# 密码学算法详解（附加材料）

- NAT类型 / Типы NAT
- NAT的优缺点 / Преимущества и недостатки

防火墙技术 / Технологии межсетевых экранов

- 防火墙的基本概念 / Основные понятия
- 防火墙技术分类 / Классификация технологий
- 防火墙策略 / Политики МЭ
- DMZ（隔离区）/ DMZ

完整性验证 / Проверка целостности

- 什么是完整性? / Что такое целостность?
- 验证完整性的方法 / Методы проверки
- HMAC公式 / Формула HMAC

哈希函数要求与攻击 / Требования к хеш-функциям и атаки

- 安全哈希函数的要求 / Требования
- 生日攻击 / Атака дня рождения
- SHA家族对比 / Сравнение семейства SHA

重要公式总结 / Важные формулы

- RSA
- Diffie-Hellman

# 密码学算法详解（附加材料）

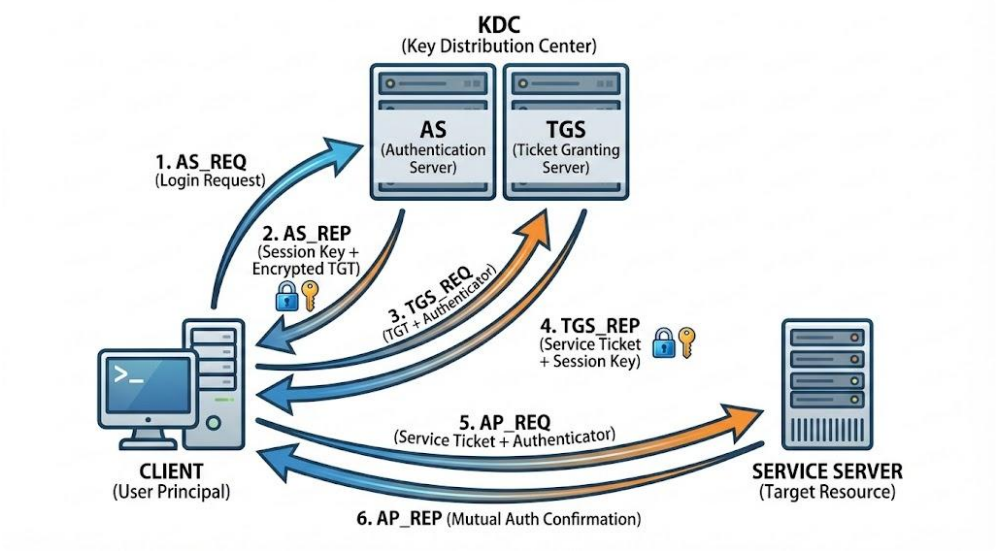
# Криптографические алгоритмы: Дополнительные материалы

## Kerberos 认证协议 / Протокол аутентификации Kerberos

这份图解展示了 Kerberos 协议如何通过可信第三方 (KDC) 实现安全的身份验证。以下是中俄对照的简洁流程说明：

| 步骤 (Step) | 中文说明 (Chinese Explanation) | Русский перевод (Russian Explanation) |
| --- | --- | --- |
| 总体概述 | Kerberos 认证流程图 | Схема процесса аутентификации Kerberos |
| 1. AS_REQ | **请求身份认证：** 客户端向 KDC 中的认证服务器 (AS) 发送请求，声明自己的身份。 | **Запрос аутентификации:** Клиент отправляет запрос серверу аутентификации (AS) в KDC, заявляя о себе. |
| 2. AS_REP | **获取 TGT (入场券)：** AS 验证用户后，发放一个加密的”票据授予票据” (TGT) 和临时会话密钥。 | **Получение TGT (пропуска):** После проверки AS выдает зашифрованный «билет на получение билетов» (TGT) и временный сеансовый ключ. |
| 3. TGS_REQ | **请求服务票据：** 客户端凭 TGT 向票据授予服务器 (TGS) 申请访问特定目标服务器的权限。 | **Запрос сервисного билета:** Клиент использует TGT, чтобы запросить у сервера выдачи билетов (TGS) доступ к конкретному целевому серверу. |
| 4. TGS_REP | **获取特定票据：** TGS 验证 TGT 无误后，发放针对目标服务器的”服务票据”。 | **Получение конкретного билета:** Проверив TGT, TGS выдает «сервисный билет» для целевого сервера. |
| 5. AP_REQ | **请求访问服务：** 客户端向目标服务服务器出示”服务票据”以申请访问。 | **Запрос доступа к услуге:** Клиент предъявляет «сервисный билет» целевому серверу для получения доступа. |
| 6. AP_REP | **双向确认：** 服务器验证票据合法性，双方完成相互身份确认，建立安全连接。 | **Взаимное подтверждение:** Сервер проверяет легитимность билета; стороны завершают взаимную аутентификацию и устанавливают защищенное соединение. |

### Kerberos 协议流程图 / Схема протокола Kerberos




Kerberos协议流程图

---

# RSA 算法例题集 / Задачи по алгоритму RSA

## 例题1：基础密钥生成 / Задача 1: Базовая генерация ключей

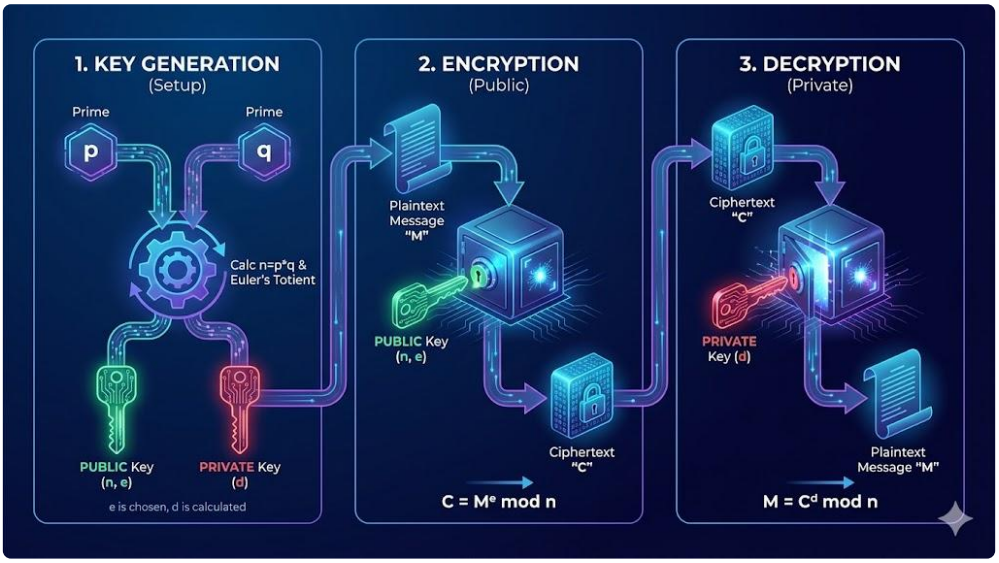
**题目 / Условие：** 已知两个质数 p = 3, q = 11，公钥指数 e = 7。计算 n, ϕ(n), d，写出公钥和私钥。

Даны два простых числа p = 3, q = 11, открытая экспонента e = 7. Вычислите n, ϕ(n), d, запишите ключи.

**解答 / Решение：**

1. $n = p \times q = 3 \times 11 = 33$
2. $\phi(n) = (p-1)(q-1) = 2 \times 10 = 20$
3. **计算 d：** $d \times 7 \equiv 1 \pmod{20} \rightarrow 7 \times 3 = 21 \equiv 1 \pmod{20} \rightarrow d = 3$
4. **公钥 / Открытый ключ：(e, n) = (7, 33) 私钥 / Закрытый ключ：(d, n) = (3, 33)**

RSA 算法示意图 / Схема алгоритма RSA




RSA算法示意图

## 例题2：加密和解密 / Задача 2: Шифрование и расшифрование

**题目 / Условие：** 使用例题1的密钥，对明文 M = 5 进行加密，然后解密验证。

**加密 / Шифрование（用公钥 e=7, n=33）：** – C = M^e \mod n = 5^7 \mod 33 – 5^4 = 625 \mod 33 = 31 – 5^7 = 31 × 25 × 5 \mod 33 = 775 × 5 \mod 33 = 16 × 5 = 80 \mod 33 = 14 – **密文 / Шифротекст C = 14**

**解密 / Расшифрование（用私钥 d=3, n=33）：** – M = C^d \mod n = 14^3 \mod 33 – 14^2 = 196 \mod 33 = 31, 14^3 = 14 × 31 = 434 \mod 33 = 5 – **明文 / Открытый текст M = 5** ✓

## 例题3：数字签名 / Задача 3: Цифровая подпись

**题目 / Условие：** Alice想对消息 M = 12 进行签名（私钥 d=3, n=33），Bob用公钥(e=7, n=33)验证。

**Alice生成签名 / Алиса создаёт подпись：** – S = M^d \mod n = 12^3 \mod 33 = 12 → **签名 S = 12**

**Bob验证签名 / Боб проверяет подпись：** – M' = S^e \mod n = 12^7 \mod 33 = 12 → **M’ = M**，签名有效！ ✓

---

# Diffie–Hellman 密钥交换 / Обмен ключами Диффи–Хеллмана

## 参数与计算示例 / Параметры и пример вычислений

| 参数 / Параметр | 值 / Значение | 说明 / Описание |
|---|---|---|
| p | 23 | 质数 / Простое число |
| g | 5 | 生成元 / Генератор |
| a | 6 | Alice的私钥 / Приватный ключ Алисы |
| b | 15 | Bob的私钥 / Приватный ключ Боба |

### Diffie–Hellman 密钥交换示意图 / Схема обмена ключами


Diffie–Hellman密钥交换示意图

## 计算步骤 / Шаги вычисления

**步骤1：Alice计算公钥 / Шаг 1: Алиса вычисляет публичный ключ** $A = g^a \mod p = 5^6 \mod 23 = 8$

**步骤2：Bob计算公钥 / Шаг 2: Боб вычисляет публичный ключ** $B = g^b \mod p = 5^{15} \mod 23 = 19$

**步骤3：交换公钥 / Шаг 3: Обмен публичными ключами** – Alice发送A = 8给Bob / Алиса отправляет A = 8 Бобу – Bob发送B = 19给Alice / Боб отправляет B = 19 Алисе

**步骤4：计算共享密钥 / Шаг 4: Вычисление общего ключа** – Alice: K = B^a \mod p = 19^6 \mod 23 = 2 – Bob: K = A^b \mod p = 8^{15} \mod 23 = 2

✅ **共享密钥K = 2，双方一致！ / Общий секретный ключ K = 2, совпадает!**

### 安全性分析 / Анализ безопасности

攻击者知道 p=23, g=5, A=8, B=19，但无法计算 K = g^{ab} \mod p，因为离散对数问题在数学上极其困难。

Злоумышленник знает p, g, A, B, но не может вычислить K, так как задача дискретного логарифмирования крайне сложна.

---

# TLS协议详解 / Протокол TLS

### TLS是什么? / Что такое TLS?

**TLS (Transport Layer Security)** – 传输层安全协议，是互联网上最重要的安全协议。

**Русский:** TLS – протокол безопасности транспортного уровня, самый важный протокол безопасности в интернете.

**主要用途 / Основное назначение:** HTTPS、邮件加密、VPN

### TLS的两个阶段 / Две стадии TLS

#### 阶段一：握手阶段 / Стадия 1: Рукопожатие (Handshake)

握手阶段做**4件事**：

| # | 中文 | Русский |
|---|---|---|
| 1 | **身份认证** – 服务器通过证书证明身份 | **Аутентификация** – сервер доказывает подлинность через сертификат |
| 2 | **协商算法和参数** – 选择加密算法和哈希函数 | **Согласование алгоритмов** – выбор алгоритмов шифрования и хеш-функций |
| 3 | **配置** – 根据协商结果进行配置 | **Настройка** – настройка в соответствии с согласованными параметрами |
| 4 | **生成主密钥** – 使用D-H或RSA交换密钥 | **Генерация мастер-ключа** – обмен ключами через D-H или RSA |

### 阶段二：记录阶段 / Стадия 2: Запись (Record)

**核心任务：将主密钥切分成6个工作密钥**

**Основная задача: нарезать мастер-ключ на 6 рабочих ключей**

| # | 密钥 / Ключ | 用途 / Назначение |
|---|---|---|
| 1 | 客户端MAC密钥 | 客户端消息完整性验证 |
| 2 | 服务器MAC密钥 | 服务器消息完整性验证 |
| 3 | 客户端加密密钥 | 加密客户端发送的数据 |
| 4 | 服务器加密密钥 | 加密服务器发送的数据 |
| 5 | 客户端IV | 客户端加密算法的初始化向量 |
| 6 | 服务器IV | 服务器加密算法的初始化向量 |

**公式 / Формула：** 6个密钥 = (MAC + 加密密钥 + IV) × 2（客户端和服务器各一套）

## TLS安全特性 / Свойства безопасности TLS

| 特性 / Свойство | 中文 | Русский |
|---|---|---|
| **机密性** | 通过加密保护数据不被窃听 | Защита от прослушивания через шифрование |
| **完整性** | 通过MAC确保数据没有被篡改 | Гарантия неизменности данных через MAC |
| **认证** | 通过证书验证服务器身份 | Проверка подлинности сервера через сертификат |

# Kerberos vs TLS vs D–H 对比 / Сравнение протоколов

## 协议类型和定位 / Тип и назначение

| 协议 | 类型 | 核心功能 |
| --- | --- | --- |
| Kerberos | 完整的网络认证协议 | 认证、授权、单点登录(SSO) |
| TLS | 端到端加密通信协议 | 加密通信、数据完整性、服务器认证 |
| D–H | 密钥交换算法 | 生成共享密钥（不提供认证！） |

## 架构模型 / Архитектурная модель

| 协议 | 架构 | 说明 |
| --- | --- | --- |
| Kerberos | 集中式 | 需要KDC（密钥分发中心） |
| TLS | 分布式 | 点对点，依赖CA证书系统 |
| D–H | 分布式 | 两方直接交换，不需第三方 |

## 认证方式 / Методы аутентификации

| 协议 | 认证方式 | 特点 |
| --- | --- | --- |
| Kerberos | 票据(Tickets) | 由KDC签发，有时效性 |
| TLS | 证书(X.509) | 由CA签发 |
| D–H | 不提供认证 | 易受中间人攻击！ |

## 加密方式 / Методы шифрования

| 协议 | 加密类型 | 说明 |
| --- | --- | --- |
| Kerberos | 纯对称加密 | 所有票据用对称密钥加密 |
| TLS | 混合加密 | 握手用非对称，传输用对称 |
| D–H | 密钥交换算法 | 不直接加密数据 |

# NAT（网络地址转换）/ NAT (Трансляция сетевых адресов)

## 什么是NAT? / Что такое NAT?

**NAT = Network Address Translation（网络地址转换）**

将内部私有IP地址转换为外部公共IP地址，使多个内部设备可以共享一个公共IP。

**Русский:** NAT преобразует частные IP-адреса во внешние публичные, позволяя множеству устройств использовать один публичный IP.

## NAT的主要功能 / Основные функции NAT

| 功能 | 中文 | Русский |
| --- | --- | --- |
| **地址转换** | 私有地址 ? 公共地址 | Частный адрес ? Публичный адрес |
| **节省IP** | 多设备共享一个公网IP | Множество устройств на одном публичном IP |
| **隐藏内网** | 外部无法直接访问内部 | Внешние не могут напрямую достучаться до внутренних |

## NAT类型 / Типы NAT

| 类型 | 中文 | Русский |
| --- | --- | --- |
| **静态NAT** | 一对一固定映射 | Статическое отображение один-к-одному |
| **动态NAT** | 从地址池动态分配 | Динамическое назначение из пула адресов |
| **NAPT/PAT** | 地址+端口转换（最常用） | Преобразование адреса и порта (наиболее распространённый) |

### NAT的优缺点 / Преимущества и недостатки

| 优点 / Преимущества | 缺点 / Недостатки |
| --- | --- |
| 节省公网IP地址 | 端到端连接被破坏 |
| 隐藏内部网络结构 | 某些协议无法穿透NAT |
| 增加一层安全性 | 增加网络复杂性 |

# 防火墙技术 / Технологии межсетевых экранов

## 防火墙的基本概念 / Основные понятия

**防火墙 (Межсетевой экран, МЭ)** – 根据安全策略允许或拒绝网络流量的设备。

**Русский:** Устройство, которое разрешает или запрещает сетевой трафик согласно политикам безопасности.

## 防火墙技术分类 / Классификация технологий

| 类型 | 中文 | Русский |
| --- | --- | --- |
| **包过滤器** | 检查IP/端口，分为有状态和无状态 | Проверка IP/портов, с состоянием и без |
| **应用层防火墙** | 分析应用协议内容 | Анализ содержимого прикладных протоколов |
| **代理防火墙** | 作为中间人转发流量 | Пересылка трафика как посредник |

## 防火墙策略 / Политики МЭ

**核心原则：默认拒绝所有 (Deny by default)**

**Русский:** Основной принцип: запрещать всё по умолчанию, разрешать только необходимое.

### DMZ（隔离区）/ DMZ

**DMZ = Demilitarized Zone（非军事区/隔离区）**

内外网之间的中间区域，存放对外开放的服务器，防止黑客直接攻击内网。

**Русский:** Промежуточная зона между внутренней и внешней сетью для публичных серверов.

---

## 完整性验证 / Проверка целостности

### 什么是完整性? / Что такое целостность?

**完整性 (Целостность)** – 确保数据在传输或存储过程中没有被修改、删除或损坏。

**Русский:** Гарантия, что данные не были изменены, удалены или повреждены.

### 验证完整性的方法 / Методы проверки

| 方法 | 中文 | Русский | 特点 |
| --- | --- | --- | --- |
| **哈希函数** | 计算固定长度摘要 | Вычисление дайджеста фиксированной длины | 不需要密钥 |
| **MAC** | 带密钥的消息认证码 | Код аутентификации с ключом | 需要共享密钥 |
| **数字签名** | 用私钥签名 | Подпись приватным ключом | 提供不可否认性 |

### HMAC公式 / Формула HMAC

HMAC = H((K \oplus opad) \| H((K \oplus ipad) \| M))

---

# 哈希函数要求与攻击 / Требования к хеш-функциям и атаки

## 安全哈希函数的要求 / Требования

| 要求 | 中文 | Русский |
|---|---|---|
| **单向性** | 从哈希值无法反推原消息 | Невозможно восстановить сообщение из хеша |
| **抗弱碰撞** | 给定M，难找M'使H(M)=H(M') | Сложно найти M' с тем же хешем для данного M |
| **抗强碰撞** | 难找任意两个M≠M'使H(M)=H(M') | Сложно найти любые два сообщения с одинаковым хешем |
| **雪崩效应** | 输入变1位，输出变约50% | Изменение 1 бита входа меняет ~50% битов выхода |

## 生日攻击 / Атака дня рождения

**原理 / Принцип：** 基于生日悖论 – 23人中有50%概率两人同生日

**公式 / Формула：** 对于n位哈希，只需约 2^{n/2} 次尝试就能找到碰撞

| 哈希长度 | 暴力破解 | 生日攻击 |
|---|---|---|
| 128 bit | 2^{128} | 2^{64} |
| 256 bit | 2^{256} | 2^{128} |

## SHA家族对比 / Сравнение семейства SHA

| 算法 | 输出长度 | 安全性 |
|---|---|---|
| SHA–1 | 160 bit | ❌ 已被破解 |
| SHA–256 | 256 bit | ✅ 安全 |
| SHA–3 | 可变 | ✅ 最新标准 |

# 重要公式总结 / Важные формулы

## RSA

| 操作 | 公式 |
| --- | --- |
| 模数 | $n = p \times q$ |
| 欧拉函数 | $\varphi(n) = (p-1)(q-1)$ |
| 加密 | $C = M^e \mod n$ |
| 解密 | $M = C^d \mod n$ |
| 密钥关系 | $d \times e \equiv 1 \pmod{\varphi(n)}$ |

## Diffie–Hellman

| 操作 | 公式 |
| --- | --- |
| Alice公钥 | $A = g^a \mod p$ |
| Bob公钥 | $B = g^b \mod p$ |
| 共享密钥 | $K = g^{ab} \mod p$ |

---

祝考试顺利！📚🎓 Удачи на экзамене!