Описание формальной модели децентрализованной системы разграничения доступа

А. Ю . Чадов,
заместитель заведующего кафедрой
защиты информации МФТИ(НИУ)

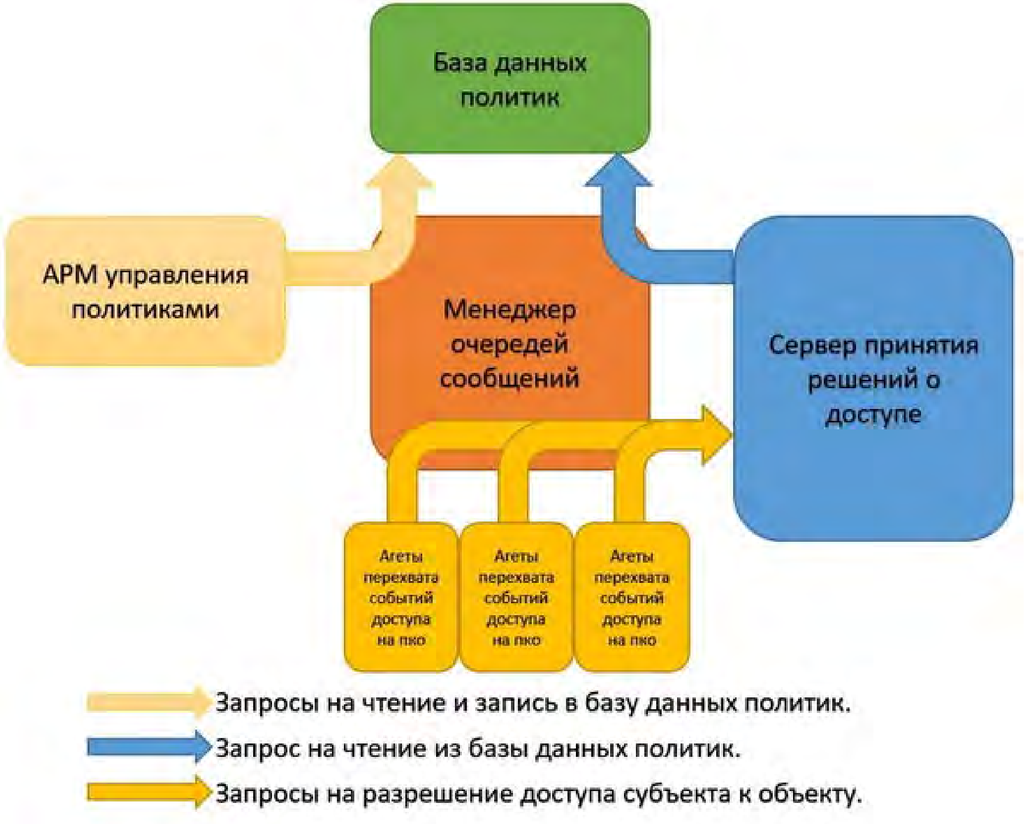
Требования к децентрализованной системе разграничения доступа

1. Система должна состоять отдельных модулей, общающихся друг с другом через определённое API. Среди этих модулей обязательно должны быть модуль принятия решений о доступе, модуль хранения политик доступа, модуль-перехватчик запросов доступа субъектов к объектам, модуль управления политиками, модуль транспортной системы для передачи сообщений.
2. Решения о доступе субъектов к объектам принимает центральный модуль, политики также хранятся централизованно.
3. Задержка, вносимая работой системы не должна ощущаться пользователем: она должна быть того же порядка, что и время обращения к диску или меньше.
4. Модули должны уметь проводить взаимную аутентификацию.
5. Данные, передаваемые в запросах модулей друг к другу должны быть защищены.

Пример того, как может выглядеть новая система

В данном примере, все модули общаются друг с другом через менеджер сообщений.

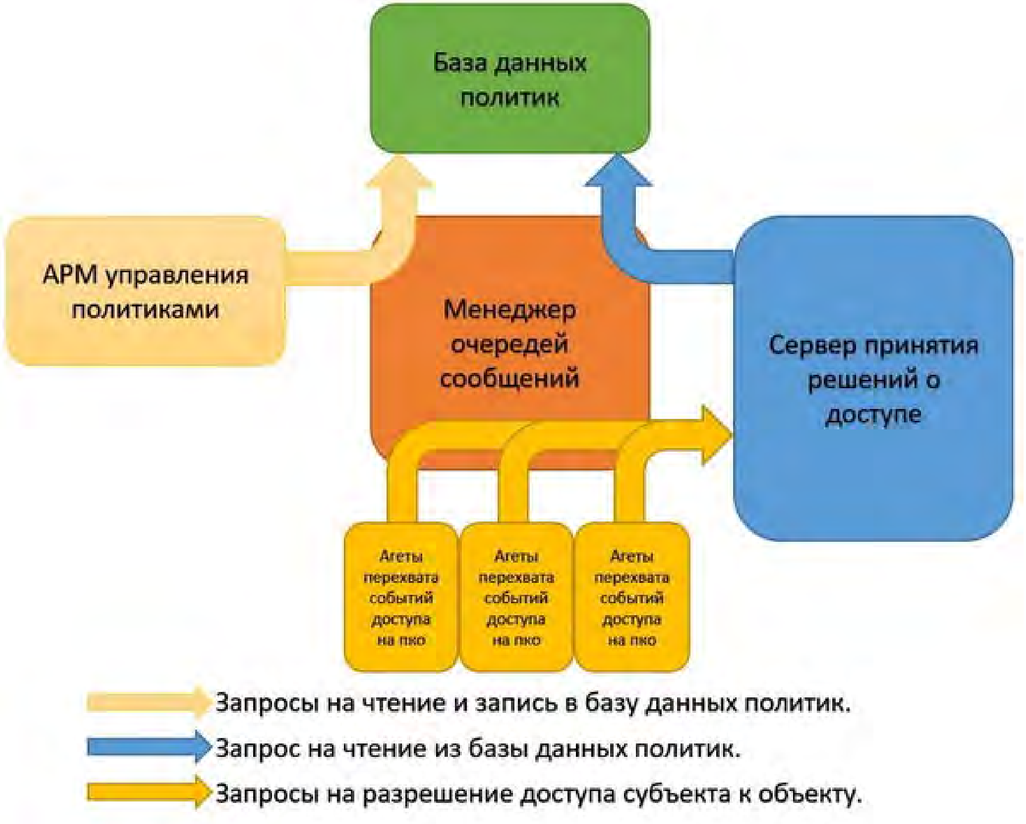
Работа строится следующим образом: сначала администратор через АРМ управления настраивает базу данных политик. Затем, в процессе работы, агенты, перехватывая события обращаются к серверу принятия решений. Он делает запрос в БД политик, и на основе полученных оттуда данных даёт агенту ответ.



Вербальная модель

Элементы модели:

- Сервер принятия решений о доступе
- Агент перехвата событий доступа на ПКО
- Менеджер сообщений
- База данных политик
- Модуль управления политиками



Формальная модель

Агент на ПКО:

- bool HandleInterrupt(string userName, string processName, string objectName, string accessType);
- AccessResult AskAccess(string userName, string pcName, string processName, string objectName, , string accessType, DateTime eventTime, Dictionary<string, string> params);

Сервер:

- AccessResult HandleAccess(string userName, string pcName, string processName, string objectName , string accessType, DateTime eventTime, Dictionary<string, string> params);
- PolicyInfo GetPolicy(string userName, string pcName, string processName, string objectName , string accessType);
- AccessResult AccessResolution(string userName, string pcName, string processName, string objectName, , string accessType, DateTime eventTime, Dictionary<string, string> params, PolicyInfo info);
- List<PolicyInfo> HandleGetPolicy(AuthInfo authInfo);
- List<PolicyInfo> GetPolicy();
- bool HandleSetPolicy(AuthInfo authInfo, PolicyInfo policyInfo);
- bool SetPolicy(PolicyInfo);

Модуль управления политиками

- List<PolicyInfo> AskPolicyInfo(AuthInfo authInfo);
- bool AskChangePolicy(AuthInfo authInfo, PolicyInfo policyInfo);

Процесс принятия решения о доступе

На ПКО:

- вызов HandleInterrupt,
- вызов AskAccess
- отправка запроса размером порядка ~кб по сети

На сервере:

- вызов AccessResult
- вызов GetPolicy(string userName, string pcName, string processName, string objectName , string accessType,);
- и обращение в БД и получение ~кб данных
- вызов AccessResolution
- отправка ответа размером порядка ~кб по сети

На ПКО:

- завершение прерывания соответственно полученному ответу

Процесс получения списка политик в модуле управления политиками

<u>В модуле управления:</u>

1) вызов AskPolicyInfo

2) передача порядка ~кб данных по сети

<u>На сервере:</u>

3) вызов HandleGetPolicy

4) вызов GetPolicy(string userName, ...); для проверки прав у запрашивающего и получение ~кб данных из БД

5) AccessResolution для проверки прав у запрашивающего

в зависимости от ответа AccessResolution, если прав недостаточно:

6) завершение HandleGetPolicy с результатом в виде пустого списка

7) отправка ответа размером порядка ~кб по сети

в зависимости от ответа AccessResolution, если прав хватает:

6) вызов GetPolicy() и получение данных порядка ~10 мб из БД

7) завершение HandleGetPolicy с заполненным списком в качестве результата

8) отправка ответа размером порядка ~10 мб по сети

Соответствие модели требованиям

1. Система должна состоять отдельных модулей, общающихся друг с другом через определённое API. Среди этих модулей обязательно должны быть модуль принятия решений о доступе, модуль хранения политик доступа, модуль-перехватчик запросов доступа субъектов к объектам, модуль управления политиками, модуль транспортной системы для передачи сообщений.
2. Решения о доступе субъектов к объектам принимает центральный модуль, политики также хранятся централизованно.
3. Задержка, вносимая работой системы не должна ощущаться пользователем: она должна быть того же порядка, что и время обращения к диску или меньше.
4. Модули должны уметь проводить взаимную аутентификацию.
5. Данные, передаваемые в запросах модулей друг к другу должны быть защищены.

Процесс принятия решения о доступе

На ПКО:

- вызов HandleInterrupt,
- вызов AskAccess
- **шифрование данных**
- **отправка запроса размером порядка ~кб по сети**

На сервере:

- **расшифрование данных**
- вызов AccessResult
- вызов GetPolicy(string userName, string pcName, string processName, string objectName , string accessType,);
- **и обращение в БД и получение ~кб данных**
- вызов AccessResolution
- **шифрование и подпись данных**
- **отправка ответа размером порядка ~кб по сети**

На ПКО:

- **расшифрование и проверка подписи**
- завершение прерывания соответственно полученному ответу

Спасибо за внимание!

А.Ю. Чадов,
заместитель заведующего кафедрой
защиты информации МФТИ(НИУ)