МАНДАТНЫЙ ПРИНЦИП КОНТРОЛЯ ДОСТУПА В ЗАЩИЩЕННЫХ ОС LINUX

Очилов Низомиддин Нажмиддин угли

Главный специалист, программист, Государственный центр тестирование при Кабинете Министров Республики Узбекистан, Узбекистан, г. Ташкент

Аннотация. Мандатная модель управления доступом, кроме дискреционной и ролевой, является основой реализации разграничительной политики доступа к ресурсам при защите информации ограниченного доступа. Для файловых систем, оно может расширять или заменять дискреционный контроль доступа и концепцию пользователей и групп.

Abstract. T The mandatory access control model, in addition to discretionary and role-based, is the basis for the implementation of the demarcation policy of access to resources while protecting restricted access information. For file systems, it can extend or replace discretionary access control and the concept of users and groups.

Ключевые слова: мандатная модель; низкий уровень секретности; средний уровень секретности; высокий уровень секретности; уровень конфиденциальности.

Keywords: mandatory model; low level of secrecy; medium level of secrecy; high level of secrecy; level of confidentiality.

Введение. Данная модель доступа практически не используется «в чистом виде», обычно на практике она дополняется элементами других моделей доступа [1,2].

Например, субъект «Пользователь № 2», имеющий допуск уровня «не секретно», не может получить доступ к объекту, имеющего метку «для служебного пользования». В то же время, субъект "Пользователь «№ 1» с допуском уровня «секретно», имеет право доступа к объекту с меткой «для служебного пользования» (рис. 1 и 2).

Рисунок 1. Схема возможностей классической модели Белла – ЛаПадулы

Мандатная система запрещает пользователю или процессу, обладающему определённым уровнем доверия, получать доступ к информации, процессам или устройствам более защищённого уровня [3]. Тем самым обеспечивается изоляция пользователей и процессов, как известных, так и неизвестных системе (неизвестная программа должна быть максимально лишена доверия, и её доступ к устройствам и файлам должен ограничиваться более строго).

Очевидно, что система, которая обеспечивает разделение данных и операций в компьютере, должна быть построена таким образом, чтобы её нельзя было «обойти». Она также должна давать возможность оценивать полезность и эффективность используемых правил и быть защищённой от постороннего вмешательства.

Основная часть. Для построения полтики была использована программа SELinux.

Для своей работы SELinux использует так называемые политики, которые состоят из набора конфигурационных файлов (пользователи SELinux, контексты стандартных файлов и прочее) и бинарных модулей политики. Данные файлы расположены в директории */etc/selinux/<имя политики>*. Активная политика, а также ее режим (*permissive/enforcing*) задаются в конфигурационном файле */etc/SELinux/config*, в формате:

SELINUX=permissive # либо enforcing, либо disabled

для полного отключения SELinux.

SELINUXTYPE=targeted-mls # имя политики

Режим работы SELinux можно задать и при загрузке ядра, тогда значение в конфигурационном файле будет проигнорировано. В реализованной системе предпочтение отдано флагам ядра, прописанным в загрузчик Lilo.

При инициализации системы SELinux, на основе конфигурационных файлов, файлов контекстов и модулей политик, создает кэш правил (временные правила) в директории с виртуальной файловой системой /SELinux. При любых изменениях в политике, даже при добавлении нового пользователя, происходит перекэширование, особенно это заметно при многомодульных политиках [4]. Таким образом, матрица доступов постоянно храниться в

оперативной памяти, что позволяет свести затраты процессора на авторизацию действий к минимуму.

На основе мандатного контроля доступа:

• Учетная запись пользователя имеет только один уровень доступа.

• В системе учетные записи пользователей по мандатному распределению делятся на три уровня: s0 (низкий уровень), s1 (средний уровень), s2 (высокий уровень секретности).

• У одного пользователя может быть до трех учетных записей в зависимости от уровня доступа. Например, s0 имеет одну учетную запись, s1 – две учетной записи, s2 – три учетной записи.

• Пользователь под учетной записью имеет право создавать, редактировать и читать файлы, уровень конфиденциальности которых равен уровню доступа данной учетной записи.

• Пользователь под учетной записью имеет право читать файлы, уровень конфиденциальности которых равен или ниже уровня доступа данной учетной записи.

• Во всех остальных случаях в доступе должно быть отказано.

• При создании нового файла ему будет присвоена метка конфиденциальности, соответствующая уровню доступа данной учетной записи пользователя.

• Для изменения меток конфиденциальности файлов в случае необходимости передачи информации на другой уровень безопасности должен привлекаться администратор безопасности.

• Обращение к приложениям и сервисам осуществляется на основе стандартной дискреционной модели Linux, предотвращающей доступ к системным приложениям и настройкам, так что пользователь не сможет изменить собственные или чужие уровни доступа или конфиденциальности, а также метки файлов.

Следует заметить, что для корректной работы системы необходимо, чтобы системный администратор в обязательном порядке связывал все учетные записи пользователей с SELinux.

Следует отметить, что данную операцию рекомендуется осуществлять *только* при первом запуске системы в режиме работы SELinux, т.к. иначе может измениться уровень конфиденциальности файлов на s0, что означает разрешенный доступ всем к данному ресурсу. При постоянной работе SELinux даже в режиме permissive, метки для новых файлов расставляются автоматически, согласно контексту и уровню доступа создающего файл процесса (например, если пользователь уровня s2 создаст какой-то файл, то файлу проставляется уровень s2).

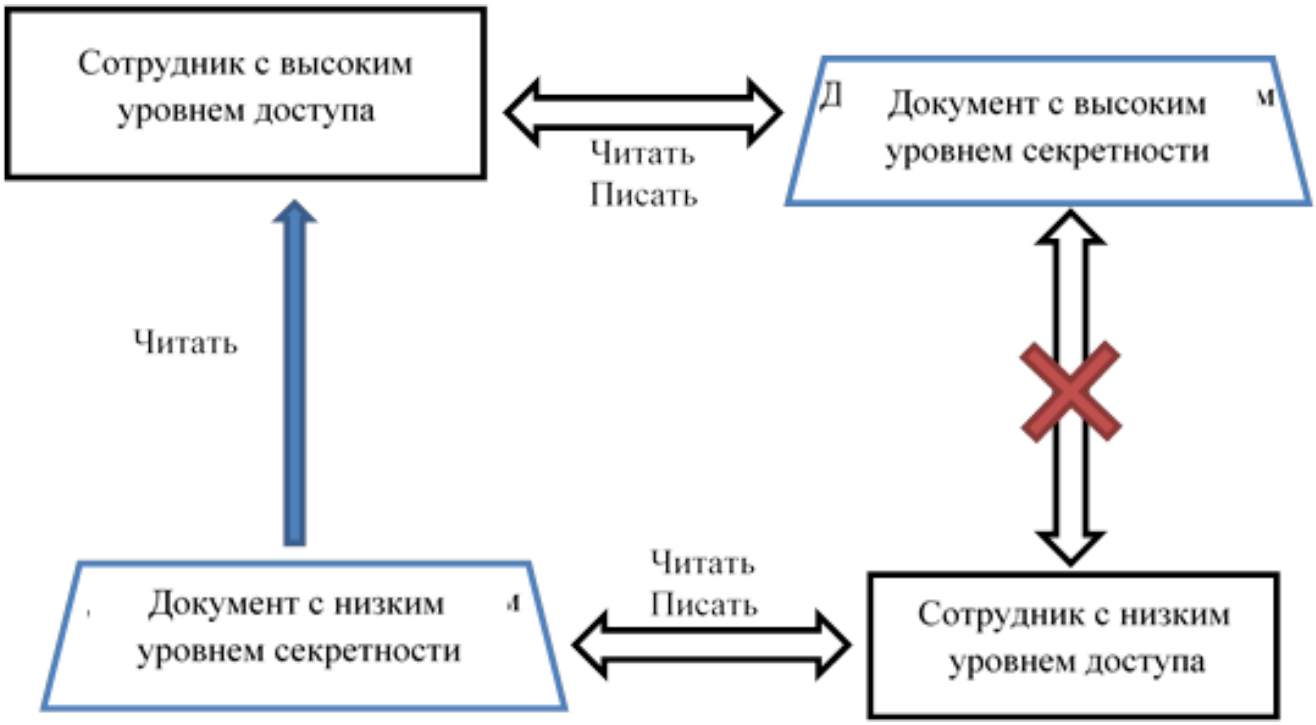

Рисунок 2. Схема реализации мандатной модели в ОС Linux

Заключение. Самое важное достоинство заключается в том, что пользователь не может полностью управлять доступом к ресурсам, которые он создаёт. Очевидно, что система, которая обеспечивает разделение данных и операций в компьютере, должна быть построена таким образом, чтобы её нельзя было «обойти». Она также должна давать возможность оценивать полезность и эффективность используемых правил и быть защищённой от постороннего вмешательства.

Список литературы:

1. Клейменов С.А., Мельников В.П., Петраков А.М. Администрирование в информационных системах; - Москва, 2008. - 272 с.

2. Хорев П. Б. Методы и средства защиты информации в компьютерных системах, М., Издательский центр "Академия", 2005 - 256с

3. Бармен Скотт. Разработка правил информационной безопасности. М.: Вильямс, 2002. - 208с.

4. Мартемьянов Ю. Ф., Яковлев А. В., Яковлев А. В. Операционные системы. Концепции построения и обеспечения безопасности; Горячая Линия - Телеком - , 2011. - 338 с.