Доклад на тему "Протокол Kerberos".

Common information

discipline: Основы информационной безопасности
group: НПМбд-02-21
author: Ермолаев А.М.

Цель работы

Ознакомиться с основными концепциями сетевого протокола Kerberos и областями его применения.

Введение

Протокол Kerberos был создан более десяти лет назад в Массачусетском технологическом институте в рамках проекта Athena. Однако общедоступным этот протокол стал, начиная с версии 4. После того, как специалисты изучили новый протокол, авторы разработали и предложили очередную версию — Kerberos 5, которая была принята в качестве стандарта IETF. Требования реализации протокола изложены в документе RFC 1510, кроме того, в спецификации RFC 1964 описывается механизм и формат передачи жетонов безопасности в сообщениях Kerberos.

Протокол Kerberos предлагает механизм взаимной аутентификации клиента и сервера перед установлением связи между ними, причём в протоколе учтён тот факт, что начальный обмен информацией между клиентом и сервером происходит в незащищённой среде, а передаваемые пакеты могут быть перехвачены и модифицированы. Другими словами, протокол идеально подходит для применения в сети Интернет и аналогичных сетях.

Напомним разницу между смежными понятиями:

- **Идентификация** - процесс, когда пользователь предоставляет свои уникальные данные для системы, чтобы она могла его распознать.

  Например, ввод имени пользователя.

- **Аутентификация** - процесс проверки того, является ли предоставленная информация правильной.

  Это может включать проверку пароля или других факторов, таких как биометрия.

- **Авторизация** - определение прав доступа данного пользователя.

  Это включает определение, какие ресурсы или функции доступны данному пользователю.

Основная концепция

Основная концепция протокола Kerberos очень проста — если есть секрет, известный только двоим, то любой из его хранителей может с лёгкостью удостовериться, что имеет дело со своим напарником. Для этого ему достаточно проверить, знает ли его собеседник общий секрет.

Рассмотрим пример: на сервер приходят запросы, но есть клиент, запросы которого надо обрабатывать определеным образом. Данная задача может быть решена использованием протокола для взаимодействия между устройствами нарямую с использованием аутентификаторов - набора данных, симметрично зашифрованного секретным ключом (которые должны быть на стороне сервера и клиента). Аутентифактор должен иметь постоянно меняющуюся информацию, иначе злоумышленник может просто перехватить пакет и воспользоваться его содержимым для входа в систему. В качестве меняющейся информации можно использовать время.

Механизм работы может быть таковым:

1. Клиент посылает серверу сообщение, содержащее её имя в открытом виде и аутентификатор, зашифрованный с помощью секретного ключа. В протоколе аутентификации такое сообщение представляет собой структуру данных с двумя полями. Первое поле содержит имя, второе поле — текущее время.
2. Сервер получает сообщение, использует секретный ключ и дешифрует время отправления сообщения. Задача сильно упрощается в случае синхронизированного времени, ведь этом случае сервер может сравнить расшифрованное время с собственным временем, и, если время оказывается в пределах допустимого отклонения, то можно с большой долей уверенности предположить, что аутентификатор поступил именно от конкретного клиента.
3. Сервер шифрует время из сообщения клиента и включает его в собственное сообщение, которое отправляет клиенту.
4. Клиент получает ответ сервера, дешифрует его, и сравнивает полученное время со временем, которое было указано в исходном аутентификаторе. Если оно совпадает, то человек на другом конце смог расшифровать её сообщение, а значит, он знает секретный ключ.

Однако у данного подхода есть несколько недостатков:

- Ключ должен иметься у взаимодействующих устройств заранее.
- Как у клиента, так и у сервера должно храниться множество ключей.

Название протокола Kerberos говорит о том, как здесь решена проблема управления ключами. Кербер (или Цербер) — персонаж классической греческой мифологии. Этот свирепый пёс о трёх головах, по поверьям греков, охраняет врата подземного царства мёртвых. Трём головам Кербера в протоколе Kerberos соответствуют три участника безопасной связи: клиент, сервер и доверенный посредник между ними. Роль посредника здесь играет так называемый центр распределения ключей Key Distribution Center, KDC.

KDC

KDC представляет собой службу, работающую на физически защищённом сервере. Она ведёт базу данных с информацией об учётных записях всех главных абонентов безопасности своей области. Вместе с информацией о каждом абоненте безопасности в базе данных KDC сохраняется криптографический ключ, известный только этому абоненту и службе KDC. Этот ключ, который называют долговременным, используется для связи пользователя системы безопасности с центром распределения ключей. В большинстве практических реализаций протокола Kerberos долговременные ключи генерируются на основе пароля пользователя, указываемого при входе в систему.

Когда клиенту нужно обратиться к серверу, он прежде всего направляет запрос в центр KDC, который в ответ направляет каждому участнику предстоящего сеанса копии уникального сеансового ключа

(session key), действующие в течение короткого времени. Назначение этих ключей — проведение аутентификации клиента и сервера. Копия сеансового ключа, пересылаемая на сервер, шифруется с помощью долговременного ключа этого сервера, а направляемая клиенту — долговременного ключа данного клиента.

Теоретически, для выполнения функций доверенного посредника центру KDC достаточно направить сеансовые ключи непосредственно абонентам безопасности, как показано выше. Однако на практике реализовать такую схему чрезвычайно сложно. Прежде всего, серверу пришлось бы сохранять свою копию сеансового ключа в памяти до тех пор, пока клиент не свяжется с ним. А ведь сервер обслуживает не одного клиента, поэтому ему нужно хранить пароли всех клиентов, которые могут потребовать его внимания. В таких условиях управление ключами требует значительной затраты серверных ресурсов, что ограничивает масштабируемость системы. Нельзя забывать и о превратностях сетевого трафика. Они могут привести к тому, что запрос от клиента, уже получившего сеансовый пароль, поступит на сервер раньше, чем сообщение KDC с этим паролем. В результате серверу придется повременить с ответом до тех пор, пока он не получит свою копию сеансового пароля. То есть, нужно будет сохранить состояния сервера, а это накладывает на его ресурсы еще одно тяжкое бремя. Поэтому на практике применяется другая схема управления паролями, которая делает протокол Kerberos гораздо более эффективным. Ее описание приводится ниже.

Сеансовые мандаты

В ответ на запрос клиента, который намерен подключиться к серверу, служба KDC направляет обе копии сеансового ключа клиенту. Сообщение, предназначенное клиенту, шифруется посредством долговременного ключа, общего для данного клиента и KDC, а сеансовый ключ для сервера вместе с информацией о клиенте вкладывается в блок данных, получивший название сеансового мандата (session ticket). Затем сеансовый мандат целиком шифруется с помощью долговременного ключа, который знают только служба KDC и данный сервер. После этого вся ответственность за обработку мандата, несущего в себе шифрованный сеансовый ключ, возлагается на клиента, который должен доставить его на сервер.

Обратите внимание, что в данном случае функции службы KDC ограничиваются генерацией мандата. Ей больше не нужно следить за тем, все ли отправленные сообщения доставлены соответствующим адресатам. Даже если какое-нибудь из них попадет не туда, — ничего страшного не случится. Расшифровать клиентскую копию сеансового ключа может только тот, кто знает секретный долговременный ключ данного клиента, а чтобы прочесть содержимое сеансового мандата, нужен секретный код сервера.

Получив ответ KDC, клиент извлекает из него мандат и свою копию сеансового ключа, которые помещает в безопасное хранилище (оно располагается не на диске, а в оперативной памяти). Когда возникает необходимость связаться с сервером, клиент посылает ему сообщение, состоящее из мандата, который по-прежнему зашифрован с применением долговременного ключа этого сервера, и собственного аутентификатора, зашифрованного посредством сеансового ключа. Этот мандат в комбинации с аутентификатором как раз и составляет удостоверение, по которому сервер определяет "личность" клиента.

Сервер, получив "удостоверение личности" клиента, прежде всего с помощью своего секретного ключа расшифровывает сеансовый мандат и извлекает из него сеансовый ключ, который затем использует для дешифрования аутентификатора клиента. Если все проходит нормально, делается заключение, что

удостоверение клиента выдано доверенным посредником, то есть, службой KDC. Клиент может потребовать у сервера проведения взаимной аутентификации. В этом случае сервер с помощью своей копии сеансового ключа шифрует метку времени из аутентификатора клиента и в таком виде пересылает ее клиенту в качестве собственного аутентификатора.

Одно из достоинств применения сеансовых мандатов состоит в том, что серверу не нужно хранить сеансовые ключи для связи с клиентами. Они сохраняются в кэш-памяти удостоверений (credentials cache) клиента, который направляет мандат на сервер каждый раз, когда хочет связаться с ним. Сервер, со своей стороны, получив от клиента мандат, дешифрует его и извлекает сеансовый ключ. Когда надобность в этом ключе исчезает, сервер может просто стереть его из своей памяти.

Такой метод дает и еще одно преимущество: у клиента исчезает необходимость обращаться к центру KDC перед каждым сеансом связи с конкретным сервером. Сеансовые мандаты можно использовать многократно. На случай же их хищения устанавливается срок годности мандата, который KDC указывает в самой структуре данных. Это время определяется политикой Kerberos для конкретной области. Обычно срок годности мандатов не превышает восьми часов, то есть, стандартной продолжительности одного сеанса работы в сети. Когда пользователь отключается от нее, кэш-память удостоверений обнуляется и все сеансовые мандаты вместе с сеансовыми ключами уничтожаются.

Мандаты на выдачу мандатов

Как уже отмечалось, долговременный ключ пользователя генерируется на основе его пароля. При регистрации клиент Kerberos пропускает указанный пользователем пароль через функцию одностороннего хеширования (все реализации протокола Kerberos 5 должны обязательно поддерживать алгоритм DES-CBC-MD5, хотя могут применяться и другие алгоритмы). В результате генерируется криптографический ключ.

В центре KDC долговременные ключи хранятся в базе данных с учетными записями пользователей. Получив запрос от клиента Kerberos, KDC обращается в свою базу данных, находит в ней учетную запись нужного пользователя и извлекает из соответствующего ее поля долговременный соответствующий ключ.

Такой процесс — вычисление одной копии ключа по паролю и извлечение другой его копии из базы данных — выполняется всего лишь один раз за сеанс, когда пользователь входит в сеть впервые. Сразу же после получения пользовательского пароля и вычисления долговременного ключа клиент Kerberos рабочей станции запрашивает сеансовый мандат и сеансовый ключ, которые используются во всех последующих транзакциях с KDC на протяжении текущего сеанса работы в сети.

На запрос пользователя KDC отвечает специальным сеансовым мандатом для самого себя, так называемый мандат на выдачу мандатов (ticket-granting ticket), или мандат TGT. Как и обычный сеансовый мандат, мандат TGT содержит копию сеансового ключа для связи службы (в данном случае — центра KDC) с клиентом. В сообщение с мандатом TGT также включается копия сеансового ключа, с помощью которой клиент может связаться с KDC. Мандат TGT шифруется посредством долговременного ключа службы KDC, а клиентская копия сеансового ключа — с помощью долговременного ключа пользователя.

Получив ответ службы KDC на свой первоначальный запрос, клиент дешифрует свою копию сеансового ключа, используя для этого копию долговременного ключа пользователя из своей кэш-памяти. После этого долговременный ключ, полученный из пользовательского пароля, можно удалить из памяти,

поскольку он больше не понадобится: вся последующая связь с KDC будет шифроваться с помощью полученного сеансового ключа. Как и все другие сеансовые ключи, он имеет временный характер и действителен до истечения срока действия мандата TGT, либо до выхода пользователя из системы. По этой причине такой ключ называют сеансовым ключом регистрации (logon session key).

С точки зрения клиента мандат TGT почти ничем не отличается от обычного. Перед подключением к любой службе, клиент прежде всего обращается в кэш-память удостоверений и достает оттуда сеансовый мандат нужной службы. Если его нет, он начинает искать в этой же кэш-памяти мандат TGT. Найдя его, клиент извлекает оттуда же соответствующий сеансовый ключ регистрации и готовит с его помощью аутентификатор, который вместе с мандатом TGT высылает в центр KDC. Одновременно туда направляется запрос на сеансовый мандат для требуемой службы. Другими словами, организация безопасного доступа к KDC ничем не отличается от организации такого доступа к любой другой службе домена — она требует сеансового ключа, аутентификатора и мандата (в данном случае мандата TGT).

С точки же зрения службы KDC, мандаты TGT позволяют ускорить обработку запросов на получением мандатов, сэкономив несколько наносекунд на пересылке каждого из них. Центр распределения ключей KDC обращается к долговременному ключу пользователя только один раз, когда предоставляет клиенту первоначальный мандат на выдачу мандата. Во всех последующих транзакциях с этим клиентом центр KDC дешифрует мандаты TGT с помощью собственного долговременного ключа и извлекает из него сеансовый ключ регистрации, который использует для проверки подлинности аутентификатора клиента.

Область применения

- Kerberos является методом аутентификации по умолчанию в Windows и играет неотъемлемую роль в Windows Active Directory (AD).
- Kerberos доступен в Apple OS, FreeBSD, UNIX и Linux. 1
- Веб-приложения могут применять Kerberos в качестве метода аутентификации для клиентов, подключённых к домену, с помощью API.

Вывод

В рамках подготовки доклада работы я ознакомился с основными концециями сетевого протокола Kerberos и областями его применения.

Список литературы

- ftp.isi.edu/isi-pubs/rs-94-412.pdf — The Evolution of the Kerberos Authentication Service
- www.kerberos.org/docs/index.html - Official Kerberos Documentation
- web.mit.edu/kerberos/www/index.html — Kerberos: The Network Authentication Protocol