Моделирование управлением доступа в ИС.
Протокол Керберос.
Безопасность парольной защиты ИС.

Контроль доступа в информационных системах

Цель: защитить конфиденциальность и целостность информации.

•Контроль того, что субъект может сделать, чтобы предотвратить повреждение системы.

•Регулирование операций, которые может выполнять пользователь по отношению к данным и ресурсам

•Поставляется в составе операционных систем и систем. управление базами данных

Идея: Основная идея контроля доступа - наличие активного субъекта, которому требуется доступ к пассивному объекту для того, чтобы выполнить определенную операцию доступа

Контрольный монитор разрешает или запрещает доступ

Субъекты (сущности)

Активный объект, выполняющий операции в системе.

Объекты можно классифицировать как:
пользователи: лица, подключающиеся к системе
группы: группы пользователей
роли: функция или положение в организации
процессы: выполнение программ от имени пользователей

Между разными типами объектов могут быть связи.
Объект: любой системный ресурс: файл, принтер, хост, комната, здание и т. д.
Защищаемые объекты: объекты, контролируемые системой.

Права доступа

•Операции, которые субъект может выполнять с объектами безопасности.

• Каждый тип операции имеет права доступа

- контроль доступа должен иметь возможность управлять конкретным типом операции

Самый простой пример прав доступа: -

читать: посмотреть содержимое объекта

записать: изменить содержимое объекта

•Другие типы прав в зависимости от защищаемых ресурсов:

выполнить, выбрать, вставить, обновить, удалить и т. д.

•Дополнительные права: право собственности, делегирование

Контроль доступа и аутентификация

Совершенно разные вещи:

Аутентификация: определение того, кто вы, есть ли у пользователя конкретный атрибут или нет

Контроль доступа: установление права пользователя на осуществление определенных видов деятельности

!!! Для управления доступом требуется аутентификация.

Модели защиты ресурсов ИС

Мандатная (mandatory) **– *каждому объекту назначается классификационны уровень (гриф секр.) , каждому пользователю (субъекту) – уровень допуска*;**

Классическая модель **Белла—Лападулы** (1975) Дэвид Белл и Леонард Лападула: для разграничения доступа к секретным документам в правит. Организациях США:

Сов. секр > Секр > ДСП > Без грифа

Мандатное управление доступом (*Mandatory access control, MAC*) — разграничение доступа субъектов к объектам, основанное на назначении метки конфиденциальности для информации, содержащейся в объектах, и выдаче официальных разрешений (допуска) субъектам на обращение к информации такого уровня конфиденциальности (иногда - **Принудительный контроль доступа**)

В SUSE Linux (дистрибутив) реализована архитектура мандатного контроля доступа под названием AppArmor.

В PostgreSQL появилась начальная поддержка selinux (*Security-Enhanced Linux* — Linux с улучшенной безопасностью)

Дискреционная, избирательная (discretionary) **– каждому пользователю назначаются разле права доступа (привилегии)**

Избирательное управление доступом (*discretionary access control*, DAC) — управление доступом субъектов к объектам на основе списков управления доступом (**Access Control List**) или матрицы доступа.

Также используются названия *"дискреционное управлением доступом"*, *"контролируемое управление доступом"* или *"разграничительное управление доступом"*.

Другим вариантом решения это проблемы является «Управление доступом на основе **ролей**» - *Role Based Access Control, RBAC* **– в *СУБД***

Access Control List или **ACL** — список управления доступом, который определяет, кто или что может получать доступ к конкретному объекту, и какие именно операции разрешено или запрещено этому субъекту проводить над объектом.

Пример. Запись *(Vanya, delete)* в ACL для файла *XYZ* даёт возможность пользователю *Vanya* удалить файл *XYZ*

Список контроля доступа - структура данных (обычно таблица), содержащая записи, определяющие права индивидуального пользователя или группы на специальные системные объекты, такие как программы, процессы или файлы.

Эти записи также известны как **ACE** (*Access Control Entries*) в операционных системах Microsoft Windows

В сетях: **ACL** - список правил, определяющих порты служб или имена доменов, доступных на узле или другом устройстве третьего уровня OSI, каждый со списком узлов и/или сетей, которым разрешен доступ к сервису.

Сетевые ACL могут быть настроены как на обычном сервере, так и на маршрутизаторе и могут управлять как входящим, так и исходящим трафиком, в качестве межсетевого экрана.

Подходы к построению :

- каждый объект системы имеет привязанного к нему субъекта, называемого **владельцем**; владелец устанавливает права доступа к объекту,
- система имеет одного выделенного субъекта — суперпользователя - устанавливает права владения для всех остальных субъектов,
- субъект с определенным правом доступа может передать это право любому другому субъекту

В смешанных вариантах (есть владелец и суперпол-ль) – ОС Windows

В файловых системах для реализации ACL используется идентификатор пользователя процесса.

Радужные Серии (The Rainbow Series)

• Национальный центр компьютерной безопасности, США (National Computer Security Center) – рекомендации, относящиеся к безопасности сетей в рамках программы технического руководства

• Серия Rainbow (известная как Rainbow Books) - это серия стандартов i. руководящих принципов компьютерной безопасности, опубликованных Правительство США в 80-х и 90-х годах.

• Программа представляет подробный обзор функций и критериев оценки безопасности компьютерных систем (**DOD 5200.28-STD**) и обеспечение руководства по выполнению каждого требования.

• Национальный центр компьютерной безопасности через Программу оценки безопасных продуктов - анализ функций безопасности производимых и эксплуатируемых компьютерных систем.

• **Все эти программы предоставляют организациям возможность защитить важные данные с использованием надежных компьютерных систем**.

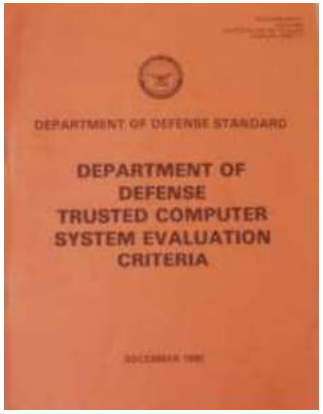
Серия «Радуга» - это стопка книг, посвященных оценке «Надежных компьютерные системы» по данным агентства национальной безопасности.

• Термин «серия радуги» происходит от того факта, что обложка каждой книги отличается цветом.

Полный список книг - см .:

https://en.wikipedia.org/wiki/Rainbow_Series

Основная книга (на которой основаны все остальные) – Оранжевая книга (документ 5200.28-STD) – Критерии оценки безопасности компьютерных систем (Trusted Computer System Evaluation Criteria**) , 15 августа 1983 г.** https://csrc.nist.gov/csrc/media/publications/white-paper/1985/12/26/dod-rainbow series / final / documents / std001.txt



Основные правила.

•Политика безопасности должна быть открытой, четко определенной

• Определены три основных правила безопасности:

Обязательная политика безопасности - обеспечивает соблюдение правил контроля доступа, основанных на непосредственном согласовании прав физического лица на разрешение на доступа к информация с уровнем конфиденциальности запрашиваемой информации.

Эта политика также должна точно отражать законы, общие принципы и другие соответствующие законы и руководящие принципы, из которых они исходят.

Маркировка системы, предназначенная для исполнения правил политики безопасности, сохранения и поддержания целостности меток контроля доступа, если объект экспортируется.

Дискреционная политика безопасности - обеспечивает согласованный набор правил контроля и ограничение доступа на основании идентифицированных лиц, которые были определены как потребителиинформации.

Ответственность - индивидуальная ответственность должна соблюдаться независимо от реализуемой политики.

• Должны существовать безопасные средства предоставления доступа уполномоченному и компетентному агенту, который затем сможет оценить информацию об исполнении ответственности в разумные сроки и без неоправданных трудностей.

• Задача соблюдения имеет три требования:

Идентификация - процесс, используемый для индивидуального распознавания Пользователя.

Аутентификация - проверка авторизации отдельного пользователя по отношению к определенным категориям информации.

Аудит - информация аудита должна выборочно храниться и защищаться, чтобы действия по обеспечению безопасности можно было проследить до аутентифицированного лица.

The Trusted Network Interpretation Environments Guideline (RED BOOK)

Руководство по интерпретации защищенных сетей (NCSC-TG 011) - дополняет раздел **Критерии оценки защищенных (безопасных) вычислительных систем** (NCSC-TG-O5) (NCSC-TG-O5), 31 июля 1987 г. Этот документ определяет минимальную безопасность, требуемую в различных сетевых средах, чтобы органы по сертификации, сетевые интеграторы и сетевые аккредитаторы могли определить, какие механизмы защиты и гарантии минимально требуются для конкретных сетевых сред.

Основные особенности:

• Network Trusted Computing Base (NTCB) - это совокупность механизмов защиты в сетевой системе, включая оборудование, микропрограммное обеспечение и программное обеспечение, комбинация которых отвечает за соблюдение политики безопасности.

• Раздел NTCB - это совокупность механизмов применения сетевой политики в одной сетевой подсистеме, назначенной этой подсистеме; он является частью NTCB в рамках единой сетевой подсистемы.

Network Security Architecture and Design (NSAD)

определяет, как NTCB **разбивается на классы** и как выполняются требования к надежной системе.
Техника обеспечения безопасности, включая NSAD Development, - это специализация системного инжиниринга.
• Инженер по безопасности (администратор) отвечает за обеспечение соответствия создаваемой системы требованиям безопасности организации. Инженер по безопасности гарантирует, что безопасность системы соответствует применимым нормам и политика и реализует требования безопасности системы.•

!!! По данным Национального центра компьютерной безопасности, все сетевые системы можно разделить на **4 класса: D, C, B или A:**
Класс D - системы, которые были оценены, но не соответствуют требованиям для более высокого рейтинга NCSC.
Класс C имеет два подкласса, **C1 и C2**, которые требуют дискреционной (дискреционной) защиты («необходимо знать»).
Класс B имеет три подкласса: **B1, B2 и B3**, которые требуют обязательной защиты и повышают устойчивость архитектуры системы.
Класс A требует дополнительных гарантий с помощью формальных методов проверки.

Определение класса безопасности С

Дискреционная защита (Discretionary Access Control, DAC) С1

- Подтверждение защиты безопасности

• Идентификация и аутентификация

• Разделение пользователей и данных

• Дискреционный контроль доступа (DAC), способный применять ограничения доступа к политике по отношению к физическому лицу

• Необходимая системная документация и руководства пользователя.

С2 - контролируемая защита

• Применение цифро-аналоговых преобразователей

• Индивидуальная ответственность через процедуры входа в систему

• Контрольные журналы

• Использование объектов изоляции ресурсов

Пример такой системы: HP-UX (Unix), также Wintows xx

B - Обязательная защита (MandatoryAccess Control, MAC)

B1 - Маркированная защита безопасности

• Неформальное заявление о модели политики безопасности.
• Ярлыки конфиденциальности данных
• Обязательный контроль доступа (MAC) к выбранным темам и объектам
• Возможность экспорта этикеток.
• Некоторые обнаруженные дефекты необходимо устранить или нейтрализовать иным образом.
• Проектные спецификации и проверка

B2 - Структурная защита

• Четко определенная и официально задокументированная модель политики безопасности.
• Обеспечение соблюдения DAC и MAC распространяется на все объекты и субъекты.
• Скрытые каналы памяти анализируются на наличие уязвимостей и пропускную способность
• Разделение на компоненты, критичные для безопасности и не критичные для безопасности.
• Дизайн и реализация позволяют проводить более всестороннее тестирование и анализ.
• Усилены механизмы аутентификации.
• Управление доверенным объектом осуществляется с разделением администратора и оператора.
• Установлены строгие меры управления конфигурацией.
• Роли оператора и администратора разделены.

Примером такой системы была MultICS (Multiplexed Information and Computing Service).
- ранняя операционная система с разделением времени, основанная на концепции одноуровневой памяти

B3 - Домены безопасности

• Отвечает требованиям для эталонного монитора.
• Организован так, чтобы исключить код, который не важен для обеспечения безопасности.
• Значительная системная инженерия, направленная на минимизацию сложности
• Определена **роль администратора безопасности**.
• Аудит инцидентов безопасности
• Автоматическое обнаружение надвигающегося вторжения, уведомление и ответ
• Надежный путь к TCB (доверительной вычислительной базе) для функции аутентификации пользователя
• Процедуры восстановления надежных систем
• Скрытые временные каналы анализируются на предмет присутствия уязвимостей и пропускной способности

Примером такой системы является XTS-300, предшественник XTS-400 (многоуровневая безопасная операционная система может использоваться в междоменных решениях, которые обычно требуют разработки привилегированного программного обеспечения

A – Верифицированная защита

A1 - Подтвержденный проект

• Функционально идентичен B3

• Методы формального проектирования и проверки, включая формальную спецификацию верхнего уровня.

• Официальные процедуры управления и распределения

Примерами систем класса A1 являются SCOMP от Honeywell, GEMSOS от Aesec и SNS от Boeing.

Над A1

• Архитектура системы показывает, что требования самозащиты и полноты для эталонных мониторов были реализованы в TCB (Trusted Computing Base).

• Тестирование безопасности автоматически генерирует тестовый пример либо из формальной спецификации верхнего уровня, либо из формальной спецификации нижнего уровня.

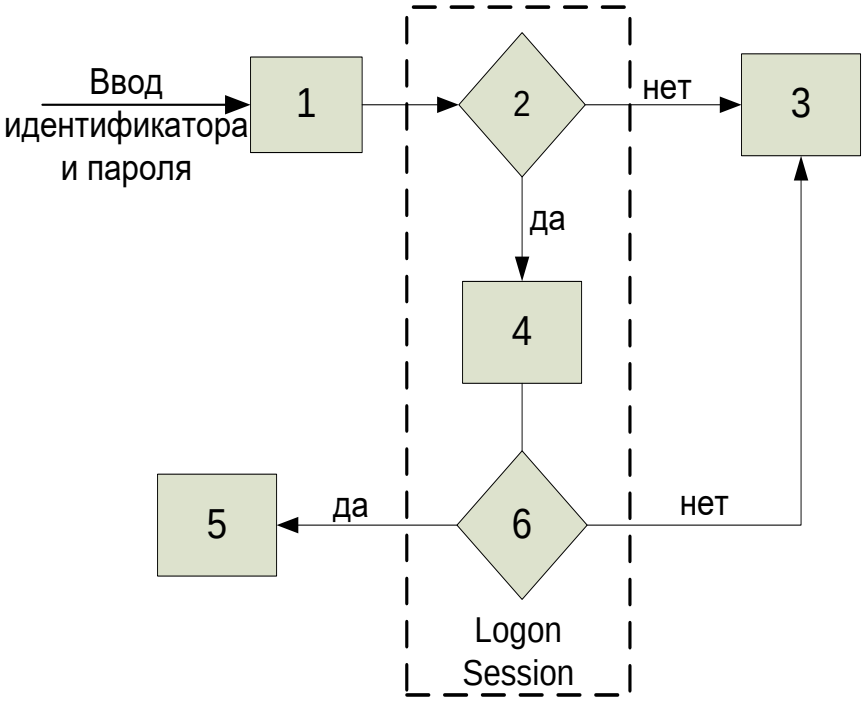
• Формальная спецификация и проверка имеют место, когда TCB проверяется до уровня исходного кода с использованием формальных методов проверки, где это возможно.

• Надежная среда разработки - это среда, в которой TCB создается на надежном предприятии, где работает только проверенный (проверенный) персонал.

Защита программного обеспечения

Защита ОС Windows

1. Защита паролем



1 – прием системой ID и Password ;

2 – проверка идентификатора и пароля на предмет содержания ID и пароля в базе Windows (в базе SAM (Security Account Manager);

3 – если ID и пароль не найдены, то выводится сообщение об отказе в доступе;

4 – устанавливаются полномочия пользователя с введенным паролем;

6 – имеет ли он полномочия на работу с ресурсами?

Если нет, то следует отказ в доступе (3). Если да, то

5 – разрешение в использовании ресурса

Рис.1. Процедура идентификации, аутентификации и установления полномочий пользователя

Идентифика́ция (от лат. *identifico* — отождествлять) в ИС — установление субъекта по имеющемуся у него идентификатору

Примеры: идентификация товара по штрих-коду, идентификация пользователя по логину, идентификация файла по контрольной сумме

Аутентифика́ция (*Authentication*) — процедура проверки подлинности

Примеры: проверка подлинности пользователя путём сравнения введённого им пароля с паролем в базе данных пользователей; подтверждение подлинности электронного письма путём проверки цифровой подписи письма по ключу шифрования отправителя; проверка **контрольной суммы** файла на соответствие сумме, заявленной автором этого файла

Эффективность использования пароля

- **A = {a_i}** – алфавит, состоящий из фиксированного набора символов, **i** ∈ **[1, N], N** – мощность алфавита
- **s -** длина пароля **H**; при **H** = ‘12AAa!!*’ **s** = 8
- Кол-во комбинаций пароля при фиксир **N** : $\mathbf{I_H = N^s}$;

Пример1. **A** = **{a,b,c,d,…,z}** , N=26; при **s** = 8 $\mathbf{N^s = 26^8}$=208 827 064 576

- Безопасное время использования пароля

$$\mathbf{t_H = ½ (I_H \cdot t),} \qquad (1)$$

$$\mathbf{t=E/R,\ E= S + S_{sl};}$$

Пример2. **N** = 5 симв, S = 6 симв, скорость передачи **R** = 3 [Кбит/с]; принимаем $\mathbf{S_{sl}}$=4 симв, тогда **E**= 6+4=10 симв (либо 80 бит) и

$\mathbf{t_H = ½ (I_H \cdot t) = 1/2(5^6 * 80/(3*1024)) = 203}$ с

Пример3. **N** = 26 симв, **S** = 6 симв, скорость передачи **R** = 32 [Кбит/с]; принимаем $\mathbf{S_{sl}}$=14 симв, тогда **E**= 6+14=20 симв (либо 160 бит) и

$\mathbf{t_H = ½ (I_H \cdot t) = 1/2(26^6 * 160/(32*1024)) = 7.5* 10^5}$ с = **3.5** ч

Безопасное время использования пароля

Принимаем **P** – это вероятность того, что пароль будет взломан за **M** мес,

P_0 – нижняя граница **P**; $P_0 = n1/n2$; **n1** – число попыток взлома пароля за **M** мес; **n2 –** число всех возможных паролей при определенных **N** и **s;**

n1 = **n11/ n12; n11 –** число символов, которые можно передать по сети за **M** мес, **n12** – число символов, передаваемых в одной попытке;

$$n1 = (R*M*24(ч/д)*60(мин/ч)*60\ (сек/мин)*30(д/мес)\ )/E\ ,\quad (2)$$

$$n2 = N^s,$$

тогда $P_0 = (R*M*24*60*60*30)/(E*N^s)$. (3)

Так как $P > P_0$, $P > (R*M*24*60*60*30)/(E*N^s)$ или иначе

$$N^s >= (4.32 * 10^4 *R*M)/(E *P)$$ - ф-ла *Андерсена* (4)

$$N^s >= (2.59 * 10^6 *R*M)/(E *P)$$

Пример. $P = 10^{-3}$, M = 3;R = 10 (сим/сек);E = 20 (сим);N = 26 (сим);s = 6 (сим);

$(2.59 * 10^6 * R * M)/(E * P) =$ $(2.59 * 10^6 * 10 * 3)/(20 * 10^{-3}) = 3.9 * 10^9$;

$N^s = 26^6 \approx 3.089 \cdot 10^8 \leq 3.9 \cdot 10^9$.

Это означает, что при выбранном размере алфавита и длине пароля, необходимое условие неравенства не выполняется.

При s = 7 (сим):

$26^7 \approx 8.03 \cdot 10^9 \geq 3.9 \cdot 10^9$.

Выполнение условия означает, что для выбранного алфавита пароль длиной 7 символов будет взоман за 3 месяца с вероятностью не более, чем ***$P = 10^{-3}$.***

Протокол Kerberos

- **Назначение** - для пересылки зашифрованного сообщения (A ⟶ B) по открытым каналам на платформе ОС **Windows** при взаимод с **T;**
- **Опирается** на протокол **Нидхэма-Шрёдера (R. Needham-M. Schröder)** и базируется на симметричном шифровании данных

Протокол Нидхэма-Шрёдера

Обозначения: **A, B, T** – имена участников, E_A - ключ, общий для **A и T**, E_B – ключ, общий для **B и T**

1. **A** ⟶ **T**: **A, B,** R_A ; R_A – случ число, сгенерир-е **A**
2. **T** генерир-т случ сеансовый ключ **K;** затем щифрует **:**

C = $E_A(R_A, B, K; E_B(K, A))$; **T: C** ⟶ **A**

3. **A** извлекает из **C: K** и убеждается, что R_A равно R_A для 1-го этапа;

извлекает $E_B(K, A) = C_3$; **A:** C_3 ⟶ **B**

4. **B,** используя E_B, извлекает **K** из C_3; **B** генерирует случ число R_B, создает шифртекст $C_4 = K(R_B)$ и **B**: C_4 ⟶ **A**
5. **A** расш-т C_4 ключом **K,** создает шифртекст $C_5 = K(R_B - 1)$; **A:** C_5 **B**
6. **B** расш-т C_5 ключом **K** и убеждается, что **известное ему** R_B **уменьшено на 1;** ***Т. о. создан секретный сеансовый ключ K для A и B***

- Установленная в сети TCP/IP служба Kerberos, является доверенной стороной (**T**)
- Основой Kerberos является БД Клиентов и их секретных ключей
- Сетевые службы, которые требуют аутентификацию, должны зарегистрировать в Kerberos свои секретные ключи
- Так как Kerberos знает все секретные ключи, он может убеждать одни объекты в подлинности других. Керберос создает сеансовые ключи, которые выдаются Клиенту и Серверу, и никому больше
- Для шифрования используется алгоритм DES
- Для организации канала связи Клиент запрашивает у Kerberos разрешение на обращение к службе организации таких сообщений, эта служба называется **Ticket Granting Service** (TGS) — служба выделения мандата

Протокол Kerberos

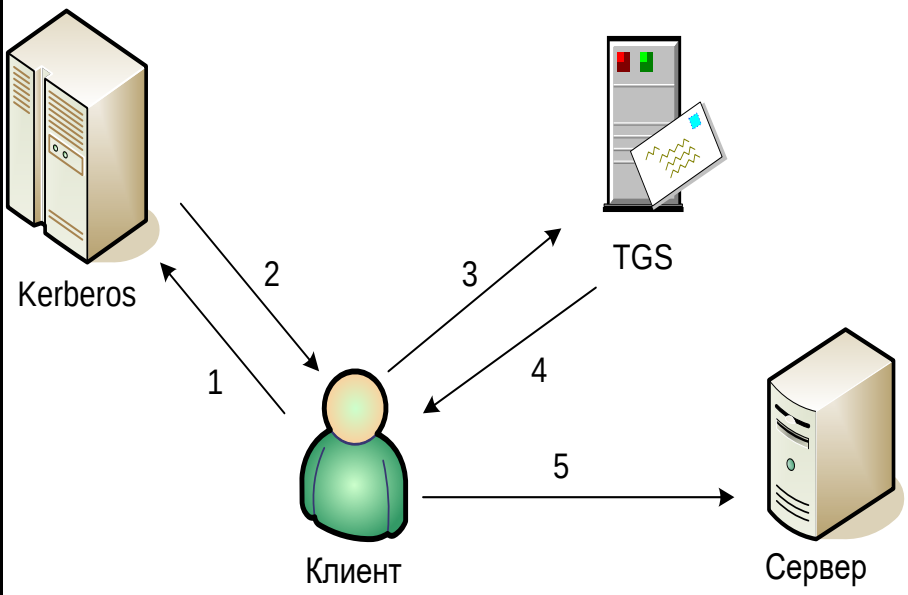


Рис.2. Общая схема взаимодействия компонент в протоколе Kerberos

1 — Клиент запрашивает Керберос разрешение на обращение к службе TGS.

2 — После анализа предоставленных документов о возможности организации сообщения между Кл и Серв Керберос выдает Кл-ту соответствующее разрешение.

3 — Пользуясь разрешением службы Керберос, Кл запрашивает TGS о выделении ему мандата на организацию канала между Клиентом и Сервером.

4 — Получение такого мандата.

5 — Клиент пересылает соответствующее сообщение серверу

C— Клиент (Client),

S – Сервер (Server),

A — Сетевой адрес Клиента (Address) — имя Клиента,

v — Временная метка, содержащая начальное и конечное время действия мандата,

t — просто метка времени, соответствующая периоду времени, в течение которого действует сеансовый ключ,

Kx — секретный ключ объекта **X**,

Kx,y — сеансовый ключ для организации сеанса между **X** и **Y**,

{m}K_x — сообщение **m**, зашифрованное ключом **Kx**,

Tx,y — мандат, выданный **X** на использование **Y**,

Ax,y — аутентификатор, выданный **X** для **Y**, то есть информация, с помощью которой **Y** аутентифицирует **X**.

Операции (стрелки 1-5 на рис.2) могут быть записаны в формализованном виде:

1 — Клиент-Kerberos: **C, TGS**

2 — Kerberos-Клиенту: **{$K_{c,TGS}$}K_c**; **{$T_{c,TGS}$}K_{TGS}**

3 — Клиент-TGS: **{$A_{c,s}$}$K_{c,TGS}$**; **{$T_{c,TGS}$}K_{TGS}**

4 — TGS-Клиенту: **{$K_{c,s}$}$K_{c,TGS}$**; **{$T_{c,s}$}K_s**

5 — Клиент-Серверу: **{$T_{c,s}$}K_s**

Kerberos использует 2 типа удостоверений:

- Мандаты (для безопасной передачи Серверу данных о личности Клиента):

$Tc,s = S, \{C, A, v, K_{c,s}\}K_s$

Клиент не может расшифровать мандат, поскольку он не знает секретный ключ Ks, но он может предъявить его Се-ру, как док-во его аутентичности, т.е. прочитать либо изменить мандат ни Клиент, ни кто-либо иной не может.

- Аутентификаторы (это дополнительная информация, предъявляемая вместе с мандатом):

$Ac,s = \{C, t, Ключ\} K_{c,s}$

Клиент создает аутентификатор на каждый сеанс, Ключ - является просто ключом и необязательным дополнительным элементом сеанса и все эти данные шифруются общим ключом, известным Клиенту и Серверу: $K_{c,s}$. В отличие от мандата, аутентификатор используется только один раз