МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

ЛАБОРАТОРНЫЙ ПРАКТИКУМ

по дисциплине

«Информационная безопасность и защита данных»

Методические указания к выполнению лабораторных работ для студентов направления 15.04.04 «Автоматизация технологических процессов и производств»

Ставрополь 2023

Содержание

Введение

Целью изучения дисциплины является формирование у студента профессиональных компетенций по направлению подготовки 15.04.04 Автоматизация технологических процессов и производств, а также приобретение теоретических знаний и практических навыков по использованию средств защиты для обеспечения информационной безопасности и защиты информации от несанкционированного использования ресурсов АСУ ТП..

Задачи изучения дисциплины заключаются:

-приобретении студентами знаний и практических навыков в области, определяемой основной целью дисциплины;

-приобретение практических навыков работы с алгоритмами защиты информации.

Код, формулировка компетенции	Код, формулировка индикатора	Планируемые результаты обучения по дисциплине (модулю), характеризующие этапы формирования компетенций, индикаторов
ПК-3. Способен собирать и анализировать исходные данные для проектирования средств и систем автоматизации	ИД-1 $_{ПК-3}$. Собирает и анализирует исходные данные для проектирования средств и систем автоматизации	Производит сбор и анализ исходных данных для проектирования средств и систем автоматизации с встроенными механизмами обеспечения информационной безопасности и защиты данных.
	ИД-2 $_{ПК-3}$. Оформляет техническое задание и обосновывает его для заказчика	Производит оформление технического задания для проектирования средств и систем автоматизации с встроенными механизмами обеспечения информационной безопасности и защиты данных
	ИД-3 $_{ПК-3}$. Использует современные информационные технологии для сбора и анализа исходных данных для проектирования средств и систем автоматизации	Имеет практический опыт использования современных информационных технологий для сбора и анализа исходных данных для проектирования средств и систем автоматизации

Лабораторная работа №1

Пороговое разделение секрета

Цель работы: исследовать возможности порогового разделения секрета.

Краткие сведения из теории

Разработка схем пространственного разделения секрета первоначально была направлена на сохранение секретного ключа от потери. На данном этапе развития информационных технологий область применения схем разделения секрета расширилась на пороговое разделение информации, динамическое распределение нагрузки передачи данных между компьютерами в сети, закрытую передачу данных, электронное голосование.

1. Пороговая схема разделения секрета Миньотта. Пороговая схема разделения секрета Миньотта использует специальные последовательности целых чисел, названных последовательностями Миньотта.

Пусть n будет целым числом, $n \geq 2$, и $2 \leq k \leq n$. (k, n)-последовательность Миньотта – последовательность положительных целых чисел $m_1 < ... < m_n$ таких, что $\left(m_i, m_j\right) = 1$, для всех $1 \leq i < j \leq n$, и $m_{n-k+2} \cdots m_n < m_1 \cdots m_k$.

С заданной (k, n)-последовательностью Миньотта схема работает следующим образом:

- секрет S выбран как случайное целое число такое, что $\beta < S < \alpha$, где $\alpha = m_1 \cdots m_k$ и $\beta = m_{n-k+2} \cdots m_n$;

- части секрета $I_i = S \bmod m_i$, для всех $1 \leq i \leq n$;

- дано k различных частей $I_{i_1}, ..., I_{i_k}$, секрет S восстанавливается с использованием стандартной Китайской теоремы об остатках, как уникальное решение системы по модулю $m_{i_1}, ..., m_{i_k}$

$$\begin{cases} x \equiv I_{i_1} \bmod m_{i_1} \\ \vdots \\ x \equiv I_{i_k} \bmod m_{i_k} \end{cases}$$

2. Обобщенная схема Миньотта позволяет использовать модули, которые не обязательно попарно взаимно простые.

Пусть *n* будет целым числом, $n \geq 2$, и $2 \leq k \leq n$. Обобщенная (*k*, *n*)-последовательность Миньотта – последовательность $m_1,...,m_n$ положительных целых чисел таких, что $\max_{1\leq i_1<...<i_{k-1}\leq n}\left(\left[\left\{i_1,...,i_{k-1}\right\}\right]\right) < \min_{1\leq i_1<...<i_k\leq n}\left(\left[\left\{i_1,...,i_k\right\}\right]\right)$.

Просто увидеть, что каждая (*k*, *n*)-последовательность Миньотта является обобщенной (*k*, *n*)-последовательностью Миньотта. Кроме того, если мы умножаем каждый элемент (*k*, *n*)-последовательности Миньотта на фиксированный элемент $\delta \in Z$, $\left(\delta, m_1,...,m_n\right)=1$ мы получаем обобщенную (*k*, *n*)-последовательность Миньотта. Обобщенная схема Миньотта работает подобно схеме Миньотта, с $\alpha = \min_{1\leq i_1<...<i_k\leq n}\left(\left[\left\{i_1,...,i_k\right\}\right]\right)$ и $\beta = \max_{1\leq i_1<...<i_{k-1}\leq n}\left(\left[\left\{i_1,...,i_{k-1}\right\}\right]\right)$.

3. Пороговая схема разделения секрета Асмута-Блума. Эта схема, предложенная Асмутом и Блумом, использует специальные последовательности целых чисел: последовательность попарно взаимно простых положительных целых чисел $r, m_1 < ... < m_n$, выбранных так, что $r \cdot m_{n-k+2} \cdots m_n < m_1 \cdots m_k$.

С заданной последовательностью, схема работает следующим образом:

- тайна *S* выбирается как случайный элемент множества Z_r;

- части $I_i = \left(S + \gamma \cdot r\right) \bmod m_i$, для всех $1 \leq i \leq n$, где γ – произвольное целое число такое, что $S + \gamma \cdot r \in Z_{m_1 \ldots m_k}$;

- дано *k* отличных частей $I_{i_1},...,I_{i_k}$, секрет *S* может быть получен как $S = x_0 \bmod r$, где x_0 получен с использованием стандартной Китайской теоремы об остатках, как уникальный решение системы по модулю $m_{i_1},...,m_{i_k}$

$$\begin{cases} x \equiv I_{i_1} \bmod m_{i_1} \\ \quad\vdots \\ x \equiv I_{i_k} \bmod m_{i_k} \end{cases}$$

4. Обобщенная схема Асмута-Блума допускает модули, которые не обязательно попарно взаимно-простые. Можно использовать любую последовательность $r, m_1 < ... < m_n$ такую, что $r \cdot \max_{1\leq i_1<...<i_{k-1}\leq n}\left(\left[\left\{i_1,...,i_{k-1}\right\}\right]\right) < \min_{1\leq i_1<...<i_{k-1}\leq n}\left(\left[\left\{i_1,...,i_k\right\}\right]\right)$.

Просто увидеть, что, если мы умножаем каждый элемент обычной последовательности Асмута-Блума за исключением r на фиксированный элемент $\delta \in Z$, $(\delta, m_1, ..., m_n) = 1$, мы получаем обобщенную последовательность Асмута-Блума.

5. Схема Ито-Саито-Нишизеки. Ито, Саито, и Нишизеки ввели так называемую методику совокупного массива для монотонных структур доступа.

Пусть A будет монотонной структурой санкционированного доступа размера n и пусть $B_1, ..., B_m$ будут соответствующими максимальными множествами несанкционированного доступа. Совокупный массив для структуры доступа A, обозначенной C^A, является $n \times m$ матрица, $C^A = \left(C^A_{i,j}\right)_{\substack{1 \le i \le n \\ 1 \le j \le m}}$, где $C^A_{i,j} = \begin{cases} 0, \text{ если } i \in B_j \\ 1, \text{ если } i \notin B_j \end{cases}$, для всех $1 \le i \le n$ и $1 \le j \le m$. Давайте рассмотрим теперь произвольную (m, m)-схему порогового разделения секрета с секретом S и соответствующими тенями $s_1, ..., s_m$. В A-схеме разделения секрета, тени $I_1, ..., I_n$ соответствующие секрету S будут определены как $I_i = \left\{ s_j \mid C^A_{i,j} = 1 \right\}$ для всех $1 \le i \le n$ и $1 \le j \le m$.

Пример. Пусть $n = 4$ и $A_{\min} = \{\{1, 2\}, \{3, 4\}\}$. В этом случае, мы получим, что $\bar{A}_{\max} = \{\{1, 3\}, \{1, 4\}, \{2, 3\}, \{2,4\}\}$ и $m = 4$. Совокупный массив для структуры доступа A

$$C^A = \begin{pmatrix} 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix}$$

В этом случае $I_1 = \{s_3, s_4\}$, $I_2 = \{s_1, s_2\}$, $I_3 = \{s_2, s_4\}$ и $I_4 = \{s_1, s_3\}$, где s_1, s_2, s_3, s_4 – тени (4, 4)-схемы порогового разделения секрета S.

6. Схема Бенало-Лихтера. Бенало и Лихтер представили структуры доступа, используя формулы. Более точно, для монотонной структуры санкционированного доступа A размера n, они определили множество F_A как множество формул на множестве переменных $\{v_1, v_2..., v_n\}$ такое, что для каждого $\mathrm{F} \in F^A$ интерпретация F относительно присваивания переменных является истиной, если и только если истинные переменные соответствуют множеству $\mathrm{A} \in A$. Они отметили, что такие формулы могут

использоваться как шаблоны для того, чтобы описать, как секрет может быть разделен относительно данной структуры доступа. Поскольку формулы могут быть выражены, используя только операторы $\wedge$ и $\vee$, достаточно указать, как "разбить" секрет поперек этих операторов. Таким образом, мы можем определить части секрета S относительно формул F следующим образом

$$Части(S,\mathrm{F}) = \begin{cases} (S,i), & \text{если } \mathrm{F} = v_i, 1 \le i \le n; \\ \bigcup_{i=1}^{k} Части(S,\mathrm{F}_i), & \text{если } \mathrm{F} = \mathrm{F}_1 \vee \mathrm{F}_2 \vee \ldots \vee \mathrm{F}_k; \\ \bigcup_{i=1}^{k} Части(s_i,\mathrm{F}_i), & \text{если } \mathrm{F} = \mathrm{F}_1 \wedge \mathrm{F}_2 \wedge \ldots \wedge \mathrm{F}_k, \end{cases}$$

где, для случая $\mathrm{F} = \mathrm{F}_1 \wedge \mathrm{F}_2 \wedge \ldots \wedge \mathrm{F}_k$, мы можем использовать любую (k, k)-пороговую схему разделения секрета для получения некоторых частей $s_1,\ldots, s_k$ соответствующих секрету S и, наконец, части как $I_i = \{s \mid (s,i) \in Части(S,\mathrm{F})\}$, для всех $1 \le i \le n$, где F – произвольная формула из множества F_A.

Пример. Пусть $n = 3$ и структура санкционированного доступа A задана $A_{\min} = \{\{1, 2\}, \{2, 3\}\}$. Например, формула $\mathrm{F} = (v_1 \wedge v_2) \vee (v_2 \wedge v_3)$ находится во множестве F_A. В этом случае $Части(S,\mathrm{F})$, для некоторого секрета S, может быть получена, как

$$Части(S,\mathrm{F}) = Части(S, v_1 \wedge v_2) \bigcup Части(S, v_2 \wedge v_3) =$$
$$= Части(s_1, v_1) \bigcup Части(s_{2,1}, v_2) \bigcup Части(s_{2,2}, v_2) \bigcup Части(s_3, v_3) =$$
$$= \{(s_1,1),(s_{2,1},2),(s_{2,2},2),(s_3,3)\},$$

где s_1, $s_{2,1}$ и, соответственно, $s_{2,2}$, s_3 – тени секрета S относительно двух произвольных (2, 2)-пороговых схем. Таким образом, части, соответствующие секрету S относительно структуры доступа A – $I_1 = \{s_1\}$, $I_2 = \{s_{2,1}, s_{2,2}\}$ и $I_2 = \{s_3\}$.

Задание на лабораторную работу

Исследовать пороговую схему разделения секрета в соответствии с вариантом (таблица 1). Модули выбрать из Приложения 1.

Таблица 1 – Варианты для лабораторной работы № 1

Вариант	**Пороговая схема**	**Тип схемы**
1	Пороговая схема разделения секрета Миньотта	3-из-5
2	Обобщенная схема Миньотта	3-из-5
3	Пороговая схема разделения секрета Асмута-Блума	3-из-5
4	Обобщенная схема Асмута-Блума	3-из-5
5	Схема Ито-Саито-Нишизеки	3-из-5
6	Схема Бенало-Лих-тера	3-из-5
7	Пороговая схема разделения секрета Миньотта	2-из-3
8	Обобщенная схема Миньотта	2-из-3
9	Пороговая схема разделения секрета Асмута-Блума	2-из-3
10	Обобщенная схема Асмута-Блума	2-из-3
11	Схема Ито-Саито-Нишизеки	2-из-3
12	Схема Бенало-Лих-тера	2-из-3

Содержание отчета

В отчете указать цель работы, представить результаты исследований.

Контрольные вопросы

1. Назначение пороговой схемы.
2. Сравните энтропию частного ключа для схем Миньотта и Асмут-Блума.
3. Китайская теорема об остатках. Численный пример.
4. За счет чего пороговая схема обеспечивает надежное хранение секретной информации.

Лабораторная работа №2

Криптография на базе эллиптических кривых

Цель работы: исследовать криптографические примитивы схем защиты данных на базе алгебры точек эллиптической кривой

Краткие сведения из теории

Операции над точками эллиптической кривой определяются над полем Галуа GF(p), где p – простое число (p > 3). Все арифметические операции выполняются по модулю p. Эллиптическая кривая E задается выражением

$$y^2 = x^3 + ax + b,$$

где $4a^2 + 27b^2 \neq 0$ и $x, y, a, b \in GF(p)$. Также существует один элемент, называемый бесконечной точкой и обозначаемый символом «O», который выступает в роли аддитивной единицы: $\forall P(x, y) \in E : P + O = P$.

Для сложения точек эллиптической кривой существуют следующие правила:

- $O = -O$,
- $P(x, y) + O = P(x, y)$,
- $P(x, y) + P(x, -y) = O$.

Операция сложения двух произвольных точек эллиптической кривой выполняется следующим образом:

$$P(x_1, y_1) + P(x_2, y_2) = P(x_3, y_3),$$

где $x_3 = \lambda^2 - x_1 - x_2$; $y_3 = \lambda(x_1 - x_3) - y_1$; $\lambda = (y_2 - y_1)/(x_1 - x_2)$ – угловой коэффициент касательной к эллиптической кривой, используемой для определения результата сложения.

Операция удвоения точки:

$$P(x_1, y_1) + P(x_1, y_1) = P(x_3, y_3),$$

где $x_3 = \lambda^2 - 2x_1$; $y_3 = \lambda(x_1 - x_3) - y_1$; $\lambda = \left(3(x_1)^2 + a\right)/(2y_1)$.

Умножение точки $P(x, y) \in E$ на скаляр k над GF(p) определяется серией сложений:

$$Q = [k]P = \underbrace{P + P + \ldots + P}_{k \text{ раз}}.$$

Умножение точки на скаляр лежит в основе криптографических алгоритмов на базе группы точек эллиптической кривой.

При разработке криптографических алгоритмов на эллиптической кривой иногда требуется сопоставить данному открытому тексту некоторую точку кривой, при этом обратное отображение должно позволять однозначно определить открытый текст. Один из способов – для фиксированного уравнения кривой $y^2 = f(x)$ с размером задачи *l* бит над полем *K* в качестве *x*-координаты использовать $(n-k)$-разрядный текст, дополненный *k* двоичными разрядами так, что число $f(x)$ – квадратный вычет в поле *K*. Варьируя число *k*, можно изменять скорость шифрования и вероятность успешного шифрования текста. Такой способ целесообразен для кривых, над простыми полями, поскольку для нахождения «добавки» к открытому тексту нужно вычислять значения квадратичного характера, а эта операция удобно выполняется в простом поле – достаточно найти символ Якоби.

Задание на лабораторную работу

В соответствии с вариантом выбрать параметры эллиптической кривой из Приложения 2. Исследовать процесс встраивания открытого текста (Приложение 3) в координату *x* точки заданной эллиптической кривой.

Содержание отчета

В отчете указать цель работы, представить результаты исследования.

Контрольные вопросы

1. Вычисление символа Якоби.
2. Извлечение квадратных и кубических корней в конечном поле.
3. Умножение точки эллиптической кривой на число. Численный пример.
4. Проективные координаты.
5. Эллиптические кривые над конечным полем и над расширением конечного поля.

Лабораторная работа №3

Электронная цифровая подпись ГОСТ Р 34.10-2018

Цель работы: исследовать процесс формирования и проверки электронной цифровой подписи по ГОСТ Р 34.10-2018

Краткие сведения из теории

Общепризнанная схема (модель) цифровой подписи (см. ИСО/МЭК 14888–1) охватывает следующие процессы:

– генерация ключей (подписи и проверки подписи);

– формирование подписи;

– проверка подписи. В настоящем стандарте процесс генерации ключей (подписи и проверки подписи) не рассмотрен. Характеристики и способы реализации данного процесса определяются вовлеченными в него субъектами, которые устанавливают соответствующие параметры по взаимному согласованию. Механизм цифровой подписи определяется посредством реализации двух основных процессов:

– формирование подписи;

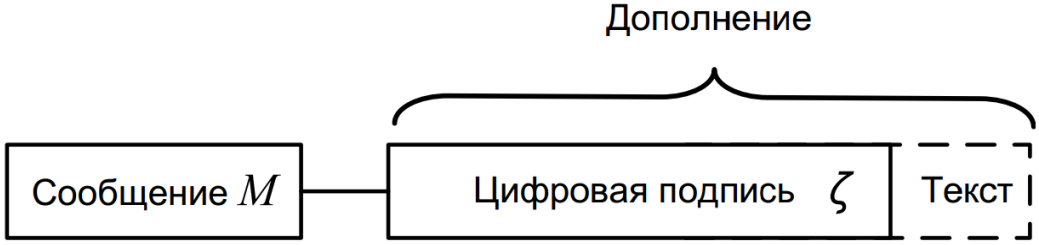
– проверка подписи.

Цифровая подпись предназначена для аутентификации лица, подписавшего электронное сообщение. Кроме того, использование ЭЦП предоставляет возможность обеспечить следующие свойства при передаче в системе подписанного сообщения:

– осуществление контроля целостности передаваемого подписанного сообщения,

– доказательное подтверждение авторства лица, подписавшего сообщение,

– защита сообщения от возможной подделки.



Риснуок 1 – Схема подписанного сообщения

Поле «Текст», показанное на рисунке 1 и дополняющее поле «Цифровая подпись», может, например, содержать идентификаторы субъекта, подписавшего сообщение, и/или метку времени.

Установленная в ГОСТ Р 34.10-2018 схема цифровой подписи реализована с использованием операций группы точек эллиптической кривой, определённой над конечным простым полем, а также хэш-функции. Криптографическая стойкость данной схемы цифровой подписи основывается на сложности решения задачи дискретного логарифмирования в группе точек эллиптической кривой, а также на стойкости используемой хэш-функции. Алгоритмы вычисления хэш-функции установлены в ГОСТ Р 34.11-2018.

Цифровая подпись, представленная в виде двоичного вектора длиной 512 или 1024 бита.

Задание на лабораторную работу

Исследовать процесс формирования ЭЦП для заданного варианта текста:

а) выполнить процесс формирования ЭЦП для введенного текста, получить результаты по всем этапам формирования ЭЦП на экране дисплея;

б) выполнить процесс проверки соответствия ЭЦП введенному тексту, получить результаты по всем этапам проверки ЭЦП на экране дисплея;

в) внести изменения в исходный текст, выполнить процесс проверки ЭЦП для модифицированного текста, убедиться в несоответствии ЭЦП модифицированному тексту;

г) повторить пункты а–в для текстов различной длины и новых исходных числовых значений для формирования ЭЦП;

Провести сеанс формирования ЭЦП и проверки ее подлинности по следующему протоколу:

а) студент А – подписант электронного документа, самостоятельно вычисляет ключи для формирования и проверки подлинности ЭЦП;

б) студент А передает по открытому каналу связи (публично объявляет, передает по локальной сети, копирует на флэш-диск) ключ проверки подлинности ЭЦП студенту В. Секретный ключ для формирования ЭЦП сохраняется им в тайне от других студентов, выполняющих лабораторную работу;

в) студент А составляет электронный документ $m \in M$, вычисляет для него хэш-значение и ЭЦП;

г) студент А передает по открытому каналу связи (по локальной сети, копирует на дискету) в адрес студента В электронный документ с ЭЦП;

д) студент В – получатель электронного документа, защищенного ЭЦП, используя известную функцию хэширования и открытый ключ ЭЦП, проверяет ее подлинность и убеждается в целостности полученного электронного документа;

е) студент С – выполняя роль "злоумышленника", на этапе передачи электронного документа с ЭЦП от студента А к студенту В осуществляет его модификацию и/или подделку ЭЦП посредством подбора секретного ключа для ее вычисления. Полученный электронный документ с ЭЦП передается студенту В;

ж) студент В, осуществляя проверку ЭЦП, убеждается в ее несоответствии полученному электронному документу. Результаты исследования объявляются студентам А и С.

Содержание отчета

В отчете указать цель работы, схему функции хэширования с параметрами, соответствующими варианту индивидуального задания, схемы алгоритмов исследованных функции хэширования и ЭЦП, результаты, полученные при апробировании процессов хэширования и формирования ЭЦП, анализ полученных результатов и выводы по лабораторной работе.

Контрольные вопросы

1. Для каких целей служит функция хэширования и какими основными свойствами она обладает?
2. В чем заключаются методы реализации функции хэширования, основанные на алгоритмах симметричного блочного шифрования?
3. В чем заключаются методы реализации функции хэширования, основанные на числовых корректирующих кодах?
4. На каких математических принципах и задачах (проблемах) основана ЭЦП по алгоритму ГОСТ Р 34.10–2018?
5. Как задается эллиптическая кривая над простым полем?
6. Что называется инвариантом эллиптической кривой?

7. Как определяются коэффициенты эллиптической кривой?

8. Какую длину имеет хэш-значение, полученное по ГОСТ Р 34.10–2018?

9. Как реализуется функция хэширования по ГОСТ Р 34.11–2018? В чем ее преимущество по сравнению с другими типами хэш-функций?

10. Какие длины ключей (исходных значений) рекомендованы для использования на практике при формировании ЭЦП ГОСТ Р 34.10–2018?

11. Каким условиям должны удовлетворять параметры ЭЦП по ГОСТ Р 34.10–2018?

12. Какие отличительные особенности, достоинства и недостатки характерны для ЭЦП по ГОСТ Р 34.10–2018?

Лабораторная работа №4

Парольная защита

Цель работы: исследовать меры противодействия угрозам парольной защиты.

Краткие сведения из теории

Наиболее распространенные методы аутентификации основаны на применении многоразовых или одноразовых паролей. Из-за своего широкого распространения и простоты реализации парольные схемы часто в первую очередь становятся мишенью атак злоумышленников. Эти методы включают следующие разновидности способов аутентификации: по хранимой копии пароля или его свёртке (plaintext-equivalent); по некоторому проверочному значению (verifier - based); без непосредственной передачи информации о пароле проверяющей стороне (zero - knowledge); с использованием пароля для получения криптографического ключа (cryptographic)

В первую разновидность способов входят системы аутентификации, предполагающие наличие у обеих сторон копии пароля или его свертки. Для организации таких систем требуется создать и поддерживать базу данных, содержащую пароли или сверки паролей всех пользователей. Их слабой стороной является то, что получение злоумышленником этой базы данных позволяет ему проходить аутентификацию от имени любого пользователя. Способы, составляющие вторую разновидность, обеспечивают более высокую степень безопасности парольной системы, так как проверочные значения, хотя они и зависят от паролей, не могут быть непосредственно использованы злоумышленником для аутентификации. Наконец, аутентификация без предоставления проверяющей стороне какой бы то ни было информации о пароле обеспечивает наибольшую степень защиты. Этот способ гарантирует безопасность даже в том случае, если нарушена работа проверяющей стороны (например, в программу регистрации в системе внедрен "троянский конь"). Особым подходом в технологии проверки подлинности являются криптографические протоколы аутентификации. Такие протоколы описывают последовательность действий, которую должны совершить стороны для взаимной аутентификации, кроме того, эти действия, как правило, сочетаются с генерацией и распределением криптографических ключей для шифрования последующего информационного обмена. Корректность протоколов аутенти-

фикации вытекает из свойств задействованных в них математических и криптографических преобразований и может быть строго доказана.

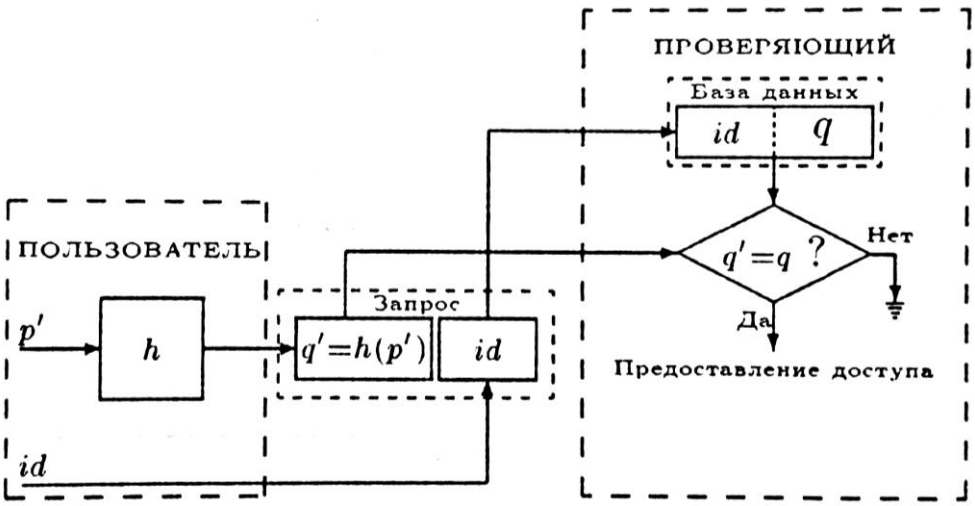


Рисунок 4.1 – Основной вариант схемы защищенных паролей

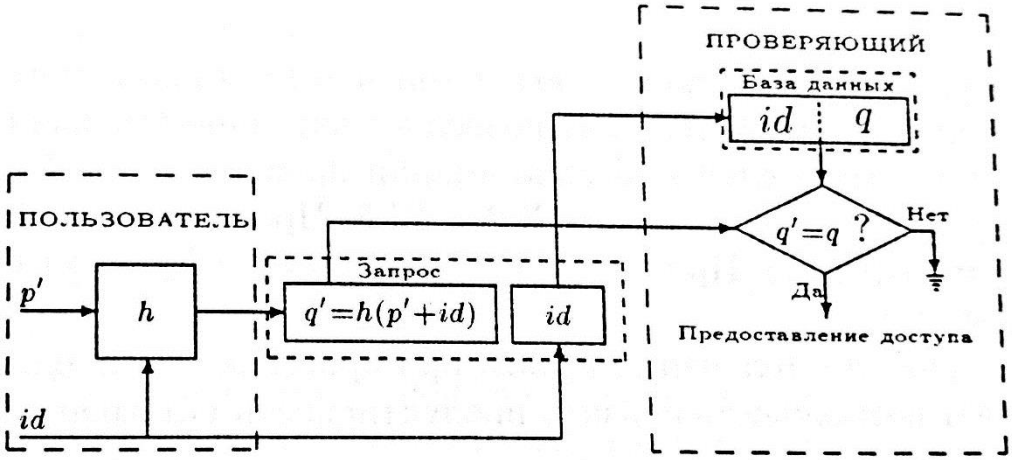


Рисунок 4.2 – Вариант схемы защищенных паролей, устойчивый к табличной атаке

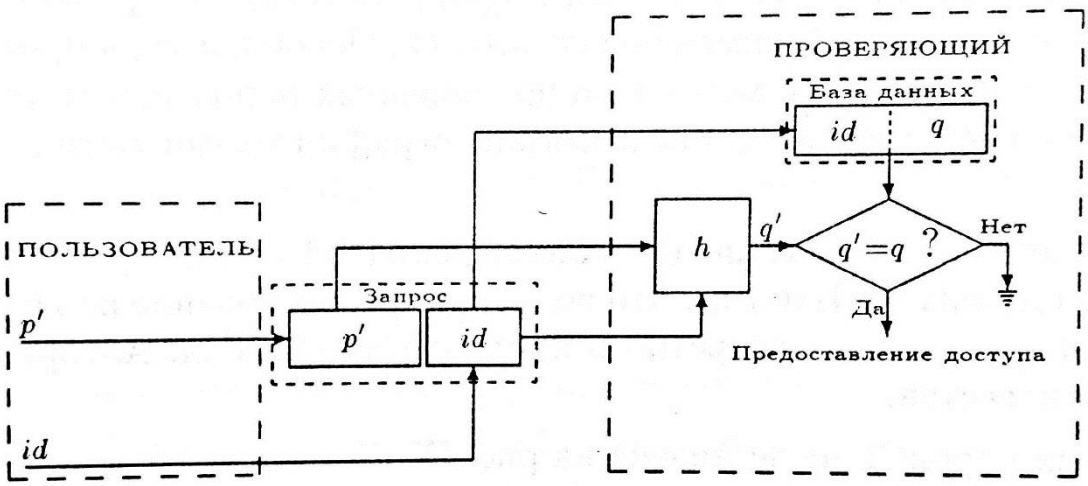


Рисунок 4.3 – Основной вариант парольной защиты при компрометации проверяющего

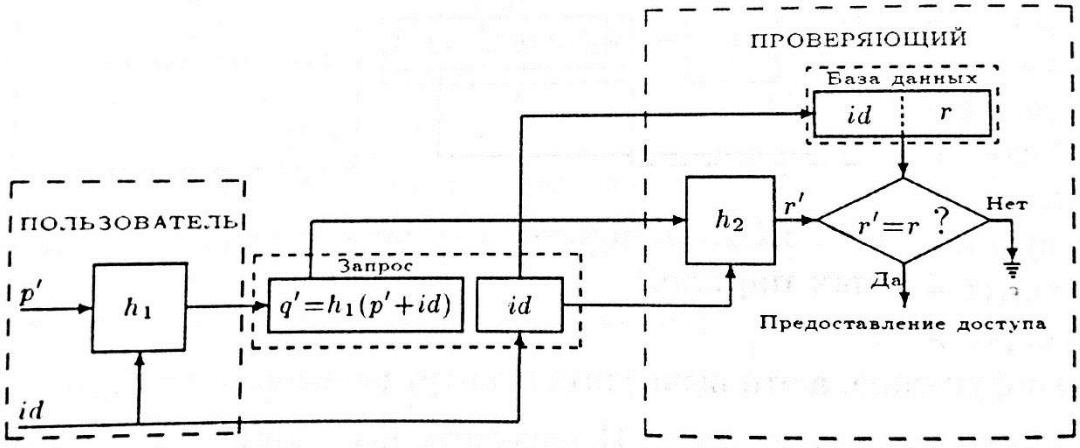


Рисунок 4.4 – Усовершенствованный вариант схемы парольной защиты при компрометации проверяющего

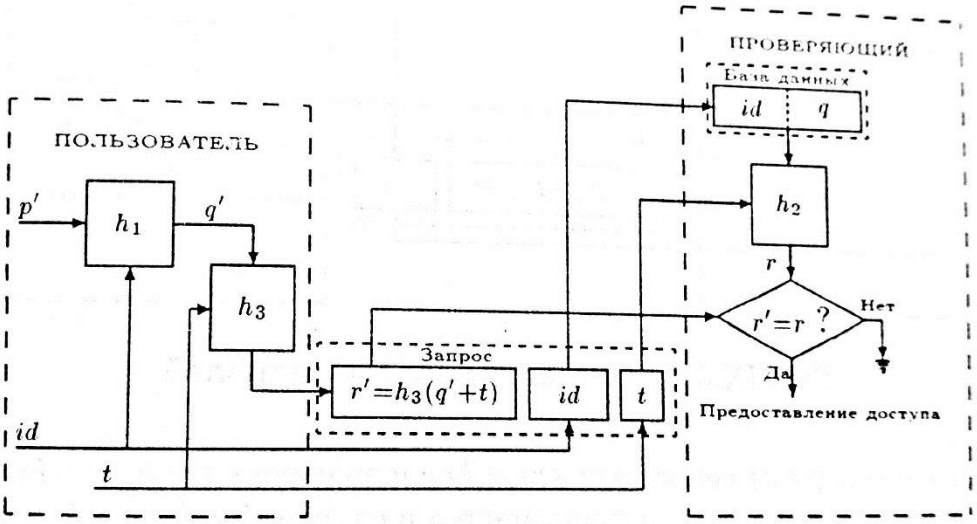


Рисунок 4.5 – Защита от несанкционированного воспроизведения

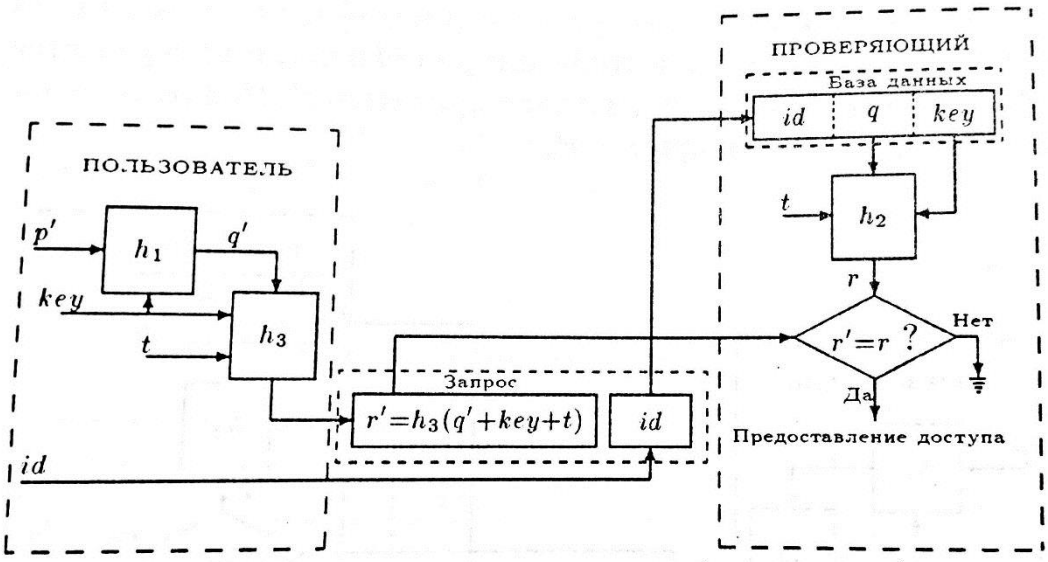


Рисунок 4.6 – Схема одноразовых паролей

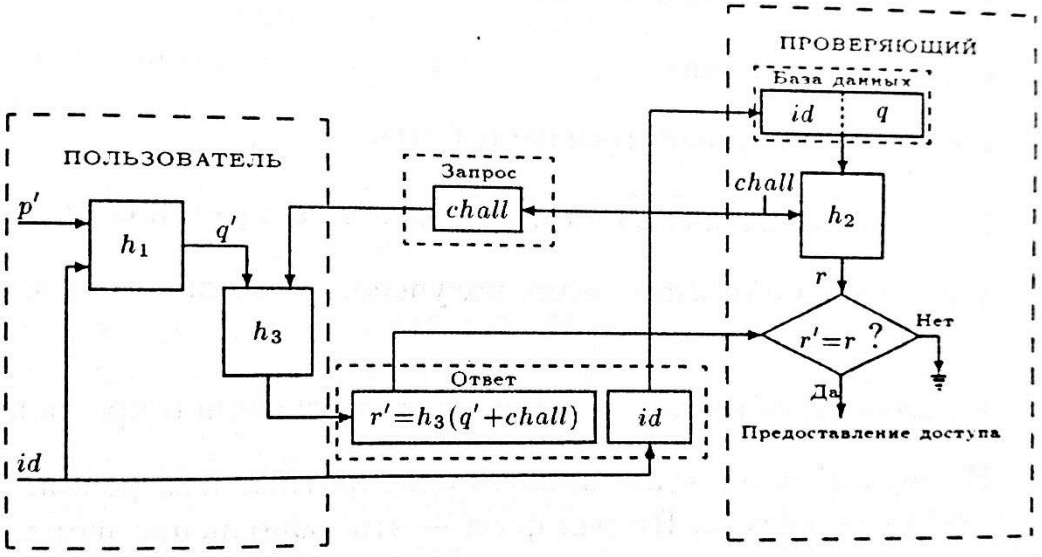


Рисунок 4.7 – Схема аутентификации по методу «запрос-ответ»

Задание на лабораторную работу

Исследовать механизм парольной защиты.

Вариант	**Схема**
1	Рисунок 4.1
2	Рисунок 4.2
3	Рисунок 4.3
4	Рисунок 4.4
5	Рисунок 4.5
6	Рисунок 4.6
7	Рисунок 4.7
8	plaintext-equivalent
9	verifier-based
10	zero-knowledge
11	cryptographic
12	One-Time Password System (N. Haller, C. Metz)

Содержание отчета

В отчете указать цель работы, представить результаты исследований.

Контрольные вопросы

1. Основной вариант схемы защищенных паролей.

2. Вариант схемы защищенных паролей, устойчивый к табличной атаке.
3. Основной вариант парольной защиты при компрометации проверяющего.
4. Усовершенствованный вариант схемы парольной защиты при компрометации проверяющего.
5. Защита от несанкционированного воспроизведения.
6. Схема одноразовых паролей.
7. Схема аутентификации по методу «запрос-ответ».

Лабораторная работа № 5

Криптографические методы аутентификации

Цель работы: исследовать режимы аутентификации.

Краткие сведения из теории

Многие системы аутентификации используют для самой аутентификации или представления контекста доступа алгоритм шифрования с открытым ключом RSA. Способы аутентификации, основанные на RSA, сводятся к следующему алгоритму.

- Система А генерирует последовательность байтов, обычно случайную, кодирует ее своим ключом и посылает системе В.
- Система В раскодирует ее своим ключом. Это возможно, только если системы владеют парными ключами.
- Системы тем или иным способом обмениваются "правильными" значениями зашифрованной посылки.

Для примера рассмотрим принцип RSA-аутентификации в пакете ssh – Secure Shell. Пакет представляет собой функциональную замену программ rlogin/rsh и соответствующего этим программам демона rshd. В пакет входят программы ssh (клиент) и sshd (сервер), а также утилиты для генерации ключей RSA и управления ими. ssh использует RSA для прозрачной аутентификации пользователя при входе в удаленную систему. Кроме того, ssh/sshd могут осуществлять шифрование данных, передаваемых по линии во время сеанса связи и выполнять ряд других полезных функций.

Когда из удаленной системы-клиента приходит запрос на аутентификацию, sshd запрашивает публичный ключ. Если полученный ключ совпадает с хранящимся в файле значением для этой системы, сервер генерирует случайную последовательность из 256 бит, шифрует ее публичным ключом и посылает клиенту. Клиент расшифровывает посылку своим личным ключом, вычисляет 128-битовую контрольную сумму и возвращают ее серверу. Сервер сравнивает полученную последовательность с правильной контрольной суммой и принимает аутентификацию в случае совпадения. Теоретически контрольные суммы могут совпасть и в случае несовпадения ключей, но вероятность такого события крайне мала.

Система групповой работы Lotus Notes также использует для аутентификации открытый ключ. При создании учетной записи пользователя генерируются 621-битный приватный и соответствующий ему публичный ключи. Публичный ключ размещается в доменной адресной книге. Приватный ключ подвергается шифрованию закрытым ключом, который потом запрашивается у пользователя в качестве пароля, и сохраняется в идентификационном файле.

Чтобы аутентифицироваться в системе, пользователь должен указать идентификационный файл и набрать пароль, который позволит расшифровать хранящийся в файле приватный ключ. После этого все пакеты, которыми пользователь обменивается с сервером Notes, снабжаются цифровой подписью на основе этого ключа. Сервер может проверить аутентичность подписи, используя хранящийся в его адресной книге публичный ключ.

Более сложная ситуация возникает, когда пользователь должен аутентифицироваться в другом домене, в адресных книгах которого он не числится. Чтобы сделать такую авторизацию возможной, Notes вводит еще одно понятие: сертификат домена. Этот сертификат также представляет собой пару ключей, пароль к приватному ключу которой известен только администраторам домена. Каждый идентификационный файл, создаваемый в домене, подписывается приватным ключом этого сертификата.

Регистрируясь в чужом домене, пользователь предъявляет свои имя и публичный ключ, подписанные сертификатом своего домена. Если принимающий домен не знает такого сертификата, аутентификация отвергается. Чтобы домен мог признать чужой сертификат, его администратор должен провести кросс-сертификацию, а попросту говоря создать в доменной адресной книге документ, в котором хранится публичный ключ сертификата домена.

Методы, основанные на RSA и других алгоритмах шифрования, не могут решить проблемы распространения прорыва безопасности между доверяемыми системами: проникший в доверяемую систему взломщик получает доступ к приватным ключам и может использовать их для немедленной регистрации в любой из доверяющих систем или даже скопировать ключи для проникновения в эти системы в более удобное время. Шифрование приватного ключа паролем несколько усложняет осу-

ществление такой операции, но в этом случае взломщик может осуществить словарную атаку. Однако, как уже говорилось ранее, это является практически неизбежной платой за разрешение автоматической регистрации в нескольких системах. В то же время криптографические методы практически устраняют опасность имитации доверяемой системы путем подмены сетевого адреса и значительно увеличивают надежность других методов аутентификации. Например, передача пароля по сети в зашифрованном виде, особенно при использовании двухключевого шифрования или динамических ключей, практически устраняет возможность раскрытия пароля с помощью его подслушивания и т. д.

Задание на лабораторную работу

Исследовать криптографические методы аутентификации. При совпадении вариантов методов аутентификации алгоритмы шифрования должны различаться.

Вариант	**Метод**
1	Аутентификация в режиме on-line на базе симметричной криптосистемы
2	Аутентификация в режиме on-line на базе асимметричной криптосистемы
3	Аутентификация при участии нескольких серверов
4	Аутентификация в режиме off-line на базе симметричной криптосистемы
5	Аутентификация в режиме off-line на базе асимметричной криптосистемы
6	Аутентификация с привлечением арбитра на базе симметричной криптосистемы
7	Аутентификация с привлечением арбитра на базе асимметричной криптосистемы
8	Аутентификация в режиме on-line на базе симметричной криптосистемы
9	Аутентификация в режиме on-line на базе асимметричной криптосистемы
10	Аутентификация при участии нескольких серверов
11	Аутентификация в режиме off-line на базе симметричной криптосистемы
12	Аутентификация в режиме off-line на базе асимметричной криптосистемы

Содержание отчета

В отчете указать цель работы, представить результаты исследования.

Контрольные вопросы

1. Аутентификация в режиме on-line на базе симметричной криптосистемы.
2. Аутентификация в режиме on-line на базе асимметричной криптосистемы.
3. Аутентификация при участии нескольких серверов.
4. Аутентификация в режиме off-line на базе симметричной криптосистемы.
5. Аутентификация в режиме off-line на базе асимметричной криптосистемы.
6. Аутентификация с привлечением арбитра на базе симметричной криптосистемы.
7. Аутентификация с привлечением арбитра на базе асимметричной криптосистемы.

Лабораторная работа № 6

Защита на канальном уровне

Цель работы: исследовать протоколы защищенных каналов (PPTP, L2F, L2TP)

Краткие сведения из теории

Виртуальные частные сети с удаленным доступом (Virtual Private Dialup Networks — VPDN) позволяют крупным компаниям расширять свои частные сети, используя линии удаленной связи. Новые технологии снимают проблему высокой стоимости междугородней или международной связи и проблему низкой защищенности общих телефонных линий и каналов Интернет, через которые удаленный пользователь получает доступ к корпоративной сети. Новые технологии предоставляют удаленным офисам и пользователям безопасный доступ к инфраструктуре предприятия через местное подключение к сети Интернет. В настоящее время для этого используются три протокола: протокол эстафетной передачи на втором уровне (Layer 2 Forwarding — L2F), сквозной туннельный протокол (Point-to-Point Tunneling Protocol — PPTP) и туннельный протокол второго уровня (Layer 2 Tunneling Protocol — L2TP).

Протокол эстафетной передачи на втором уровне (Layer 2 Forwarding — L2F) был разработан компанией Cisco Systems. Он обеспечивает туннелирование протоколов канального уровня (то есть фреймов High-Level Data Link Control [HDLC], async HDLC или Serial Line Internet Protocol [SLIP]) с использованием протоколов более высокого уровня, например, IP. С помощью таких туннелей можно разделить местоположение сервера удаленного доступа, к которому подключается пользователь, используя местные коммутируемые линии связи, и точки, где происходит непосредственная обработка протокола удаленного доступа (SLIP, PPP), и пользователь получает доступ в сеть. Эти туннели дают возможность использовать приложения, требующие удаленного доступа с частными адресами IP, IPX и AppleTalk через протокол SLIP/PPP по существующей инфраструктуре Интернет. Поддержка таких многопротокольных приложений виртуального удаленного доступа очень выгодна конечным пользователям и независимым поставщикам услуг, поскольку позволяет разделить на всех расходы на средства доступа и базовую инфраструктуру и дает возможность осуществлять доступ через местные линии связи. Кроме того, такой подход защищает инвестиции, сделанные в существующие приложения, работающие не по протоколу

IP, предоставляя защищенный доступ к ним и в то же время поддерживая инфраструктуру доступа к Интернет.

Сквозной туннельный протокол Point-to-Point Tunneling Protocol (PPTP) создан корпорацией Microsoft. Он никак не меняет протокол PPP, но предоставляет для него новое транспортное средство. В рамках этого протокола определяется архитектура клиент/сервер, предназначенная для разделения функций, которые существуют в текущих NAS, и для поддержки виртуальных частных сетей (VPN). Сервер сети PPTP (PNS) должен работать под управлением операционной системы общего назначения, а клиент, который называется концентратором доступа к PPTP (PAC), работает на платформе удаленного доступа. PPTP определяет протокол управления вызовами, который позволяет серверу управлять удаленным коммутируемым доступом через телефонные сети общего пользования (PSTN) или цифровые каналы ISDN или инициализировать исходящие коммутируемые соединения. PPTP использует механизм общей маршрутной инкапсуляции (GRE) для передачи пакетов PPP, обеспечивая при этом контроль потоков и сетевых заторов. Безопасность данных в PPTP может обеспечиваться при помощи протокола IPSec. Протоколы L2F и PPTP имеют сходную функциональность. Компании Cisco и Microsoft согласились вместе (в рамках IETF) разработать единый стандартный протокол, который получил название туннельного протокола второго уровня (Layer 2 Tunneling Protocol — L2TP).

Задание на лабораторную работу

Исследовать протоколы туннелирования.

Содержание отчета

В отчете указать цель работы, привести результаты исследований.

Контрольные вопросы

1. Формат заголовка.
2. Типы управляющих сообщений.
3. Протокольные операции.
4. Применение протоколов туннелирования. Примеры.
5. Безопасность пакетного уровня.
6. Безопасность на конце туннеля.

Лабораторная работа № 7

Защита на сетевом уровне

Цель работы: исследование средств безопасности IPSec

Краткие сведения из теории

Безопасный протокол IP (IPSec) представляет собой набор стандартов, используемых для защиты данных и для аутентификации на уровне IP. Текущие стандарты IPSec включают независимые от алгоритмов базовые спецификации, которые являются стандартными RFC.

IPsec предназначен для безопасного взаимодействия на основе криптографии для IPv4 и IPv6. Набор сервисов безопасности включает управление доступом, целостность соединения, аутентификацию исходных данных, защиту от replay-атак (целостность последовательности), конфиденциальность (шифрование) и конфиденциальный поток трафика. Эти сервисы предоставляются на уровне IP, обеспечивая защиту для IP и/или протоколов более высокого уровня.

IPsec поддерживает две формы целостности: целостность соединения и частичную целостность последовательности. Целостность соединения является сервисом безопасности, который определяет модификацию конкретной IP датаграммы, безотносительно последовательности датаграмм в потоке трафика. Частичная целостность последовательности является anti-reply сервисом, с помощью которого определяется получение дубликатов IP датаграм.

IPsec обеспечивает сервисы безопасности на IP-уровне, выбирая нужные протоколы безопасности, определяя алгоритмы, используемые сервисами, и предоставляя все криптографические ключи требуемым сервисам. IPsec может использоваться для защиты одного или нескольких «путей» между парой хостов, между парой шлюзов безопасности или между шлюзом безопасности и хостом.

IPsec использует два протокола для обеспечения безопасности трафика – Authentication Header (AH) и Encapsulating Security Payload (ESP).

- Authentication Header (AH) обеспечивает целостность соединения, аутентификацию исходных данных и дополнительно может обеспечивать anti-replay сервис.

- Encapsulating Security Payload (ESP) протокол может обеспечивать конфиденциальность (шифрование) трафика. ESP также может обеспечивать целостность соединения, аутентификацию исходных данных и дополнительно anti-replay сервис. Целостность обеспечивается только для протоколов более высокого уровня. Хотя бы один из этих сервисов должен быть задействован при использовании ESP.

Эти протоколы могут применяться как по отдельности так и в комбинации друг с другом для обеспечения необходимого набора сервисов безопасности в IPv4 и IPv6. Каждый протокол поддерживает два режима использования: режим транспорта и режим туннелирования.

IPsec позволяет системному администратору управлять детализацией, с которой предоставляется сервис безопасности. Например, можно создать единственный зашифрованный туннель между двумя безопасными шлюзами, или для каждого TCP соединения может быть создан зашифрованный туннель между парой хостов. IPsec позволяет указывать следующие параметры:

- какие сервисы используются и в какой комбинации;
- необходимый уровень детализации применяемой защиты;
- алгоритмы, используемые для обеспечения безопасности на основе криптографии.

Протокол IPSec включает криптографические методы, удовлетворяющие потребности управления ключами на сетевом уровне безопасности. Протокол управления ключами Ассоциации безопасности Интернет (Internet Security Association Key Management Protocol — ISAKMP) создает рамочную структуру для управления ключами в сети Интернет и предоставляет конкретную протокольную поддержку для согласования атрибутов безопасности. Само по себе это не создает ключей сессии, однако эта процедура может использоваться с разными протоколами, создающими такие ключи.

Протокол определения ключей Oakley Key Determination Protocol пользуется гибридным методом Диффи-Хеллмана, чтобы создать ключи сессии Интернет для центральных компьютеров и маршрутизаторов. Протокол Oakley решает важную задачу обеспечения полной безопасности эстафетной передачи данных. Он основан на криптографических методах. Полная защита эстафетной передачи означает, что если

хотя бы один ключ раскрыт, раскрыты будут только те данные, которые зашифрованы этим ключом. Что же касается данных, зашифрованных последующими ключами, они останутся в полной безопасности.

Протоколы ISAKMP и Oakley были совмещены в рамках гибридного протокола IKE — Internet Key Exchange. Протокол IKE, включающий ISAKMP и Oakley, использует рамочную структуру ISAKMP для поддержки подмножества режимов обмена ключами Oakley. Новый протокол обмена ключами обеспечивает (в виде опции) полную защиту эстафетной передачи данных, полную защиту ассоциаций, согласования атрибутов, а также поддерживает методы аутентификации, допускающие отказ от авторства и не допускающие такого отказа. Этот протокол может, к примеру, использоваться для создания виртуальных частных сетей (VPN) и для того, чтобы предоставить пользователям, находящимся в удаленных точках (и пользующимся динамически распределяемыми адресами IP), доступ к защищенной сети.

Задание на лабораторную работу

Исследовать режимы и способы использования IPSec.

Вариант	Режим	Протокол	Тип приложение
1	Транспортный	AH	Клиент
2	Туннельный	AH	Клиент
3	Транспортный	ESP	Клиент
4	Туннельный	ESP	Клиент
5	Транспортный	AH	Сервер
6	Туннельный	AH	Шлюз
7	Транспортный	ESP	Сервер
8	Туннельный	ESP	Шлюз
9	Транспортный	AH	Сервер
10	Туннельный	AH	Шлюз
11	Транспортный	ESP	Сервер
12	Туннельный	ESP	Шлюз

Содержание отчета

В отчете указать цель работы, привести результаты исследований.

Контрольные вопросы

1. Архитектура IPSec.
2. Реализация IPSec в .NET Framework.
3. Обработка входных и выходных пакетов.
4. Протоколы IPSec.
5. Режимы IPSec.
6. Internet Key Exchange. Фазы IKE.

Лабораторная работа № 8

Защита на сеансов уровне

Цель работы: исследование протоколов SSL/TLS и SOCKS

Краткие сведения из теории

SSL – это открытый протокол, разработанный компанией Netscape. SSL определяет механизм поддержки безопасности данных на уровне между протоколами приложений (такими как Hypertext Transfer Protocol [HTTP], Telnet, Network News Transfer Protocol [NNTP] или File Transfer Protocol [FTP]) и протоколом TCP/IP. Он поддерживает шифрование данных, аутентификацию серверов, целостность сообщений и (в качестве опции) аутентификацию клиентов в канале TCP/IP. SSL был представлен рабочей группе по безопасности консорциума W3 (W3C) для утверждения в качестве стандартного средства безопасности Web-браузеров и серверов в сети Интернет.

Основная цель протокола SSL состоит в том, чтобы обеспечить защищенность и надежность связи между двумя подключенными друг к другу приложениями. Этот протокол состоит из двух уровней. Нижний уровень, который располагается поверх надежного транспортного протокола (например, TCP), называется SSL Record Protocol. SSL Record Protocol используется для встраивания различных протоколов высокого уровня. Один из таких встроенных протоколов, SSL Handshake Protocol, позволяет серверу и клиенту аутентифицировать друг друга и согласовывать алгоритм шифрования и криптографические ключи, прежде чем протокол приложения произведет обмен первыми битами данных. Одно из преимуществ SSL состоит в том, что он независим от протоколов приложений. Протокол высокого уровня может совершенно прозрачно располагаться поверх протокола SSL. Протокол SSL поддерживает безопасность связи, придавая ей следующие свойства:

- Защищенность связи. После первоначального квитирования связи применяются средства шифрования и определяется секретный ключ. Для шифрования данных используются средства симметричной криптографии (например, DES).
- Участник сеанса связи может быть аутентифицирован средствами асимметричной криптографии (например, RSA, DSS).
- Надежность связи. Транспортные средства проводят проверку целостности сообщений с помощью зашифрованного кода целостности (MAC). Для вычисления

кодов MAC используются безопасные хэш-функции (SHA, MD5).

Протокол SSL принят только в рамках HTTP. Другие протоколы доказали свою способность работать с SSL, но используют ее не часто.

SOCKS разработан для того, чтобы дать возможность приложениям клиент/сервер в доменах TCP и UDP удобно и безопасно пользоваться услугами межсетевого экрана. Он дает пользователям возможность преодолевать межсетевой экран организации и получать доступ к ресурсам, расположенным в сети Интернет. SOCKS является «посредником уровня приложений»: он взаимодействует с общими сетевыми средствами (например, Telnet и браузер Netscape) и с помощью центрального сервера (прокси-сервера) от имени компьютера устанавливает связь с другими центральными компьютерами.

SOCKS версия 4 решает вопрос незащищенного пересечения межсетевых экранов приложениями клиент/сервер, основанными на протоколе TCP, включая Telnet, FTP и популярные информационные протоколы, такие как HTTP, Wide Area Information Server (WAIS) и GOPHER. SOCKS версия 5, RFC 1928, является дальнейшим расширением четвертой версии SOCKS. Он включает в себя UDP, расширяет общую рамочную структуру, придавая ей возможность использования мощных обобщенных схем аутентификации, и расширяет систему адресации, включая в нее имя домена и адреса IP v6.

В настоящее время предлагается создать механизм управления входящими и исходящими многоадресными сообщениями IP, которые проходят через межсетевой экран. Это достигается определением расширений для существующего протокола SOCKS V.5, что создает основу для аутентифицированного перехода межсетевого экрана одноадресным пользовательским трафиком TCP и UDP. Однако ввиду того, что поддержка UDP в текущей версии SOCKS V.5 имеет проблемы с масштабируемостью и другие недостатки (и их обязательно нужно разрешить, прежде чем переходить к многоадресной передче), расширения определяются двояко: как базовые расширения UDP и как многоадресные расширения UDP.

Функционирование SOCKS заключается в замене стандартных сетевых системных вызовов в приложении их специальными версиями. Эти новые системные вы-

зовы устанавливают связь с прокси-сервером SOCKS (который конфигурируется самим пользователем в приложении или системным файлом конфигурации), подключаясь к хорошо известному порту (обычно это порт 1080/TCP). После установления связи с сервером SOCKS приложение отправляет серверу имя машины и номер порта, к которому хочет подключиться пользователь. Сервер SOCKS реально устанавливает связь с удаленным центральным компьютером, а затем прозрачно передает данные между приложением и удаленной машиной.

Трудность с использованием SOCKS состоит в том, что кто-то должен проводить работу по замене сетевых системных вызовов версиями SOCKS (этот процесс обычно называется «SOCKS-ификацией» приложения). Большинство обычных сетевых приложений (Telnet, FTP, finger, whois) уже SOCKS-ифицированы, и многие производители включают поддержку SOCKS в свои коммерческие приложения.

Задание на лабораторную работу

Студенты соответствующих вариантов объединяются в группу для исследования протоколов.

Вариант	**Протокол**	**Приложение**
1	SSL Протокол рукопожатия	Клиент
2		Сервер
3	SSL Протокол тревоги	Клиент
4		Сервер
5	SSL Протокол изменения шифра	Клиент
6		Сервер
7	SSL Протокол приложения	Клиент
8		Сервер
9	SOCKS 4	Клиент
10		Сервер
11	SOCKS 5	Клиент
12		Сервер

В работе необходимо исследовать устойчивость протоколов к известным атакам.

Содержание отчета

В отчете указать цель работы, представить результаты исследований.

Контрольные вопросы

1. Цель использования SSL/TLS.
2. Цель использования SOCKS.
3. Способы получения SSL-сертификата/
4. Механизмы образования ключа для текущего сессии в SSL/TLS.
5. Применение SSL/TLS.

Приложение 1

Простые числа для лабораторных работ

Вариант	Простые числа
1	170141183460469231731687303715884105757 170141183460469231731687303715884105773 170141183460469231731687303715884105793 170141183460469231731687303715884105829 170141183460469231731687303715884105851
2	170141183460469231731687303715884105979 170141183460469231731687303715884106001 170141183460469231731687303715884106031 170141183460469231731687303715884106123 170141183460469231731687303715884106207
3	170141183460469231731687303715884106213 170141183460469231731687303715884106231 170141183460469231731687303715884106273 170141183460469231731687303715884106303 170141183460469231731687303715884106309
4	170141183460469231731687303715884106319 170141183460469231731687303715884106409 170141183460469231731687303715884106439 170141183460469231731687303715884106721 170141183460469231731687303715884106723
5	170141183460469231731687303715884106787 170141183460469231731687303715884107009 170141183460469231731687303715884107029 170141183460469231731687303715884107149 170141183460469231731687303715884107237
6	170141183460469231731687303715884107339 170141183460469231731687303715884107467 170141183460469231731687303715884107477 170141183460469231731687303715884107579 170141183460469231731687303715884107587
7	170141183460469231731687303715884107621 170141183460469231731687303715884107639 170141183460469231731687303715884107717 170141183460469231731687303715884107923 170141183460469231731687303715884108073
8	170141183460469231731687303715884108077 170141183460469231731687303715884108227 170141183460469231731687303715884105727 170141183460469231731687303715884105703 170141183460469231731687303715884105689

9	17014118346046923173168730371588410543317014118346046923173168730371588410541917014118346046923173168730371588410522117014118346046923173168730371588410521717014118346046923173168730371588410515117014118346046923173168730371588410503117014118346046923173168730371588410503117014118346046923173168730371588410499317014118346046923173168730371588410492717014118346046923173168730371588410477117014118346046923173168730371588410464717014118346046923173168730371588410462317014118346046923173168730371588410458717014118346046923173168730371588410452717014118346046923173168730371588410449717014118346046923173168730371588410428117014118346046923173168730371588410424317014118346046923173168730371588410423317014118346046923173168730371588410398717014118346046923173168730371588410398117014118346046923173168730371588410396
10	
11	
12	

Приложение 2

Параметры эллиптической кривой для лабораторных работ

Вариант	**Параметры эллиптической кривой**
1	p = DB7C 2ABF62E3 5E668076 BEAD208B a = DB7C 2ABF62E3 5E668076 BEAD2088 b = 659E F8BA0439 16EEDE89 11702B22 G = 04 09487239 995A5EE7 6B55F9C2 F098A89C E5AF8724 C0A23E0E 0FF77500 n = DB7C 2ABF62E3 5E7628DF AC6561C5
2	p = DB7C 2ABF62E3 5E668076 BEAD208B a = 6127 C24C05F3 8A0AAAF6 5C0EF02C b = 51DE F1815DB5 ED74FCC3 4C85D709 G = 04 4BA30AB5 E892B4E1 649DD092 8643ADCD 46F5882E 3747DEF3 6E956E97 n = 36DF 0AAFD8B8 D7597CA1 0520D04B
3	p = FFFFFFFD FFFFFFFF FFFFFFFF FFFFFFFF a = FFFFFFFD FFFFFFFF FFFFFFFF FFFFFFFC b = E87579C1 1079F43D D824993C 2CEE5ED3 G = 04 161FF752 8B899B2D 0C28607C A52C5B86 CF5AC839 5BAFEB13 C02DA292 DDED7A83 n = FFFFFFFE 00000000 75A30D1B 9038A115
4	p = FFFFFFFD FFFFFFFF FFFFFFFF FFFFFFFF a = D6031998 D1B3BBFE BF59CC9B BFF9AEE1 b = 5EEEFCA3 80D02919 DC2C6558 BB6D8A5D G = 04 7B6AA5D8 5E572983 E6FB32A7 CDEBC140 27B6916A 894D3AEE 7106FE80 5FC34B44 n = 3FFFFFFF 7FFFFFFF BE002472 0613B5A3
5	p = FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFE FFFFAC73 a = 00000000 00000000 00000000 00000000 00000000 b = 00000000 00000000 00000000 00000000 00000007 G = 04 3B4C382C E37AA192 A4019E76 3036F4F5 DD4D7EBB 938CF935 318FDCED 6BC28286 531733C3 F03C4FEE n = 01 00000000 00000000 0001B8FA 16DFAB9A CA16B6B3
6	p = FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF 7FFFFFFF a = FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF 7FFFFFFC b = 1C97BEFC 54BD7A8B 65ACF89F 81D4D4AD C565FA45 G = 04 4A96B568 8EF57328 46646989 68C38BB9 13CBFC82 23A62855 3168947D 59DCC912 04235137 7AC5FB32 n = 01 00000000 00000000 0001F4C8 F927AED3 CA752257
7	p = FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFE FFFFAC73 a = FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFE FFFFAC70 b = B4E134D3 FB59EB8B AB572749 04664D5A F50388BA G = 04 52DCB034 293A117E 1F4FF11B 30F7199D 3144CE6D FEAFFEF2 E331F296 E071FA0D F9982CFE A7D43F2E n = 01 00000000 00000000 0000351E E786A818 F3A1A16B
8	p = FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFE FFFFEE37 a = 00000000 00000000 00000000 00000000 00000000 00000000 b = 00000000 00000000 00000000 00000000 00000000 00000003

	G = 04 DB4FF10E C057E9AE 26B07D02 80B7F434 1DA5D1B1 EAE06C7D 9B2F2F6D 9C5628A7 844163D0 15BE8634 4082AA88 D95E2F9D n = FFFFFFFF FFFFFFFF FFFFFFFE 26F2FC17 0F69466A 74DEFD8D
9	p = FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFE FFFFFFFF FFFFFFFF a = FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFE FFFFFFFF FFFFFFFC b = 64210519 E59C80E7 0FA7E9AB 72243049 FEB8DEEC C146B9B1 G = 04 188DA80E B03090F6 7CBF20EB 43A18800 F4FF0AFD 82FF1012 07192B95 FFC8DA78 631011ED 6B24CDD5 73F977A1 1E794811 n = FFFFFFFF FFFFFFFF FFFFFFFF 99DEF836 146BC9B1 B4D22831
10	p = FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFE FFFFE56D a = 00000000 00000000 00000000 00000000 00000000 00000000 00000000 b = 00000000 00000000 00000000 00000000 00000000 00000000 00000005 G = 04 A1455B33 4DF099DF 30FC28A1 69A467E9 E47075A9 0F7E650E B6B7A45C 7E089FED 7FBA3442 82CAFBD6 F7E319F7 C0B0BD59 E2CA4BDB 556D61A5 n = 01 00000000 00000000 00000000 0001DCE8 D2EC6184 CAF0A971 769FB1F7
11	p = FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF 00000000 00000000 00000001 a = FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFE FFFFFFFF FFFFFFFF FFFFFFFE b = B4050A85 0C04B3AB F5413256 5044B0B7 D7BFD8BA 270B3943 2355FFB4 G = 04 B70E0CBD 6BB4BF7F 321390B9 4A03C1D3 56C21122 343280D6 115C1D21 BD376388 B5F723FB 4C22DFE6 CD4375A0 5A074764 44D58199 85007E34 n = FFFFFFFF FFFFFFFF FFFFFFFF FFFF16A2 E0B8F03E 13DD2945 5C5C2A3D
12	p = FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFE FFFFFC2F a = 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 b = 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000007 G = 04 79BE667E F9DCBBAC 55A06295 CE870B07 029BFCDB 2DCE28D9 59F2815B 16F81798 483ADA77 26A3C465 5DA4FBFC 0E1108A8 FD17B448 A6855419 9C47D08F FB10D4B8 n = FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFE BAAEDCE6 AF48A03B BFD25E8C D0364141

Приложение 3

Варианты открытого текста для лабораторных работ

Вариант	Текст
1	They met me in the day of success: and I have learned by the perfectest report, they have more in them than mortal knowledge.
2	When I burned in desire to question them further, they made themselves air, into which they vanished.
3	Whiles I stood rapt in the wonder of it, came missives from the king, who all-hailed me 'Thane of Cawdor;
4	by which title, before, these weird sisters saluted me, and referred me to the coming on of time, with 'Hail, king that shalt be!'
5	This have I thought good to deliver thee, my dearest partner of greatness, that thou mightst not lose the dues of rejoicing, by being ignorant of what greatness is promised thee.
6	Lay it to thy heart, and farewell.' Glamis thou art, and Cawdor; and shalt be What thou art promised: yet do I fear thy nature;
7	It is too full o' the milk of human kindness To catch the nearest way: thou wouldst be great;
8	Art not without ambition, but without The illness should attend it: what thou wouldst highly, That wouldst thou holily;
9	wouldst not play false, And yet wouldst wrongly win: thou'ldst have, great Glamis, That which cries 'Thus thou must do, if thou have it;
10	And that which rather thou dost fear to do Than wishest should be undone.' Hie thee hither, That I may pour my spirits in thine ear;
11	And chastise with the valour of my tongue All that impedes thee from the golden round, Which fate and metaphysical aid doth seem To have thee crown'd withal.
12	Thou'rt mad to say it: Is not thy master with him? who, were't so, Would have inform'd for preparation.

Литература

1. Алексеев, А.П.; Многоуровневая защита информации Электронный ресурс : монография / А.П. Алексеев. - Самара : Поволжский государственный университет телекоммуникаций и информатики, 2017. - 128 с. - Книга находится в базовой версии ЭБС IPRbooks. - ISBN 978-5-904029-72-2

2. Бурькова, Е. В; Физическая защита объектов информатизации : учебное пособие / Е.В. Бурькова ; Министерство образования и науки Российской Федерации ; Оренбургский государственный университет ; Кафедра вычислительной техники и защиты информации. - Оренбург : Оренбургский государственный университет, 2017. - 158 с. : табл., схем. - http://biblioclub.ru/. - Библиогр. в кн. - ISBN 978-5-7410-1697-8

3. Горев, А.И.; Обработка и защита информации в компьютерных системах Электронный ресурс : практическое пособие / А.А. Симаков / А.И. Горев. - Омск : Омская академия МВД России, 2016. - 88 с. - Книга находится в базовой версии ЭБС IPRbooks. - ISBN 978-5-88651-642-5

4. Шаньгин, В. Ф; Информационная безопасность и защита информации Электронный ресурс / В. Ф. Шаньгин. - Информационная безопасность и защита информации,2019-04-19. - Саратов : Профобразование, 2017. - 702 с. - Книга находится в премиум-версии ЭБС IPR BOOKS. - ISBN 978-5-4488-0070-2