресс в области квантовых вычислений вновь поставил вопрос о стойкости AES перед лицом квантового перебора ключей. Хорошо известно, что асимметричные криптосхемы — такие как RSA, ECC и другие — полностью взламываются под воздействием квантового исчерпывающего поиска благодаря огромному параллелизму, обеспечиваемому суперпозицией кубитов [26].

Существуют различные алгоритмы, использующие принцип суперпозиции, такие как алгоритм Саймона, поиск Гровера и алгоритм Шора.

Алгоритм Гровера уменьшает сложность полного перебора ключей с $O(N)$ до $O(\sqrt{N/M})$, где значение M может быть сведено к 1 при выборе функций, имеющих единственное решение, что справедливо для реализации AES в квантовой схеме.

Ключевым недостатком криптоанализа симметричных систем на основе поиска Гровера является то, что сама криптосистема должна быть реализована в виде квантовой схемы. Существуют работы, посвящённые реализации AES в квантовой форме, однако атака на основе алгоритма Гровера требует наличия квантового оракула, способного выполнять AES. Таким образом, применимость атаки ограничена квантовыми моделями вычислений.

Кроме того, как отмечалось выше, квантовые алгоритмы позволяют сократить число испытаний только до $N^{1/2}$, где N — размер ключевого пространства. Следовательно, стойкость можно сохранить простым увеличением длины ключа. Переход от AES-128 к AES-256 делает алгоритм практически непригодным для взлома методом квантового перебора.

4.3 Квантовая стойкость RSA

Квантовый алгоритм факторизации Шора представляет собой квантовую схему, способную разлагать большие числа на простые множители за полиномиальное время [27]. Это является огромным ускорением по сравнению с классическими методами, для которых задача факторизации требует субэкспоненциального времени. По мере развития квантовых вычислений задача разложения

на множители перестанет быть трудной, что сделает взлом RSA тривиальным. Первая практическая оценка ресурсов для факторизации RSA требовала более миллиарда кубитов. Однако к 2019 году этот показатель был снижен до примерно 20 миллионов кубитов [28]. В частности, показано, что 2048-битное число RSA может быть разложено менее чем за 8 часов, используя около 20 миллионов шумных кубитов. Современные квантовые компьютеры обладают лишь 50–100 кубитами, но в течение следующих 25 лет программные и аппаратные улучшения могут сделать факторизацию крупных чисел практической.

Алгоритм Шора использует квантовое преобразование Фурье (QFT), чтобы воспользоваться квантовым параллелизмом. Задача разложения числа на множители сводится к задаче поиска периода. В работе авторов реализовано множество оптимизаций, уменьшающих требуемое число кубитов. Вариант алгоритма, предложенный Экерой и Хастадом [28], сокращает число необходимых операций умножения. Дополнительно используются оптимизации, такие как оконная арифметика и «oblivious carry runways».

Основная идея алгоритма факторизации Шора описывается следующим образом [29]. Пусть $N = pq$, где p и q — простые числа. Чтобы разложить N, выбирается случайное число $a < N$ такое, что $\gcd(a, N) = 1$. Рассматривается функция $f(x) = a^x \bmod N$, и определяется её период r (на квантовой схеме). Если период r нечётный, алгоритм повторяется с новым значением a. Если r — чётный, то выполняется $f(x) = f(x + r)$, и можно вычислить делители N как $\gcd(a^{r/2\pm1}, N)$. Нахождение периода осуществляется с помощью квантового преобразования Фурье.

Первая часть алгоритма заключается в инициализации входного и выходного регистров: входной регистр – суперпозиция всех значений от 0 до $N - 1$, выходной – нулевой. Эти регистры затем запутываются посредством квантовой схемы, реализующей функцию $f(x) = a^x \bmod N$.

После вычисления суперпозиции значений функции $f(x)$ над всеми x к входному регистру применяется квантовое преобразование Фурье. Затем выполняется измерение входного регистра, которое даёт значение y. Поскольку регистры запутаны, состояние выходного регистра коллапсирует в значение $f(x_0)$, соответствующее одному из x.

Отношение Y/N приводится к несократимой дроби, и знаменатель этой дроби рассматривается как возможный кандидат на период. Если он действительно является периодом функции, задача решена; если нет – проверяются кратные значения. В случае неудачи регистры повторно инициализируются, и процедура выполняется заново (результат измерения случаен и может приводить к отличным значениям y).

Алгоритм Шора подрывает фундаментальную идею, лежащую в основе RSA. Следовательно, в постквантовую эпоху требуется замена RSA, и в настоящее время NIST проводит поиск и стандартизацию новых криптографических алгоритмов, устойчивых к квантовым атакам.

4.4 Квантовая безопасность протокола Диффи–Хеллмана

Протокол обмена ключами Диффи–Хеллмана основан на задаче дискретного логарифмирования. Идею алгоритма Шора можно использовать для получения экспоненциального ускорения при решении этой задачи.

Задача дискретного логарифмирования может быть преобразована в задачу нахождения периода двумерной функции. Эта задача решается с помощью квантового преобразования Фурье, которое использует квантовый параллелизм для ускорения вычислений.

Пусть p — большое простое число, а q — порождающий элемент группы $\mathbb{Z}_p$. Пусть также $Y = q^k$. Чтобы взломать протокол, необходимо определить значение k. Определим двумерную функцию: $f(x_1, x_2) = q^{x_1} y^{x_2}$. Период этой функции находится из условия: $f(x_1 + w_1,\, x_2 + w_2) = f(x_1, x_2)$. После нахождения периода можно вычислить: $k = -(w_1/w_2) \bmod p$ [30].

Этот алгоритм является полиномиальным решением для взлома протокола Диффи–Хеллмана. Так как безопасность DH основана на задаче дискретного логарифмирования, а эта задача станет разрешимой с появлением мощных квантовых компьютеров, протокол перестанет быть безопасным.

4.5 Квантовая безопасность на эллиптических кривых

Криптография на эллиптических кривых (ECC) основана на задаче дискретного логарифмирования в группах точек эллиптических кривых. В своей фундаментальной работе Шор опубликовал два квантовых алгоритма: один — для факторизации больших чисел, а другой — для решения задачи дискретного логарифмирования в произвольной группе.

Все возможные атаки на ECC сводятся к атаке на задачу дискретного логарифмирования. На классических компьютерах известны только экспоненциальные алгоритмы её решения. Алгоритм Шора является полиномиальным и использует квантовый параллелизм и квантовое преобразование Фурье для нахождения периода функции. Это позволяет эффективно решать дискретное логарифмирование и, следовательно, взламывать ECC за полиномиальное время.

Так как ECC требует значительно меньших размеров ключей по сравнению с RSA, взлом эллиптических кривых станет возможен раньше. Количество кубитов, необходимых для взлома ECC, существенно меньше, чем количество кубитов, требуемых для взлома RSA. Поэтому в условиях появления полноценного квантового компьютера ECC станет уязвимой раньше RSA.

5 Постквантовая криптография

Постквантовая криптография — это область криптографии, в которой разрабатываются криптографические алгоритмы, устойчивые к противнику, обладающему квантовым компьютером. Все современные асимметричные алгоритмы (RSA, ECC, DH, DSA) становятся уязвимыми в квантовой модели вычислений. Они основаны на задачах факторизации и дискретного логарифмирования, которые эффективно решаются на квантовых компьютерах с помощью алгоритма Шора. Ранее математики и криптографы полагались на трудность этих задач числовой теории, однако теперь возникает необходимость искать новые математические проблемы, которые не поддаются квантовым атакам.

Симметричные алгоритмы и хеш-функции считаются сравнительно безопасными в постквантовом мире. Алгоритм Гровера ускоряет атакующую переборную операцию лишь до квадратного корня от сложности $\mathcal{O}(\sqrt{N})$ [31]. Однако большинство симметричных систем можно сделать вновь стойкими просто удвоением длины ключа.

Все существующие асимметричные алгоритмы основаны на математических задачах, которые исследуются сотни лет. Их слабое место состоит в том, что квантовые компьютеры эффективно выполняют параллельные вычисления, если необходимо получить единственный итоговый результат. Благодаря суперпозиции кубитов квантовая машина может параллелизовать вычисления над всей областью поиска и затем выполнить измерение.

Чтобы исключить возможность использования квантового параллелизма, перспективные постквантовые алгоритмы должны основываться на задачах, в которых требуется не один, а множество результирующих значений, либо задачах, чья структура плохо поддаётся квантовым преобразованиям.

В настоящее время большинство постквантовых алгоритмов относится к шести основным семействам. Каждое семейство использует собственный класс математических задач, которые остаются трудными даже для квантовых компьютеров. Эти задачи лягут в основу следующего поколения асимметричной криптографии.

NIST начал процесс стандартизации постквантовой криптографии в 2016 году. Цель процесса — выбор новых схем цифровых подписей и алгоритмов шифрования/обмена ключами, устойчивых к атакам квантовых компьютеров.

5.1 Криптография на решетках

Криптография на решётках основана на трудных задачах теории решёток. В таких схемах безопасность опирается на семейство задач на решётках, причём ключевой особенностью является то, что стойкость основана на задачах в худшем случае. Большинство других криптосистем опираются на сложность в среднем случае [32].

Решётка представляет собой регулярно расположенную бесконечную сетку точек. Вектор можно рассматривать как точку решётки, то есть набор координат. Начало координат соответствует вектору, все координаты которого равны нулю. Вектор называют *коротким*, если он расположен близко к началу координат, и *длинным*, если он находится далеко от него.

Базисом решётки называется небольшой набор векторов, линейные комбинации которых порождают всю решётку. Для n-мерной решётки выбирают n векторов так, чтобы никакая другая точка решётки не лежала на прямой между произвольным базисным вектором и началом координат.

Для одной и той же решётки может существовать множество базисов. Базис называется *коротким*, если он состоит из коротких векторов, и *длинным*, если векторы базиса длинные. На основе решёток сформулированы несколько фундаментальных трудных задач:

- **Задача кратчайшего вектора (SVP)**: дан длинный базис решётки L. Требуется найти точку решётки L, расположенную к началу координат как можно ближе.
- **Задача короткого базиса (SBP)**: дан длинный базис решётки L. Необходимо найти короткий базис той же решётки.
- **Задача ближайшего вектора (CVP)**: дан длинный базис решётки L и точка P в пространстве. Требуется найти точку решётки, наиболее близкую к P.
- **Задача короткого целочисленного решения (SIS)**: даны m векторов размерности n, $v_i \in \mathbb{Z}_q^n$. Требуется найти вектор коэффициентов $y \in \mathbb{Z}_q^m$ из малых значений (например, $y_i \in \{0, 1\}$), такой что линейная комбинация $\sum_i y_i v_i = 0$.

Во «в малых размерностях» эти задачи выглядят элементарными. Однако в криптографии решётки имеют очень большие размерности, и при заданном длинном базисе указанные задачи становятся крайне сложными. Если бы противнику был известен короткий базис, они были бы легко решаемы, но использование только длинного базиса делает их хорошими кандидатами для построения криптосхем.

Изучение решёток начинается ещё в XIX веке, и накопленные результаты дают хорошее понимание того, что именно можно и нельзя эффективно делать с такими структурами. Это укрепляет уверенность в надёжности решёточных задач как основы асимметричных алгоритмов. Основной класс атак использует алгоритмы редукции решёток, например LLL: это полиномиальный алгоритм, позволяющий по заданному длинному базису получить относительно короткий (но не оптимальный) базис.

В 1997 году Аджтай предложил криптосистему, основанную на задаче кратчайшего вектора [32]. Уже в 1998 году Нгуен и Штерн показали криптоанализ криптосистемы Аджтая–Дворка [33]. Алгоритм Голдрейха–Голдвассера–Халеви [34], основанный на задаче ближайшего вектора (CVP), был впоследствии взломан Нгуеном [35].

Схема NTRU была предложена в 1996 году и затем формально описана в [36]. За прошедшие годы она неоднократно модифицировалась и усиливалась, и современный вариант NTRU рассматривается как один из финальных кандидатов в процессе стандартизации NIST.

5.1.1 Схема шифрования GGH

В схеме шифрования GGH приватный ключ получателя представляет собой *короткий базис* решётки, а публичный ключ — *длинный базис*. Число измерений решётки выбирается равным длине сообщения в битах. Отправитель использует «плохой» (длинный) базис, чтобы найти точку в пространстве решётки, расположенную близко к некоторой точке решётки, которая и кодирует сообщение. Обладая коротким базисом, получатель может эффективно «спроецировать» полученную точку на ближайшую решёточную точку и тем самым восстановить исходное сообщение.

Для противника задача расшифрования трудна, поскольку в его распоряжении только длинный базис. Схема изначально рассматривалась как потенциально стойкая к квантовым атакам, однако была взломана на классических компьютерах из-за определённых уязвимостей: при накоплении большого числа зашифрованных сообщений становилось возможно восстановить часть информации о закрытых текстах.

Тем не менее идеи, заложенные в GGH, с учётом уменьшения уязвимостей и модификаций конструкций, нашли отражение во многих современных кандидатах на постквантовые криптосистемы.

5.1.2 Обучение с ошибками

Learning With Errors (обучение с ошибками, LWE) — это подмножество решёточной криптографии. Оно основано на новой функции с потайным ходом (trapdoor function), которую легко вычислить, но трудно инвертировать.

В исходной задаче система уравнений имеет вид $AX = B$, и если заданы A и B, то найти X несложно — достаточно применить метод Гаусса [37]. Однако если к произведению AX добавить случайную ошибку e, так что получается $AX + e = B$ то, обладая только A и B, найти X становится крайне трудной математической задачей. Эта трудность и используется как криптографическая опора.

Существуют модификации LWE: обучение с ошибками в кольце Ring-LWE и обучение с ошибками по модулю Module-LWE, которые уменьшают размеры ключей, сохраняя стойкость схемы.

Система уравнений $AX = B$ может иметь больше уравнений, чем переменных, но она конструируется так, чтобы быть разрешимой. Приватный ключ — это вектор переменных X, а публичный ключ — матрицы A и B.

Для шифрования сообщения отправитель выбирает случайное подмножество строк матрицы, складывает их и затем: добавляет большую ошибку, чтобы закодировать бит 1 и добавляет маленькую ошибку, чтобы закодировать бит 0. Получатель, зная приватный ключ, легко определяет, была добавлена большая или маленькая ошибка, и таким образом восстанавливает сообщение. Для злоумышленника же задача определения X по A и B остаётся вычислительно сложной.

Сегодня 27 из 69 алгоритмов, поданных на стандартизацию NIST, основаны на LWE-подобных задачах. Компания Google уже внедрила LWE в Chrome для обеспечения постквантовой безопасности.

5.2 Цифровые подписи на основе хеширования

Цифровые подписи на основе хеширования представляют собой альтернативу современным схемам цифровой подписи, использующим асимметричные алгоритмы вроде RSA. Их криптографическая стойкость опирается на два ключевых свойства хеш-функций: стойкость к коллизиям и стойкость к нахождению прообраза.

Стойкость к нахождению прообраза означает, что по заданному выходному значению y затруднительно найти вход x, такой что $H(x) = y$. Слабая стойкость к коллизиям означает, что по заданному сообщению m_1 трудно найти другое сообщение m_2 с тем же значением хеша: $H(m_1) = H(m_2)$. Сильная стойкость к коллизиям означает сложность поиска двух сообщений m_1 и m_2, удовлетворяющих $H(m_1) = H(m_2)$. Замечание: атака на сильную стойкость упрощается благодаря парадоксу дней рождения. Если слабая стойкость к коллизиям и стойкость к прообразу требуют порядка $O(2^{n-1})$ операций (где n — длина выходного значения хеша), то сильная стойкость требует лишь $O(2^{n/2})$.

При условии использования хорошей криптографической хеш-функции задача нахождения коллизий или прообразов остаётся крайне сложной. Создание квантовых алгоритмов, способных эффективно решать такие задачи, вероятно, также будет крайне сложным или невозможным. Поэтому цифровые подписи на основе хеширования рассматриваются как надёжный инструмент аутентификации в постквантовом мире. Однако у таких схем есть важный недостаток: каждая одноразовая подпись может быть использована только один раз.

Из 69 схем, представленных на конкурс NIST, две относятся к криптографии на основе хеширования. Лэмпорт предложил первую такую схему цифровой подписи в 1979 году [38]. Винтерниц разработал одноразовую схему подписи, значительно более эффективную, чем схема Лэмпорта: она использует более короткие ключи и подписи. Меркл предложил схему, объединяющую идею Винтерница с бинарными хеш-деревьями, которая получила название "Подпись Меркла". Схема SPHINCS+, являющаяся одним из кандидатов NIST, использует комбинацию одноразовой подписи Винтерница (WOTS+) и хеш-деревьев Меркла в конструкции Forest of Random Subsets.

5.2.1 Схема цифровой подписи Лэмпорта

Схема цифровой подписи Лэмпорта представляет собой одноразовую схему подписи. Для её работы требуется криптографически стойкая хеш-функция. Если уровень безопасности равен b, то необходима хеш-функция с длиной выхода $2b$. Например, при необходимости обеспечить 128-битную безопасность можно использовать любую хеш-функцию с 256-битным выходом.

Закрытый ключ генерируется с помощью генератора случайных чисел: создаётся 256 пар случайных чисел, каждое длиной 256 бит. Эти значения и являются закрытым ключом. Размер закрытого ключа равен $8b^2$.

Открытый ключ состоит из 512 хешей всех случайных чисел, сгенерированных для закрытого ключа. Размер открытого ключа также составляет $8b^2$. Эти значения публикуются пользователем, который собирается подписывать документы. Алгоритм подписи [3] требует вычисления хеша сообщения. Поскольку хеш имеет длину 256 бит, для каждого бита (в зависимости от его значения 0 или 1) выбирается одно число из соответствующей пары закрытого ключа. Таким образом, цифровая подпись представляет собой последовательность из 256 чисел, которая публикуется вместе с сообщением.

Проверяющий вычисляет хеш полученного сообщения. Затем для каждого бита хеша выбирает соответствующий хеш из открытого ключа. После этого проверяющий хеширует каждое число из подписи и убеждается, что полученные значения совпадают с выбранными элементами открытого ключа. Так как раскрытие отдельных элементов закрытого ключа происходит во время подписи, данная схема является одноразовой: после одного использования пара ключей (закрытый и открытый) должна быть уничтожена.

5.3 Криптография на основе кодов, исправляющих ошибки

Криптография на основе кодов опирается на коды, исправляющих ошибки [3]. Учёные изучают их уже более 40 лет. Коды, исправляющие ошибки, широко используются в системах связи для исправления ошибок передачи. Чтобы отправить сообщение, текст кодируется с помощью кода, исправляющего ошибки, закодированную информацию отправляют по каналу связи. Если полученное сообщение содержит несколько ошибок, эти ошибки исправляются , после чего исправленная информация декодируется в отправленное сообщение.

Одним из наиболее известных криптосистем данного типа является алгоритм МакЭлиса [39]. Он использует линейные коды исправления ошибок. Получатель обладает хорошим кодом исправления ошибок, который является закрытым ключом. Этот код умножается на две маскирующие матрицы, что приводит к получению «плохого» кода исправления ошибок — открытого ключа. Открытый ключ публикуется. Отправитель пропускает открытый текст через плохой код исправления ошибок, а затем добавляет некоторое число ошибок согласно параметрам избыточности. Полученный результат является итоговым шифртекстом. Получатель использует свой хороший код коррекции ошибок для расшифровки.

Алгоритм МакЭлиса был предложен в 1978 году [40], и до настоящего времени никаких уязвимостей в нём обнаружено не было. Основная причина отсутствия широкого практического применения — очень большой размер открытого ключа, который значительно превышает размеры открытых ключей в асимметричных схемах вроде RSA.

В соревновании NIST из 69 представленных алгоритмов 21 относится к криптосистемам на основе кодов, исправляющих ошибки.

5.4 Некоммутативная криптография

Некоммутативная криптография опирается на некоммутативные группы (где $A + B \neq B + A$). Простым примером служит кубик Рубика: последовательность ходов представляет собой набор действий, применяемых к кубику [41]. Операция сложения соответствует конкатенации последовательностей ходов. Такое сложение является некоммутативным.

Обратный ход — это действие, противоположное предыдущему ходу, что позволяет устранять пары действий и записывать последовательность без них. Отрицание последовательности ходов состоит из всех противоположных ходов, записанных в обратном порядке. Таким образом, вся последовательность может быть обращена. Отрицание последовательности A обозначается как $-A$.

Задача сопряжённости: даны две последовательности ходов A и B. Требуется найти X такое, что $X + A - X = B$.

Поскольку операция сложения некоммутативна, задача является очень трудной. Однонаправленная функция легко вычисляется, но её обращение крайне затруднено.

5.4.1 Протокол обмена ключами Стикеля

Поскольку протокол Диффи–Хеллмана не является квантово-устойчивым, в качестве альтернативы используется протокол обмена ключами Стикеля. Он считается квантово-устойчивым и основан на некоммутативной криптографии [42].

В протоколе Стикеля [43] задаются две последовательности ходов A и B. У Алисы и Боба также есть два натуральных числа, которые служат их закрытыми ключами. Пусть у Алисы это n и m, а у Боба — r и s. Алиса генерирует открытый ключ $PK_a = mA + nB$ и отправляет его Бобу. Боб генерирует открытый ключ $PK_b = rA + sB$ и отправляет его Алисе. Затем Алиса вычисляет $K_a = mA + PK_b + nB = (m + r)A + (s + n)B$, а Боб вычисляет $K_b = rA + PK_a + sB = (r + m)A + (n + s)B$.

Полученные ключи совпадают, и противник не может их вычислить из-за некоммутативности группы: для перехватчика решение задачи сопряжённости остаётся вычислительно трудным.

В конкурсе NIST только один алгоритм из 69 был некоммутативным криптосистемным подходом. Он был взломан, поэтому ни один некоммутативный криптосистемный алгоритм не будет стандартизирован.

5.5 Многомерная криптография

Многомерная криптография основана на трудной математической задаче решения системы уравнений для многочленов от нескольких переменных. Все многомерные криптосистемы опираются на многомерное квадратичное отображение [44]. Квадратичное отображение берёт последовательность $x = (x_1, \ldots, x_n) \in \mathbb{F}_q^n$ и возвращает вектор $y = (p_1(x), \ldots, p_m(x)) \in \mathbb{F}_q^m$, где $p_i(x)$ — квадратичные многочлены от нескольких переменных для $i = 1, \ldots, m$, а коэффициенты многочленов принадлежат $\mathbb{F}_q$. Такое отображение называется многомерным квадратичным отображением P с m компонентами и n переменными.

Задача MQ: дано квадратичное отображение $P : \mathbb{F}_q^n \to \mathbb{F}_q^m$ и целевой вектор $t \in \mathbb{F}_q^m$. Требуется найти s такое, что $P(s) = t$. Значение s не является уникальным, так как отображение P не является инъективным.

Эта задача считается трудной даже для квантовых компьютеров. Существуют методы, такие как базисы Грёбнера, позволяющие решать данную задачу. В последние годы для решения MQ-задачи используются усовершенствованные алгоритмы, похожие на базисы Грёбнера: алгоритмы F4/F5 и XL [45].

Основная область применения MQ — цифровые подписи. Наиболее известной схемой цифровой подписи в многомерной криптографии является схема «Масло и Уксус» (Oil and Vinegar). Rainbow — схема цифровой подписи, созданная на основе несбалансированной версии Oil and Vinegar, и является финалистом конкурса NIST.

5.5.1 Схема «Масло и уксус»

Схема цифровой подписи основана на сложной математической MQ-задаче. Она использует два типа переменных — «масло» и «уксус» (oil и vinegar). Выбирается случайное квадратичное отображение $P : \mathbb{F}_q^n \to \mathbb{F}_q^m$. Число переменных равно n, а число квадратичных многочленов — m. В отображении выделяются m переменных типа «масло» и $n - m$ переменных типа «уксус» [46].

Ключевая идея схемы заключается в том, что квадратичные члены многочленов могут содержать только комбинации вида «уксус–уксус» и «масло–уксус». Комбинаций вида «масло–масло» нет. После построения многочлены маскируются обратимым линейным преобразованием, чтобы противник не мог различить типы переменных.

Алиса, желая подписать документ, публикует композицию $R = P \circ T$, где T — обратимое линейное преобразование, а сами P и T хранятся как закрытый ключ. Для подписи документа d она вычисляет хэш $y = H(d)$, затем вычисляет $s' = T^{-1}(y)$, после чего находит s такое, что $s = P^{-1}(s')$ — это и есть подпись. Получатель использует открытый ключ R, вычисляет $R(s)$ и сравнивает результат с хэшем сообщения.

Для Алисы нахождение s просто, поскольку она знает, какие переменные являются «маслом», а какие — «уксусом». Она выбирает случайные значения уксусных переменных и подставляет их. Поскольку квадратичные члены зависят только от уксуса, получается линейная система уравнений по масляным переменным. Число масляных переменных равно m, и уравнений также m, поэтому система решается методом Гаусса. Для противника же, не знающего разделения переменных, обратная задача сводится к NP-трудной MQ-проблеме.

Изначальная схема «Масло и уксус» [47] имела одинаковое число масляных и уксусных переменных и была разобрана Кипнисом и Шамиром [48]. Впоследствии была предложена новая схема — Unbalanced Oil and Vinegar (UOV) [46], в которой числа переменных разных типов отличаются. Подбор корректных параметров n и m позволяет повысить устойчивость схемы.

Схема цифровой подписи Rainbow представляет собой многослойную систему, построенную на нескольких уровнях UOV [49]. Это усложняет прямые атаки на MQ-задачу, но одновременно увеличивает уязвимость к атаке MinRank.

В задаче **MinRank** даны k матриц M_i размером $n \times m$ и целевой ранг r. Необходимо найти коэффициенты y, такие что их линейная комбинация имеет ранг не выше r. Увеличение числа слоёв Rainbow повышает уязвимость к MinRank, поэтому в стандартизируемой версии используется оптимизация из двух слоёв. Недавно Бёлленс [50] предложил новую пересекающую атаку, уменьшающую предполагаемый уровень безопасности Rainbow на несколько бит.

5.6 Криптосистемы на основе изогений

Криптосистемы на основе изогений используют свойства эллиптических кривых. Эллиптическая кривая определяется как множество решений полиномиального уравнения от двух переменных. Обычно кривые задаются уравнением вида $y^2 = x^3 + ax + b$, где выполняется условие $4a^3 + 27b^2 \neq 0$. Это условие гарантирует отсутствие особых точек.

Эллиптические кривые обладают следующим свойством: если соединить две точки кривой прямой, то эта прямая пересечёт кривую в третьей точке. Отражение этой точки относительно оси X даёт результат операции сложения на эллиптической кривой. Удвоение точки осуществляется аналогично, только вместо прямой используется касательная в этой точке. Скалярное умножение $[n]P$ означает сложение точки P самой с собой n раз.

Алгоритм Диффи—Хеллмана на эллиптических кривых (ECDH) стал первым алгоритмом обмена ключами на эллиптических кривых [51]. В нём фиксируется эллиптическая кривая над полем по модулю простого числа p и точка P со специальными свойствами. Алиса и Боб выбирают случайные числа n_a и n_b как свои закрытые ключи, вычисляют $P_a = [n_a]P$ и $P_b = [n_b]P$ и обмениваются ими. Далее Алиса вычисляет $[n_a]P_b = [n_a n_b]P$, а Боб — $[n_b]P_a = [n_a n_b]P$. Полученная точка служит общим секретом.

Безопасность ECDH основана на сложности задачи дискретного логарифмирования на эллиптических кривых. Однако эта задача решается квантовым алгоритмом Шора за полиномиальное время, поэтому ECDH считается небезопасным при наличии квантовых компьютеров.

После этого началось интенсивное исследование эллиптических кривых и их морфизмов, что привело к появлению криптосистем на основе изогений.

5.6.1 Схема Диффи–Хеллмана на суперсингулярных изогениях (SIDH)

Изогения — это гомоморфное рациональное отображение между двумя эллиптическими кривыми $\Phi : E_0 \to E_1$. В SIDH Алиса и Боб генерируют приватные изогении, которые и являются их закрытыми ключами. Они применяют свои изогении к общей эллиптической кривой E, после чего обмениваются полученными результатами. Получив новую кривую, другая сторона применяет свою приватную изогению, чтобы получить ещё одну кривую. В отличие от ECDH, конечные кривые у Алисы и Боба не обязаны быть одной и той же кривой, однако они имеют одинаковую структуру: кривые являются изоморфными. Для эллиптической кривой можно вычислить величину j-инварианта, которая одинакова для изоморфных кривых. Поэтому j-инвариант вычисляется Аликой и Бобом независимо и служит общим секретом. Торсионная подгруппа $E[n]$ эллиптической кривой — это множество всех точек $P \in E$, удовлетворяющих $[n]P = 0$, где 0 — аддитивная единица на E.

SIDH был предложен в 2011 году [52]. В схеме фиксируется суперингулярная эллиптическая кривая E, определённая над полем $\mathbb{F}_q$, где $q = p^2$ и $p = 2^a 3^b - 1$. Алиса и Боб совместно используют кривую E, а также базисы P_a, Q_a для $E[2^a]$ и P_b, Q_b для $E[3^b]$ (базис — это набор точек, линейные комбинации которых порождают всю группу точек).

Алиса и Боб выбирают случайные числа r_A и r_B из диапазонов $0 \leq r_A < 2^{a-1}$ и $0 \leq r_B < 3^{b-1}$ соответственно. Алиса вычисляет свою изогению Φ_A, ядро которой задаётся точкой $R_A = P_A + [r_A]Q_A$. Затем она публикует $E_A = \Phi_A(E)$, а также $\Phi_A(P_b)$ и $\Phi_A(Q_b)$ как свой открытый ключ. Эти данные позволяют Бобу вычислить свою вторую изогению Ψ_B. Боб строит Ψ_B с ядром, порождённым точкой $\Phi_A(P_b) + [r_B]\Phi_A(Q_b)$, и затем вычисляет j-инвариант кривой $\Psi_B(E_A)$. Алиса выполняет аналогичную процедуру, в результате чего обе стороны получают один и тот же общий секрет.

Безопасность схемы основана на сложности задачи l^e-изогении. Эта задача сформулирована так: даны две эллиптические кривые E_1 и E_2, между которыми существует изогения Φ с ядром размера l^e; необходимо определить ядро Φ. В SIDH это задачи 2^a-изогении и 3^b-изогении. Математики исследуют эти задачи уже более 20 лет, и для больших степеней e и любого натурального l не найдено эффективных атак.

Механизм инкапсуляции ключей SIKE (Supersingular Isogeny-Based Key Encapsulation) основан на SIDH и входил в список альтернативных кандидатов третьего раунда NIST на стандартизацию схем инкапсуляции ключей.

6 Стандартизационный процесс постквантовой криптографии NIST

Как было показано в предыдущих разделах, современные алгоритмы с открытым ключом и схемы цифровой подписи не обеспечивают безопасность при наличии достаточно мощного квантового компьютера. Последние достижения в области квантовых вычислений привели к созданию 53-кубитного квантового компьютера компанией Google. Хотя такая машина ещё совершенно не подходит для взлома криптографии с открытым ключом, сам факт её существования изменил вопрос с «если» на «когда» такие схемы будут сломаны. Google и IBM ведут гонку по созданию устойчивых к шуму квантовых компьютеров с большим числом кубитов.

В результате разрабатываются новые алгоритмы, которые не опираются на задачу факторизации или другие задачи, уязвимые к параллельным квантовым вычислениям. Большинство таких алгоритмов основано на решётках и на кодах с исправлением ошибок. Наиболее перспективными считаются алгоритмы, использующие труднорешаемые задачи на решётках, поскольку они гарантируют безопасность как в худшем, так и в среднем случае.

Учитывая возрастающую необходимость постквантовой защиты, NIST объявил мировой конкурс задач и алгоритмов, которые могут заменить существующие схемы шифрования с открытым ключом и схемы цифровой подписи.

На конкурс NIST по постквантовой криптографии было подано 69 алгоритмов. Лучшие из них предстоит выбрать к 2023 году. Эти алгоритмы будут стандартизованы и будут использоваться в течение многих последующих лет. Из 69 алгоритмов 20 представляют собой схемы цифровой подписи и 49 — схемы шифрования с открытым ключом. После первого раунда были отобраны 26 алгоритмов.

Алгоритмы первого раунда оценивались по трём основным критериям: безопасность, стоимость и производительность, а также характеристики алгоритма и его реализации [53].

Безопасность являлась ключевым фактором. Алгоритмы должны были обеспечивать семантическую стойкость к адаптивным атакам с выбранным шифртестом. NIST также учитывал алгоритмы, обладающие семантической стойкостью к атакам с выбранным открытым текстом. Схемы цифровой подпи-

си должны были обеспечивать невозможность подделки подписей при адаптивной атаке с выбранным сообщением. NIST определил пять категорий безопасности и представил перечень дополнительных требований (desiderata), включая устойчивость к побочным каналам и атакам многократного ключа.

Стоимость и производительность были вторым по значимости аспектом. Стоимость включает вычислительную эффективность (скорость работы алгоритма) и требования к памяти (размер кода и объём ОЗУ).

Характеристики алгоритма и реализации включали простоту и элегантность конструкции, а также гибкость (работу на разных платформах и возможность параллелизации).

После второго раунда NIST отобрал 15 алгоритмов. Эти алгоритмы можно классифицировать по типу математически сложной задачи, лежащей в основе их безопасности.

6.1 Кандидаты на основе решёток

6.1.1 CRYSTALS-KYBER (финалист)

CRYSTALS-KYBER является финалистом в процессе стандартизации NIST PQC. Он относится к семейству примитивов Cryptographic Suite for Algebraic Lattices (CRYSTAL). Та же криптографическая платформа используется для создания алгоритма цифровой подписи DILITHIUM. Kyber представляет собой механизм инкапсуляции ключей (KEM), соответствующий IND-CCA2 безопасности. Алгоритм основан на сложности задачи *module learning-with-errors* (MLWE) [54].

KYBER предлагается как алгоритм с открытым ключом, обеспечивающий уровни безопасности, сопоставимые со схемами AES. NIST рассматривает AES-128, AES-192 и AES-256 как эталоны безопасности для уровней 1, 3 и 5 соответственно. Варианты Kyber (kyber-512, kyber-768, kyber-1024) обеспечивают сопоставимые уровни безопасности (примерно в пределах 2^{30} от AES-256). Алгоритм считается одним из наиболее конкурентоспособных благодаря своей высокой производительности по сравнению с другими предлагаемыми криптосхемами. Повышение уровня безопасности достигается изменением порядка блочных матриц, используемых в алгоритме.

Из-за высокой скорости работы решёточных систем и меньших размеров открытых ключей такие системы представляют собой привлекательный вариант для внедрения [55]. KYBER демонстрирует это, используя ключи от 800 байт до 1,5 килобайт, в то время как современным схемам, например RSA, требуется ключ размером около 2 килобайт.

Так как Kyber основан на MLWE, необходимо защищаться от многих алгебраических атак, применимых к LWE. В ранних версиях возникали вопросы о влиянии сжатия открытого ключа на криптоанализ шифртеста, но негативных эффектов обнаружено не было. В дальнейших версиях сжатие было удалено, а параметры изменены в сторону симметричных примитивов, основанных на SHAKE256, вместо ранее использовавшегося SHA3-256. CRYSTALS-KYBER также разделяет программный каркас с CRYSTALS-DILITHIUM (схемой цифровой подписи), что делает эту пару более удобной для совместной реализации.

CRYSTALS-KYBER, наряду с другими постквантовыми алгоритмами, внедрён Cloudflare в библиотеку Reusable Cryptographic Library. Amazon добавил гибридный режим KYBER в AWS Key Management Service. IBM использует комплект KRYSTAL (KYBER и DILITHIUM) в своём, по их утверждению, первом в мире ленточном накопителе, защищённом от квантовых атак.

6.1.2 SABER (финалист)

SABER — это ещё один решёточный KEM-алгоритм, являющийся финалистом процесса стандартизации NIST PQC. Он представлен в трёх вариантах: LightSABER, SABER и FireSABER, обеспечивающих уровни безопасности 1, 3 и 5 соответственно, согласно требованиям NIST. Уровни безопасности LightSABER, SABER и FireSABER соответствуют AES-128, AES-192 и AES-256. Алгоритм основан на задаче *Module Learning with Rounding* (MLWR), которая отличается от MLWE тем, что ошибки вносятся с помощью округления значений. Как и в CRYSTALS-KYBER, переход между версиями осуществляется изменением размерностей блочных матриц, используемых в алгоритме. В отличие от KYBER, SABER применяет умножение без использования NTT, что делает его уникальным среди алгоритмов третьего раунда NIST.

SABER использует обучение с округлением (learning with rounding), которое не требует выборки ошибок из распределения. Это уменьшает потребность в псевдослучайности и упрощает реализацию. Кроме того, безопасность схемы опирается всего на один базовый элемент, что позволяет легко адаптировать алгоритм к различным уровнем требований по безопасности.

SABER обеспечивает значения core-SVP, сходные с CRYSTALS-KYBER, и характеризуется простыми операциями, что делает его значительно проще в реализации. SABER работает в константное время, поэтому атаки по времени выполнения не влияют на безопасность схемы. Он рассматривается как потенциально очень подходящий для анонимных коммуникаций.

Все целые значения вычисляются по модулю 2, что снижает пропускную способность алгоритма; в сочетании с LWR это делает сжатие PKE криптографически обоснованным. По итогам третьего раунда стандартизации NIST предлагается улучшить SABER, проверив его устойчивость к атакующим побочным каналам и устойчивость к неправильному использованию. Это один из наиболее перспективных кандидатов на стандартизацию.

6.1.3 NTRU (финалист)

NTRU — это решёточный криптосистемный финалист, чья безопасность не выводится из сложности задач Ring Learning With Errors (RLWE) или Module Learning With Errors (MLWE). Это отличает его от других решёточных криптосистем. Вместе с Classic McEliece он является одним из самых старых алгоритмов среди всех поданных на конкурс. Благодаря возрасту существует обширная исследовательская литература, посвящённая атакам и оценке сложности задач, что повышает доверие к алгоритму. Текущая заявка NTRU третьего раунда представляет собой объединение заявок второго раунда NTRUEncrypt и NTRU HRSS-KEM [54].

Для NTRU отсутствует формальное сведение «в худшем случае $\rightarrow$ в среднем». В нём используются две модели оценки стоимости вычислений. Нелокальная модель использует метрику, аналогичную core-SVP, которой пользуются другие решёточные схемы. Также существует локальная модель, которая даёт более высокие оценки безопасности. Нелокальная модель обеспечивает максимум 4-ю категорию безопасности.

Алгоритм работает быстрее и компактнее, чем широко используемый RSA. В схеме присутствуют некоторые избыточности, которые можно убрать ценой утраты идеальной корректности. Параметризация NTRU исследуется с 1990-х годов, поэтому он является одной из наиболее хорошо описанных и проверенных криптосистем третьего раунда. Дополнительным преимуществом является отсутствие ограничений интеллектуальной собственности: большинство патентов, связанных со структурами алгоритма, уже истекли.

6.1.4 CRYSTALS-DILITHIUM

CRYSTALS–DILITHIUM — это алгоритм цифровой подписи, относящийся к криптографическому набору Cryptographic Suite for Algebraic Lattices [54]. Сложность алгоритма опирается на задачу коротких целых решений (SIS) и задачу модуля "обучения с ошибками" (MLWE). Его стойкость к атакам с адаптивным выбором сообщений основана на трудности решёточных задач над модульными решётками. Дизайн схемы основан на подходе Fiat–Shamir with Aborts, однако в ней используется равномерное распределение вместо гауссовского. Это делает реализацию значительно проще в вычислительном отношении, что даёт преимущество по сравнению с её основным конкурентом — FALCON.

DILITHIUM обладает наименьшей суммарной длиной открытого ключа и подписи среди решёточных схем цифровой подписи, использующих только равномерное распределение [56]. У DILITHIUM также самые низкие показатели core-SVP среди всех решёточных криптосистем третьего раунда, и на данный момент он не достигает уровня безопасности NIST Level 5. Тем не менее, DILITHIUM демонстрирует хорошие результаты в практических экспериментах и доступен для использования с различными наборами параметров.

6.1.5 FALCON (финалист)

Fast-Fourier Lattice-based Compact Signature over NTRU — это решёточная криптосистема. Её безопасность основана на трудности задачи кратчайшего целочисленного решения (Shortest Integer Solution, SIS) в решётках NTRU. Конструкция алгоритма опирается на решётки NTRU и использует "ловушечный" алгоритм выборки Fast Fourier sampling. Схема представляет собой комбинацию решёток NTRU, быстрой выборки по Фурье и структуры GPV (решёточные схемы подписи формата "хэшируй-и-подписывай").

Существуют доказательства стойкости FALCON как в модели случайного оракула (ROM), так и в квантовой модели случайного оракула (QROM). Алгоритм требует сложной реализации, включающей большое количество операций с плавающей точкой, древовидные структуры данных и выборку из дискретных гауссовских распределений.

FALCON обладает минимальными требованиями к пропускной способности среди всех схем цифровой подписи. Он обеспечивает высокую скорость создания подписи и проверки подписи, однако этап генерации ключей работает медленнее. FALCON можно легко интегрировать в существующие протоколы и приложения с хорошими показателями производительности. После второго раунда алгоритм получил реализацию с постоянным временем выполнения.

NIST включил FALCON в финальный список кандидатов в категории цифровых подписей [54]. Планируется стандартизировать либо FALCON, либо DILITHIUM. Дополнительный анализ необходим в части ошибок, связанных с операциями с плавающей точкой, и уязвимостей к побочным каналам. Также требуется более детальное изучение используемого выборщика (sampler). Алгоритм генерации ключей FALCON использует менее 30 KB оперативной памяти [57].

6.2 Кандидаты криптогафии на основе корректирующих кодов

6.2.1 Classic McEliece

Classic McEliece — самый старый криптосистемный кандидат, представленный в третьем раунде стандартизации NIST PQC [54]. Он основан на криптосистеме McEliece 1979 года, использующей скрытые коды Гоппы. Изначальная схема не разрабатывалась под ограничения массового применения и обеспечивала безопасность в модели OW-CPA (one-way chosen-plaintext attack), то есть противник не может эффективно восстановить сообщение по известным открытому ключу и шифртексту при случайном выборе сообщения [58]. Текущее представление модифицирует оригинальную систему, обеспечивая более эффективную реализацию и CCA-защищённость, что частично ослабляет прежнюю OW-CPA стойкость. Современный вариант представляет собой объединение схем NTS-KEM и Classic McEliece первого раунда.

Classic McEliece уникален тем, что за десятилетия исследований не было достигнуто *никакого* прогресса в атаках на эту криптосистему, несмотря на рост вычислительных ресурсов. Она использует строгие методы преобразования из PKE в IND-CCA2 KEM, что обеспечивает стойкость в ROM, а при корректной параметризации — и в QROM. Для защиты от атак на хеш-функции применяются хорошо изученные, неструктурированные хеш-функции.

Classic McEliece уязвим к атакам декодирования по множеству позиций (Information Set Decoding), что устраняется выбором такого числа ошибок, чтобы в шифртексте не возникало неискажённых участков сообщения. Устойчивость к атакам по сторонним каналам обеспечивается реализацией с постоянным временем выполнения. Схема также стойка против CCA-атак благодаря хешированию ошибок, вносимых в шифртекст; при условии криптостойкости хеш-функций этот метод работает существенно медленнее, чем другие возможные атаки.

Classic McEliece генерирует очень маленькие шифртексты — около 256 байт, что делает его удобным для внедрения в сетевые протоколы. Однако главным недостатком схемы является огромный размер открытого ключа — порядка 1.5 мегабайт.

6.2.2 BIKE (Альтернативный кандидат)

BIKE — это кодовый алгоритм для механизмов инкапсуляции ключей. Название является аббревиатурой от *Bit Flipping Key Encapsulation*. Алгоритм основан на квазициркульных кодах с матрицей проверок умеренной плотности (Quasi-Cyclic Moderate Density Parity-Check). Он берёт идеи из схемы McEliece. Перед финальной подачей ко второму раунду разработчики существенно изменили алгоритм: уменьшили полосу пропускания и представили новый декодер (Black Gray Flip) [59].

Безопасность BIKE опирается на сложную задачу кодовой теории кодирования — задачу различения. Алгоритм предоставил параметры только для категорий безопасности 1–4, но не для категории 5. BIKE базируется на предположениях о сложности задач Quasi-Cyclic Syndrome Decoding и QCCF. При этих предположениях схема IND-CPA-стойкая. Кроме того, её IND-CCA-стойкость доказана при условии корректности декодера. Устойчивость к атакам опреде-

ляется через атаки декодированием по множеству позиций и оценки их сложности. Тем не менее существуют риски атак по сторонним каналам и вопросы CCA-безопасности — в первую очередь из-за недостаточной уверенности в новом декодере.

Поскольку BIKE недавно изменила архитектуру, полноценно оценить её безопасность по широкому спектру направлений пока невозможно. Однако NIST рассматривает её как перспективного кандидата и включил в список альтернативных алгоритмов третьего раунда [54]. После устранения замечаний по безопасности и повышению доверия к декодеру алгоритм может быть стандартизован. BIKE рассматривается как потенциально надёжный резервный вариант на случай обнаружения уязвимостей в решениях на решётках.

6.2.3 HQC (Альтернативный кандидат)

Hamming Quasi-Cyclic (HQC) — это криптосистема, основанная на теории кодирования для шифрования с открытым ключом. Она базируется на сложности *decisional quasi-cyclic syndrome decoding with parity* — задачи декодирования синдрома в квазициркульных кодах. HQC является IND-CPA-стойкой, и авторы также заявляют её CCA2-стойкость [54].

Разработчики HQC представили новый декодер, основанный на кодах Рида–Маллера и Рида–Соломона. Это позволило существенно уменьшить размер ключей. Несмотря на сокращение, публичный ключ и шифртест HQC всё ещё в 1.6–2 раза и 4–5 раз больше, чем у BIKE соответственно. Пропускная способность HQC хуже, чем у BIKE, однако генерация ключей и процедуры декапсуляции работают значительно быстрее. Алгоритм использует жёстко заданные таблицы для ускорения арифметики в $GF(2^m)$, а также имеет декодирование с постоянным временем выполнения.

HQC раньше был уязвим к атаке по стороннему каналу через выбранный шифртест, но эта уязвимость была устранена постоянным временем декодирования.

Основные недостатки HQC включают ненулевую вероятность ошибки при расшифровании, крупные шифртесты по сравнению с BIKE и большие открытые ключи. Атаки на метрику Хэмминга изучаются уже более 50 лет, что повышает уверенность в криптографической стойкости HQC [60]. HQC представляет собой эффективный алгоритм с декодированием в постоянное время.

HQC был выбран как альтернативный кандидат третьего раунда благодаря детально проработанным аспектам безопасности среди кодовых алгоритмов. Однако он не был включён в финал из-за сравнительно слабых показателей производительности.

6.3 Кандидаты на основе многомерной криптографии

6.3.1 Rainbow (финалист)

Rainbow — это алгоритм цифровой подписи, основанный на варианте несбалансированной схемы «масло–уксус» (Unbalanced Oil and Vinegar, UOV). С момента появления в 2005 году схема практически не изменялась. Rainbow характеризуется малыми размерами подписей, а также очень быстрой процедурой подписания и проверки. Многоуровневая структура UOV делает его естественно устойчивым к атакам по сторонним каналам и защищённым от классических атак на UOV, таких как атака Кипниса–Шамира. Однако усложнение структуры привело к новым возможностям для эксплуатации алгоритма. После 2008 года новые атаки отсутствовали, и считалось, что существующие атаки можно нейтрализовать выбором корректных параметров.

В настоящий момент Rainbow соответствует уровням безопасности NIST 1, 3 и 5 и считается NP-трудным [54]. Однако после подачи в раунд 3 были опубликованы две атаки Уарда Бёлленса. Первая атака снижает стойкость Rainbow-1, Rainbow-3 и Rainbow-5 на 7, 4 и 19 бит соответственно; она основана на модификации классической атаки Кипниса–Шамира и также затрагивает схему подписи UOV. Вторая атака ещё сильнее — она уменьшает стойкость Rainbow-1, Rainbow-3 и Rainbow-5 на 20, 40 и 55 бит соответственно.

В условиях появления этих атак маловероятно, что NIST утвердит текущие параметры схемы. Разработчикам теперь требуется подобрать новые параметры, обеспечивающие требуемый уровень безопасности. Дополнительным недостатком Rainbow являются очень крупные открытые и закрытые ключи — их размер может достигать 1.8 МБ.

6.3.2 GeMSS (альтернативный кандидат)

GeMSS (A Great Multivariate Short Signature) — это схема цифровой подписи, основанная на парадигме «большого поля». Схема заявляет стойкость уровня EUF-CMA благодаря универсальной нефиксифицируемости примитива HFEv (Hidden Field Equations). GeMSS была вдохновлена алгоритмом QUARTZ и использует современные результаты в области многомерной криптографии для повышения эффективности и безопасности.

GeMSS опирается на хорошо изученную математическую задачу. Схема обеспечивает быстрое вычисление проверки подписи и формирует наиболее короткие подписи среди всех кандидатов. Однако ей требуются большие открытые ключи и длительное время генерации подписи. Кроме того, схема сложна для реализации на малопроизводительных устройствах. Крупные размеры открытых ключей затрудняют её использование в протоколах TLS и SSH.

Авторы GeMSS продолжают работу над повышением производительности алгоритма [61]. Основным конкурентом является Rainbow, который более эффективен на низкопроизводительных устройствах, обладает хорошей безопасностью и меньшими размерами ключей. В случае серьёзных уязвимостей в Rainbow именно GeMSS станет основным кандидатом на стандартизацию [54]. По этой причине NIST включил GeMSS в список альтернативных алгоритмов.

6.4 Кандидаты криптографии на основе изогений

6.4.1 SIKE (альтернативный кандидат)

SIKE является единственным алгоритмом, основанным на эллиптических кривых. Несмотря на то, что алгоритмы на эллиптических кривых легко взламываются алгоритмом Шора, SIKE решает эту проблему, используя псевдослучайные суперсингулярные изогенные кривые, которые обеспечивают взаимнооднозначное отображение точек, а не умножение точки на скаляр. Благодаря близости SIKE к ECC и протоколам Диффи–Хеллмана, в случае стандартизации он станет одним из самых простых для внедрения в существующие системы.

SIKE имеет самый маленький размер ключей среди всех алгоритмов третьего раунда (около 750 байт), даже для высоких уровней безопасности. Благодаря широким исследованиям эллиптических кривых выбор параметров и защита от побочных атак значительно проще.

Основным недостатком SIKE является сравнительно небольшое количество исследований, посвящённых поиску изогеней для эллиптических кривых, поэтому уверенность в стойкости ниже, чем у других классов алгоритмов. Кроме того, SIKE примерно на порядок медленнее большинства остальных кандидатов.

В настоящее время вокруг SIKE продолжаются дискуссии, поскольку были обнаружены атаки на упрощённые версии алгоритма, что вызывает вопросы о его долгосрочной стойкости. Однако разработчики отмечают, что для текущих параметров, представленных в рамках конкурса, алгоритм сохраняет конкурентный уровень безопасности [54].

6.5 Кандидаты на основе хешированных подписей

6.5.1 SPHINCS+ (альтернативный кандидат)

SPHINCS+ — это схема цифровой подписи, основанная на хеш-функциях. Она представляет собой усовершенствованную версию SPHINCS и обеспечивает защиту от мультицелевых атак [62]. Её безопасность полностью сводится к стойкости используемой хеш-функции.

SPHINCS+ принадлежит к классу хешированных схем подписи, изучавшихся ещё до появления современных криптосистем вроде RSA. В постквантовом мире SPHINCS+ считается наименее вероятным кандидатом среди всех схем третьего раунда, для которых возможно появление криптоаналитической атаки. Алгоритм имеет ясную структуру и чёткие спецификации.

SPHINCS+ уязвим для атак по отказам и атак по побочным каналам. Разработчики работают над уменьшением этих уязвимостей и улучшением производительности.

Генерация подписи заметно медленнее, чем у других кандидатов, а сами подписи значительно больше по размеру. Даже самые короткие подписи SPHINCS+ в четыре раза больше подписей DILITHIUM и требуют примерно в тысячу раз больше вычислительных ресурсов.

Из-за низкой скорости и большого размера подписи интеграция SPHINCS+ в TLS потребует серьёзной переработки, поскольку нынешние протоколы ориентируются на схемы с открытым ключом. NIST рассматривает SPHINCS+ как резервный вариант на случай провала основных кандидатов третьего раунда [54]. Также он может быть стандартизован для систем, где требуется максимально высокий уровень безопасности и допустимо снижение скорости и увеличение размера подписи.

6.6 Прочие криптографические кандидаты

6.6.1 PICNIC

PICNIC — это алгоритм цифровой подписи, обеспечивающий стойкость как к квантовым, так и к классическим атакам. Его безопасность не опирается на какие-либо числовые или алгебраические трудные задачи. Вместо этого PICNIC основан на системе доказательств с нулевым разглашением и примитивах симметричной криптографии, таких как хеш-функции и блочные шифры [63]. Текущая реализация PICNIC базирует свою безопасность на безопасности хеш-функции и блочного шифра LowMC.

PICNIC характеризуется большими размерами подписей и медленными процедурами формирования и проверки подписи. При этом алгоритм обладает маленькими публичными ключами. Его реализация уязвима к серьёзным атакам по побочным каналам. PICNIC имеет модульную архитектуру: все его строительные блоки могут заменяться. Рассматривались варианты схемы, основанные на AES, но они требуют значительно большего размера ключей.

NIST считает PICNIC ещё недостаточно зрелым [54]. Его дизайн активно развивается, и алгоритм остаётся концептуально новым. Существенным преимуществом является гибкость: компоненты схемы можно подбирать отдельно. PICNIC демонстрирует высокий потенциал развития как в части повышения производительности, так и в части укрепления безопасности. NIST включил его в список альтернативных кандидатов.

7 Сравнение производительности постквантовых криптосистем

Таблица 1 показывает уровни безопасности, обеспечиваемые постквантовыми криптосистемами, рассмотренными в данном исследовании. Производительность алгоритма обычно измеряется числом тактов, необходимых для выполнения, а также объёмом аппаратных ресурсов (логики/площади), требуемых для его аппаратной реализации. На производительность также влияет длина ключа или любой другой внутренний параметр, используемый алгоритмом. По этой причине многие схемы имеют несколько вариантов с различными параметрами, что существенно влияет на их производительность.

Таблица 1 – Семейства и уровни безопасности PQC-алгоритмов

Алгоритм	**Семейство алгоритмов**	**Уровень безопасности**
Classic McEliece	Кодовые	5 [64]
SABER	Решётки	1, 3, 5 [64]
CRYSTALS-Kyber	Решётки	1, 3, 5 [64]
NTRU-HRSS	Решётки	1 [64]
NTRU-HPS	Решётки	1, 3, 5 [65]
CRYSTALS-Dilithium	Решётки	1, 2, 3 [64]
SIKE	Изогении	1, 2, 3, 5 [65]
SPHINCS+	Хеш-подписи	1, 3, 5 [64]

Для примера, kyber512, kyber768 и kyber1024 по сути являются одним и тем же алгоритмом, но используют разную длину ключа и другие параметры, что влияет на производительность и уровень безопасности шифра. Чтобы сравнение было корректным, сравнивать нужно алгоритмы с одинаковым уровнем безопасности.

В анализе, проведённом Канадом Басу и др. [65], авторы установили, что среди алгоритмов, использующих механизмы инкапсуляции ключей, NTRU-HRSS обладает самой высокой латентностью (в тактах процессора). Среди алгоритмов электронных подписей уровня 1 наименьшую латентность показывает CRYSTALS-Dilithium. Для операций декапсуляции NTRU-HRSS (уровень 3) и Classic McEliece (уровень 5) имеют наибольшую задержку, в то время как CRYSTALS-Dilithium демонстрирует наименьшую, что отражено в таблице 2. Эти результаты получены при тестировании алгоритмов на ПЛИС Virtex-7.

Из этих результатов можно сделать вывод, что при использовании базовых алгоритмов без модификаций CRYSTALS-Dilithium является хорошим кандидатом для IoT как для инкапсуляции, так и для декапсуляции. Однако для декапсуляции практически применимы только алгоритмы уровня 1, поскольку все протестированные алгоритмы уровня 5 обладают слишком большой задержкой. Поскольку IoT обычно предполагает аппаратные реализации, тестирование на FPGA является хорошим ориентиром для оценки производительности алгоритма на устройстве IoT. Техники такие как loop unrolling и loop pipelining могут дополнительно снизить общую задержку.

Брайан Хешион и др. [66] использовали расширение eXternal Benchmarking eXtension (XXBX) для тестирования производительности PQC-алгоритмов на 32-битном ARM-микроконтроллере EK-TM4C123GXL. Алгоритмы оценивались по потреблению RAM, ROM, скорости (тактовые циклы) и энергии.

По использованию RAM NTRU является самым дорогим алгоритмом, а наименьший объём оперативной памяти потребляет Kyber (во всех трёх уровнях безопасности). По ROM NTRU также самый тяжёлый, а следующим идёт Saber, хотя он использует лишь половину объёма ROM по сравнению с NTRU. По скорости (числу тактов) SIKE является самым медленным с большим отрывом — одна операция инкапсуляции занимает семь минут. Kyber, Saber и NTRU имеют сопоставимую производительность, причём уровни безопасности выше требуют немного больше времени, что показано в таблице 2. Та же картина наблюдается и в энергопотреблении, поскольку более длительное выполнение алгоритма означает и более длительное потребление мощности.

Таблица 2 – Сравнение производительности постквантовых алгоритмов на разных платформах

Алгоритм	**Уровень безопасности**	**Задержка капсулирования в тактах (FPGA) [64]**	**Задержка декапсулирования в тактах (FPGA) [64]**	**Задержка капсулирования в тактах (микроконтроллер) [66]**	**Задержка декапсулирования в тактах (микроконтроллер) [66]**
CRYSTALS Kyber	1	5.6×10^4	5.3×10^4	8.0×10^6	2.0×10^6
	3	–	–	9.0×10^6	3.0×10^6
	5	–	–	1.1×10^7	5.0×10^6
CRYSTALS DILITHIUM	1	6.0×10^5	5.3×10^3	–	–
NTRU-HPS	1	–	–	9.0×10^6	1.5×10^6
	3	–	–	1.0×10^7	2.0×10^6
NTRU-HRSS	1	1.4×10^6	1,003,222	9.5×10^6	2.5×10^6
Saber	1	–	–	8.0×10^6	2.5×10^6
	3	4.9×10^5	89,392	9.0×10^6	3.0×10^6
Classic McEliece	5	5.1×10^6	146,126,996	–	–
SIKE	2	–	–	$> 3.3 \times 10^{10}$	$> 3.3 \times 10^{10}$
SPHINCS+	1	6.2×10^8	937,935	–	–

8 Проблемы миграции к постквантовой криптографии

Необходимость перехода к постквантовой криптографии обсуждается очень активно, однако вопрос перехода от классических криптографических систем к постквантовым до сих пор остаётся недостаточно изученным. Поскольку переход к алгоритмам PQC неизбежен, становится особенно важным исследовать способы его реализации. Однако из-за масштабности сети Интернет и цифровизации практически всех сфер такой переход окажется значительно сложнее, чем кажется. Ниже приведены ключевые сложности, связанные с переходом к PQC.

а) Непредсказуемые сроки развития квантовых вычислений: существует риск ускоренного появления квантовых компьютеров, темпы разработки которых могут превысить ранее ожидаемые.

б) Сложные критерии перехода к PQC: переход от нынешних алгоритмов с открытым ключом к постквантовым решениям не является простым обновлением программного обеспечения. Этот факт подчёркивается и в документах NIST [67].

в) Атаки типа «записать сейчас — взломать позже»: перехваченные сегодня зашифрованные данные могут быть расшифрованы квантовыми компьютерами спустя годы, если к тому моменту информация сохранит свою ценность.

г) Выбор релевантных стандартов NIST: при подготовке к этому переходу необходимо учитывать последствия внедрения новых стандартов PQC ещё на стадии стандартизации.

Существует ещё много работы, необходимой для лучшего понимания вопросов перехода и решения задач интеграции, безопасности, производительности и других аспектов, чтобы гарантировать безопасное внедрение квантовых решений в экосистему глобальной индустрии. При рассмотрении исследовательских проблем следует выделить две пересекающиеся области:

а) *Исследования по переходу на постквантовую криптографию*: Это исследования того, как кандидатные алгоритмы могут быть использованы в определённых условиях, а также исследования безопасного перехода для конкретной области криптографического применения.

б) *На пути к науке криптографической гибкости*: Организации используют криптографическую гибкость как подход к шифрованию данных, позволяющий быстро реагировать на криптографические атаки. Цель криптоагильности — переходить на новые криптографические стандарты без необходимости в крупных изменениях инфраструктуры. Гибкость в криптографическом домене проявляется в обобщённом подходе к переходу на PQC различными способами.

Существует значительное сходство между широким пространством перехода к PQC и областью криптографической гибкости. Эти сходства демонстрируют весь спектр трудностей, которые необходимо решить, чтобы обеспечить переход криптографических систем на постквантовые алгоритмы. В то же время имеются области, где эти две дисциплины не совпадают или вовсе не пересекаются.

8.1 Предмиграционные вызовы PQC

Переход от современных алгоритмов открытого ключа к постквантовой криптографии не является простой задачей, подобной обновлению программного обеспечения до новой версии, что также было признано NIST [67]. PQC затрагивает размеры ключей, шифртестов и подписей, а также требования к взаимодействию и вычислениям. В итоге NIST заявил, что будет выбран только один алгоритм в качестве основной замены, поскольку остальные алгоритмы имеют различные компромиссы по размерам ключей и вычислительным требованиям. В целом NIST считает безопасным предоставление нескольких вариантов в рамках новых стандартов PQC.

Ниже представлены несколько областей, которые могут получить выгоду от дополнительных исследований:

1. *Соображения производительности.* Алгоритмы PQC требуют больше вычислительных ресурсов, памяти, хранилища и пропускной способности, поэтому оценка производительности в различных сценариях внедрения является критически важной.

2. *Соображения безопасности.* Переход на новые алгоритмы создаст множество новых проблем безопасности. Поскольку PQC изучена хуже, чем существующие системы RSA и ECC, возникают вопросы о размерах ключей, времени вычислений и других параметрах. Ещё одна важная область — криптоанализ PQC-алгоритмов.

3. *Соображения реализации.* Реализация PQC-алгоритмов окажется сложнее, чем кажется. Сложность математических конструкций переносится на архитектуру конкретных платформ и устройств. Существуют несколько методов внедрения новых алгоритмов:

1) Один из наиболее исследованных методов — гибридный подход, при котором используются два криптографических алгоритма: один из текущих стандартов и один из кандидатов PQC. Это позволяет защититься от атак типа «записать сейчас — расшифровать позже», сохраняя при этом устойчивость к известным атакам в ранних этапах перехода. Преимущество метода — возможность сохранять сертификаты соответствия во время перехода. Недостаток — значительное увеличение требований к вычислениям, памяти и коммуникациям.

2) Другой подход — использование механизма согласования наборов шифров (cipher suite negotiation), применяемого, например, в TLS [68]. Во время протокола рукопожатия стороны обмениваются списками поддерживаемых наборов шифров и выбирают наиболее надёжный, который поддерживают обе стороны. Можно согласовывать версии наборов шифров, размеры ключей и параметры [69]. В эту структуру можно добавить новые алгоритмы PQC и удалить устаревшие.

3) *Формальное моделирование.* Формальное моделирование перехода в криптографии — крайне востребованная область исследований. Независимо от выбранной стратегии перехода, безопасность системы остаётся проблемой. Формальные методы позволяют оценивать безопасность фундаментальным образом. Такое моделирование также необходимо для оценки безопасности внедрения механизмов перехода в широко используемые криптографические протоколы. Например, протокол TLS может быть декомпозирован на компоненты, устойчивые к квантовым атакам, для исследовательских целей.

4) *Автоматизированные инструменты.* Из-за масштабов криптографической инфраструктуры переход невозможен без автоматизированных средств. Поэтому исследования и разработки в этой области также крайне необходимы.

8.2 Расширение области криптографической гибкости

Криптографическая гибкость является устоявшейся концепцией в исследовательском сообществе, однако масштабность миграции к постквантовой криптографии делает необходимым расширение этой области исследований. Следует разработать науку криптографической гибкости, которая охватывала бы более широкий спектр целей, вычислительных доменов, форм гибкости и временных масштабов. Для обеспечения криптографической гибкости необходимо разрабатывать и исследовать соответствующие архитектуры и фреймворки в широком диапазоне вычислительных контекстов, а также формировать согласованные интерфейсы, удовлетворяющие потребностям различных категорий пользователей.

Существуют и другие важные вызовы, такие как внедрение гибкости после определения корректной области внутри криптографического решения. Система криптографической гибкости должна уметь предсказывать области, в которых потребуется изменение криптографических примитивов, а также обеспечивать механизмы для своевременного и безопасного переключения между ними.

9 Перспективы и будущие вызовы

Процесс стандартизации NIST PQC проведёт свою следующую встречу в июне 2021 года. На ней будут представлены новые комментарии по всем финалистам и альтернативным алгоритмам. Постквантовая криптография — это новая область исследований, которая в ближайшие годы станет повсеместной.

Существует шесть основных семейств, на основе которых разрабатываются квантовые криптосистемы. Все они имеют свои преимущества и недостатки с точки зрения битовой безопасности, размеров ключей, накладных расходов по памяти и вычислительных требований. Исследователи могут находить новые математические задачи, трудные для квантовых компьютеров, и создавать на их основе новые криптосистемы. Также возможно выявление новых задач, которые требуют меньших накладных расходов и меньших вычислительных ресурсов.

Исследователи активно работают над предложенными NIST криптосистемами. В течение долгого времени они проводят криптоанализ алгоритмов, изучают проблемы реализации, математические атаки, атаки по побочным каналам и так далее. Это важнейшая область исследований, и криптоанализ финалистов и альтернатив позволит повысить безопасность алгоритмов и предотвратить стандартизацию потенциально небезопасных решений.

Существующие алгоритмы также могут быть оптимизированы за счёт новых аппаратных и программных решений. Это может помочь уменьшить временные и пространственные накладные расходы алгоритмов. Исследователи могут разрабатывать методы снижения вычислительных требований, создавая специализированное оборудование или улучшая программные реализации. Также возможно уменьшить накладные расходы по памяти.

Перспективным направлением является разработка низкоресурсной постквантовой криптографии. Рост числа IoT-устройств делает низкоресурсные криптографические алгоритмы особенно актуальными. Такие устройства обмениваются зашифрованными данными, так как часть информации может быть конфиденциальной (например, коммерческие тайны). Поэтому требуется постквантовая низкоресурсная криптография — алгоритмы, устойчивые к квантовым атакам и требующие минимальных ресурсов.

Числовые методы и алгоритмические исследования — ещё одно направление, связанное с постквантовыми сложными задачами. Учёные могут работать над новыми математическими техниками взлома задач, на которых основана безопасность постквантовых криптосистем. Значительное количество исследований проводится по многомерным квадратичным отображениям и решётчатым задачам. Исследователи предлагают новые методы решения этих трудных задач.

Квантовый криптоанализ и разработка новых квантовых алгоритмов — ещё одна область, исследованная пока очень слабо. Шор предложил два квантовых алгоритма, которые нарушили безопасность современных асимметричных алгоритмов. Возможно, для постквантовых криптосистем также будут обнаружены новые квантовые алгоритмы, способные скомпрометировать их безопасность. Поэтому необходимо проводить дальнейшие исследования в этом направлении.

10 Заключение

Квантовая криптография является зарождающейся и стремительно развивающейся областью. Многие компании по всему миру вкладывают ресурсы в увеличение знаний и развитие практик, связанных с постквантовой безопасностью. В связи с разнообразием интересов и исследований по всему миру возникла необходимость систематизировать текущее внимание к квантовой безопасности и представить обзор современных достижений в этой области.

Симметричные алгоритмы являются устойчивыми как к классическим, так и к квантовым атакам (AES-256 используется для характеристики наивысшего уровня безопасности всех новых алгоритмов), однако их трудно реализовывать в квантовых схемах, особенно учитывая, что существующие квантовые машины способны обрабатывать лишь очень малые размеры сообщений (порядка 20 бит). Дальнейшие достижения в области квантовых технологий могут расширить эти возможности, что позволит эффективнее реализовывать симметричные криптосистемы, такие как AES.

Для симметричных криптосистем квантовые атаки требуют квантового оракула. Пока симметричная криптография не реализована с использованием квантовых оракулов, она остаётся безопасной против квантовых атак. Вся современная классическая информация остаётся защищённой. Однако влияние квантовых вычислений на асимметричные криптосистемы куда серьёзнее. Для их взлома не требуется квантовая реализация самих алгоритмов — злоумышленнику достаточно локальных квантовых ресурсов, чтобы эксплуатировать уязвимости и взламывать шифрование. Это означает, что все данные, зашифрованные асимметричными алгоритмами, становятся уязвимыми при появлении достаточно мощных квантовых компьютеров.

На данный момент квантовые алгоритмы уже существуют для всех крупных асимметричных криптосистем, и вопрос состоит лишь в том, когда они будут полностью сломаны. Исследователи пытаются либо повысить сложность задач, на которых основаны современные алгоритмы (RSA, ECC), либо найти новые задачи, достаточно трудные даже для квантового компьютера. Второй подход получил наибольшее развитие, и активные исследования ведутся в обла-

стях решётчатых задач, кодов с исправлением ошибок, некоммутативных криптосистем и криптосистем на основе хеширования. Однако многие предлагаемые алгоритмы сложно реализовать, и их производительность требует оптимизации для широкого практического применения.

В 2017 году NIST объявил международный конкурс будущего стандарта для асимметричной криптографии. Было отмечено, что необходимость определения стандарта становится всё более острой, поскольку время разработки и внедрения алгоритмов не должно превышать время, за которое появятся системы, способные взломать используемые сегодня криптосистемы. Конкурс NIST по постквантовой криптографии включал три раунда, причём все материалы были публичными. Это позволило сообществу выявить множество уязвимых алгоритмов, и лишь 15 из первоначальных 69 дошли до финального раунда.

В настоящем обзоре были рассмотрены эти 15 алгоритмов. NIST объявил приём статей для третьей конференции по стандартизации PQC, которая состоится в июне 2021 года. На конференции планируется обсуждение различных аспектов алгоритмов и получение ценной обратной связи, которая поможет принять решения о стандартизации.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Humble T. S. Consumer Applications of Quantum Computing: A Promising Approach for Secure Computation, Trusted Data Storage, and Efficient Applications // IEEE Consumer Electronics Magazine. — 2018.

2. Grover L. K. A Fast Quantum Mechanical Algorithm for Database Search // arXiv:quant-ph/9605043. — 1996. — arXiv preprint.

3. Bernstein D. J. Introduction to Post-Quantum Cryptography // Post-Quantum Cryptography. — Springer, 2009. — С. 1—14.

4. Mavroeidis V., Vardanyan A., Jøsang A. The Impact of Quantum Computing on Present Cryptography // International Journal of Advanced Computer Science and Applications. — 2018.

5. Cryptography in Everyday Life. — 2021. — Accessed: 2021-02. https://laits.utexas.edu/anorman/BUS.FOR/course.mat/SSim/life.html.

6. Paar C., Pelzl J. The Data Encryption Standard (DES) and Alternatives // Understanding Cryptography. — Springer, 2010. — С. 55—86.

7. Bhargavan K., Leurent G. On the Practical (In-)Security of 64-Bit Block Ciphers: Collision Attacks on HTTP over TLS and OpenVPN // Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security. — Vienna, Austria, 2016.

8. National Institute of Standards and Technology. Advanced Encryption Standard (AES) : FIPS Publication / NIST. — 2001. — № 197.

9. SANS Institute. VPN Security Whitepaper. — 2020. — Online; accessed December 2020. https://www.sans.org/reading-room/whitepapers/vpns/paper/1006.

10. Paar C., Pelzl J. Public-Key Cryptosystems Based on the Discrete Logarithm Problem // Understanding Cryptography. — Springer, 2010.

11. Paar C., Pelzl J. Elliptic Curve Cryptosystems // Understanding Cryptography. — Springer, 2010.

12. Wagner L. Basic Intro to Elliptic Curve Cryptography. — 2020. — Online; accessed December 2020. https://qvault.io/2020/09/17/very-basic-intro-to-elliptic-curve-cryptography/.

13. O. L. D. Side-Channel Attacks in ECC: A General Technique for Varying the Parametrization of the Elliptic Curve // Lecture Notes in Computer Science. T. 3156. — Berlin, 2004.

14. Giles M. What is Quantum Computing? — 2019. — Online; accessed 2019. https://www.technologyreview.com/2019/01/29/66141/what-is-quantum-computing/.

15. Feldman V. Basics of Quantum Mechanics. — 2003. — MIT OCW, 2003. https://ocw.mit.edu/courses/mathematics/18-435j-quantum-computation-fall-2003/lecture-notes/qc_lec02.pdf.

16. Yanofsky N. S. An Introduction to Quantum Computing // arXiv:0708.0261 [quant-ph]. — 2007.

17. Kuzyk M. G. Quantum No-Cloning Theorem and Entanglement // American Journal of Physics. — 2019.

18. Quantum Key Distribution (QKD) Protocols: A Survey / A. I. Nurhadi, N. R. Saputra [и др.] // 2018 4th International Conference on Wireless and Telematics (ICWT). — 2018.

19. Shor P. W. Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer // SIAM Journal on Computing. — 1997. — С. 1484—1509.

20. Hui J. QC — Simon's Algorithm. — 2019. — Online; accessed April 2019. https://jonathan-hui.medium.com/qc-simons-algorithm-be570a40f6de.

21. Bernstein E., Vazirani U. Quantum Complexity Theory // Proceedings of the 25th Annual ACM Symposium on Theory of Computing. — Association for Computing Machinery, 1993. — С. 11—20.

22. D-Wave Systems. D-Wave Publications. — 2020. — Online resource. https://www.dwavesys.com/resources/publications?type=technology.

23. CB Insights. Quantum Computing Report. — 2020. — Online resource. https://www.cbinsights.com/research/report/quantum-computing/.

24. IBM Research. IBM Quantum Roadmap. — 2020. — Online; accessed 2020. https://www.ibm.com/blogs/research/2020/09/ibm-quantum-roadmap/.

25. Quantum Exhaustive Key Search with Simplified-DES as a Case Study / M. Almazrooie [и др.] // SpringerPlus. — 2016.

26. Towards Post-Quantum Secure Symmetric Cryptography: A Mathematical Perspective : тех. отч. / X. Bogomolec [и др.] ; IACR Cryptology ePrint Archive. — 2019. — № 2019/2019.

27. Post-Quantum RSA / L. Valenta [и др.] // Post-Quantum Cryptography. — Springer, 2017.

28. Gidney C., Ekerå M. How to Factor 2048 Bit RSA Integers in 8 Hours Using 20 Million Noisy Qubits // arXiv:1905.09749 [quant-ph]. — 2019.

29. Blanda S. Shor's Algorithm – Breaking RSA Encryption. — 2014. — Online; accessed 2014. https://blogs.ams.org/mathgradblog/2014/04/30/shors-algorithm-breaking-rsa-encryption/.

30. Smith B. Pre- and Post-Quantum Diffie–Hellman from Groups, Actions, and Isogenies // arXiv:1809.04803 [cs]. — 2019.

31. Breaking Symmetric Cryptosystems Using Quantum Period Finding : тех. отч. / M. Naya-Plasencia [и др.] ; Cryptology ePrint Archive. — 2016.

32. Ajtai M. A Public-Key Cryptosystem with Worst-Case/Average-Case Equivalence // Proceedings of the 29th Annual ACM Symposium on Theory of Computing (STOC '97). — New York, 1997.

33. Nguyen P. Q., Stern J. Cryptanalysis of the Ajtai–Dwork Cryptosystem // Advances in Cryptology – CRYPTO '98. — Springer, 1998. — С. 223—242.

34. Goldreich O., Goldwasser S., Halevi S. Public-Key Cryptosystems from Lattice Reduction Problems // Advances in Cryptology – CRYPTO '97. — Springer, 1997.

35. Nguyen P. Q. Cryptanalysis of the Goldreich–Goldwasser–Halevi Cryptosystem // Advances in Cryptology – CRYPTO '99. — Springer, 1999. — С. 288—304.

36. Hoffstein J., Pipher J., Silverman J. H. NTRU: A Ring-Based Public Key Cryptosystem // International Algorithmic Number Theory Symposium. — 1998.

37. Regev O. The Learning With Errors Problem. — 2010. — Online survey. https://cims.nyu.edu/~regev/papers/lwesurvey.pdf.

38. Lamport L. Constructing Digital Signatures from a One-Way Function : тех. отч. / SRI International, Computer Science Laboratory. — 1979. — CSL—98.

39. Biswas S., Nandi S. K. McEliece Cryptosystem Implementation: Theory and Practice // PQCrypto 2008. — 2008.

40. McEliece R. J. A Public-Key Cryptosystem Based on Algebraic Coding Theory // DSN Progress Report, Jet Propulsion Laboratory. — 1978. — С. 114—116.

41. Schmeh K. Understanding and Explaining Post-Quantum Cryptography. — 2020.

42. Shpilrain V. Cryptanalysis of Stickel's Key Exchange Scheme // Proceedings in Cryptology. — Springer, 2008. — С. 283—288.

43. Stickel E. A New Method for Exchanging Secret Keys // Proceedings of the Third International Conference on Information Technology and Applications (ICITA 2005). — 2005.

44. Ding J., Schmidt D. Multivariate Public Key Cryptography // Post-Quantum Cryptography. — Berlin : Springer, 2009. — С. 193—241.

45. Efficient Algorithms for Solving Overdefined Systems of Multivariate Polynomial Equations / N. Courtois [и др.] // EUROCRYPT 2000. — 2000.

46. Kipnis A., Shamir A. Unbalanced Oil and Vinegar Signature Schemes // EUROCRYPT 1999. — Berlin, 1999.

47. Patarin J. The Oil and Vinegar Signature Scheme // Dagstuhl Workshop on Cryptography. — 1997.

48. Kipnis A., Shamir A. Cryptanalysis of the Oil and Vinegar Signature // CRYPTO '98. — 1998.

49. Ding J., Schmidt D. Rainbow, a New Multivariable Polynomial Signature Scheme // International Conference on Applied Cryptography and Network Security. — 2005.

50. Beullens W. Improved Cryptanalysis of UOV and Rainbow : тех. отч. / Cryptology ePrint Archive. — 2020.

51. Haakegaard R., Lang J. The Elliptic Curve Diffie–Hellman (ECDH). — 2015. — Online project report. http://koclab.cs.ucsb.edu/teaching/ecc/project/2015Projects/Haakegaard+Lang.pdf.

52. Jao D., De Feo L. Towards Quantum-Resistant Cryptosystems from Supersingular Elliptic Curve Isogenies // Post-Quantum Cryptography. — Berlin : Springer, 2011. — С. 19—34.

53. Report on Post-Quantum Cryptography : NISTIR / L. Chen [и др.] ; National Institute of Standards ; Technology. — 2016. — № 8105.

54. Status Report on the Second Round of the NIST Post-Quantum Cryptography Standardization Process : NISTIR / G. Alagic [и др.] ; National Institute of Standards ; Technology. — 2020. — № 8309.

55. CRYSTALS-Kyber / P. Schwabe [и др.]. — 2020. — Online; accessed December 2020. https://pq-crystals.org/kyber/index.shtml.

56. CRYSTALS-Dilithium / P. Schwabe [и др.]. — 2020. — Online; accessed 2020. https://pq-crystals.org/dilithium/index.shtml.

57. Falcon. — 2020. — Online; accessed 2020. https://falcon-sign.info/.

58. Classic McEliece: Introduction. — 2020. — Online; accessed December 2020. https://classic.mceliece.org/.

59. BIKE – Bit Flipping Key Encapsulation. — 2020. — Online; accessed December 2020. https://bikesuite.org.

60. HQC. — 2020. — Online; accessed December 2020. http://pqc-hqc.org/implementation.html.

61. GeMSS – A Great Multivariate Short Signature / A. Casanova [и др.]. — 2020. — Online; accessed December 2020. https://www-polsys.lip6.fr/Links/NIST/GeMSS.html.

62. SPHINCS+ / P. Schwabe [и др.]. — 2020. — Online; accessed December 2020. https://sphincs.org/.

63. Picnic / G. Zaverucha [и др.]. — 2020. — Online; accessed December 2020. https://microsoft.github.io/Picnic/.

64. NIST Post-Quantum Cryptography – A Hardware Evaluation Study : тех. отч. / K. Basu [и др.] ; Cryptology ePrint Archive. — 2019. — № 2019/047.

65. Implementation and Benchmarking of Round 2 Candidates in the NIST Post-Quantum Cryptography Standardization Process Using Hardware and Software/Hardware Co-design Approaches : тех. отч. / V. B. Dang [и др.] ; Cryptology ePrint Archive. — 2020. — № 2020/795.

66. Hession B., Kaps J.-P. Feasibility and Performance of PQC Algorithms on Microcontrollers // Second NIST PQC Standardization Conference. — 2019.

67. National Institute of Standards and Technology. Submission Requirements and Evaluation Criteria for the Post-Quantum Cryptography Standardization Process : тех. отч. / NIST. — 2017.

68. Rescorla E. The Transport Layer Security (TLS) Protocol Version 1.3 : тех. отч. / RFC Editor. — 2018. — RFC 8446.

69. Housley R. Guidelines for Cryptographic Algorithm Agility and Selecting Mandatory-to-Implement Algorithms : тех. отч. / RFC Editor. — 2015. — RFC 7696.