

17.07.2012

#####



Руководство по безопасности

Copyright © 2006– 2012 Novell, Inc. и Сообщество. Все права защищены.

Разрешается копировать, распространять и/или изменять этот документ в соответствии с условиями лицензии GNU Free Documentation License, версии 1.2 или (на ваше усмотрение) версии 1.3; с инвариантным разделом, в котором указываются информация об авторском праве и лицензия. Копия лицензии версии 1.2 включена в раздел, озаглавленный «GNU Free Documentation License».

Для торговых марок Novell обратитесь к списку Novell Trademark и Service Mark <http://www.novell.com/company/legal/trademarks/tmlist.html>. Linux* — зарегистрированная торговая марка Линуса Торвальдса. Все другие торговые марки являются собственностью их владельцев. Знаки (®, ™ и другие) используются для обозначения торговых марок Novell; звездочкой (*) обозначены товарные марки третьих лиц.

Вся информация в этой книге была составлена с предельным вниманием к деталям. Однако, это не гарантирует абсолютной точности. Ни авторы из Novell, Inc., SUSE LINUX Products GmbH, ни переводчики, не несут ответственности за возможные ошибки и их последствия.

Содержание

Об этом руководстве	ix
----------------------------	-----------

1 Безопасность и конфиденциальность	1
1.1 Локальная и сетевая безопасность.	2
1.2 Общие советы и хитрости безопасности.	11
1.3 Использование Центрального адреса для сообщений о безопасности.	14

Часть I Аутентификация	17
-------------------------------	-----------

2 Авторизация с помощью PAM	19
2.1 Что такое PAM?.	19
2.2 Структура файла конфигурации PAM.	20
2.3 Конфигурация sshd с использованием PAM.	23
2.4 Настройка модулей PAM.	26
2.5 Настройка PAM при помощи pam-config.	28
2.6 Ручная настройка PAM.	29
2.7 Дальнейшие инструкции.	29

3 Использование NIS	31
3.1 Настройка NIS серверов.	31
3.2 Настройка NIS-клиентов.	38

4 LDAP — Сервис директорий	41
4.1 LDAP против NIS.	42
4.2 Структура дерева каталога LDAP.	43
4.3 Конфигурирование сервера LDAP с помощью YaST.	46
4.4 Конфигурирование клиента LDAP с помощью YaST.	56
4.5 Конфигурация пользователей и групп LDAP в YaST.	63
4.6 Просмотр дерева каталогов LDAP.	65
4.7 Конфигурация сервера LDAP вручную.	66
4.8 Управление данными в каталоге LDAP.	67

4.9	Дополнительная информация.	71
5	Active Directory Support	73
5.1	Integrating Linux and AD Environments.	73
5.2	Background Information for Linux AD Support.	74
5.3	Configuring a Linux Client for Active Directory.	78
5.4	Logging In to an AD Domain.	81
5.5	Changing Passwords.	83
6	Сетевая аутентификация при помощи Kerberos	85
6.1	Терминология Kerberos.	86
6.2	Как работает Kerberos.	87
6.3	Пользовательский взгляд на Kerberos.	90
6.4	Инсталляция и администрирование Kerberos.	91
6.5	Дополнительная информация.	113
7	Использование сканера отпечатков пальцев	115
7.1	Программы, поддерживающие биометрическую аутентификацию.	115
7.2	Управление биометрической аутентификацией через YaST.	116
Часть- II	Локальная безопасность	119
8	Настройка параметров безопасности с помощью YaST	121
8.1	<i>Обзор безопасности</i>	121
8.2	<i>Предопределенные настройки безопасности</i>	122
8.3	<i>Настройки пароля</i>	123
8.4	<i>Настройки загрузки</i>	124
8.5	<i>Настройки входа в систему</i>	124
8.6	<i>Добавление пользователя</i>	125
8.7	<i>Различные настройки</i>	125
9	Списки управления доступом в Linux	127
9.1	Традиционные файловые привилегии.	127
9.2	Преимущества ACL.	129
9.3	Определения.	130
9.4	Работа с ACL.	130
9.5	Поддержка ACL приложениями.	138
9.6	Дополнительная информация.	139

10	Шифрование файлов и разделов	141
10.1	Создание зашифрованной файловой системы при помощи YaST. . .	142
10.2	Использование зашифрованных домашних директорий.	146
10.3	Использование vi для шифрования отдельных текстовых ASCII фай- лов.	147
11	Обнаружение вторжений при помощи AIDE	149
11.1	Для чего нужна AIDE?.	149
11.2	Настройка базы данных AIDE.	150
11.3	Локальные проверки AIDE.	152
11.4	Проверка из независимой системы.	154
11.5	Для дальнейшей информации.	155
Часть-	Сетевая безопасность	157
III		
12	SSH: Безопасная работа в сети	159
12.1	ssh—Secure Shell.	159
12.2	scp—безопасное копирование.	161
12.3	sftp—безопасная передача файлов.	162
12.4	SSH демон (sshd).	162
12.5	Механизм аутентификации SSH.	164
12.6	Проброс порта.	167
12.7	Конфигурация SSH демона при помощи YaST.	168
12.8	Дополнительная информация.	169
13	Masquerading and Firewalls	171
13.1	Packet Filtering with iptables.	171
13.2	Masquerading Basics.	173
13.3	Firewalling Basics.	174
13.4	SuSEfirewall2.	174
13.5	For More Information.	181
14	Configuring VPN Server	183
14.1	Conceptual Overview.	183
14.2	Creating the Simplest VPN Example.	185
14.3	Setting Up Your VPN Server Using Certificate Authority.	187
14.4	Changing Nameservers in VPN.	193
14.5	KDE- and GNOME Applets For Clients.	194
14.6	For More Information.	196

15 Managing X.509 Certification	197
15.1 The Principles of Digital Certification.	197
15.2 YaST Modules for CA Management.	202
15.3 For More Information.	213

Часть- Ограничение привилегий с AppArmor **215**

IV

16 Introducing AppArmor	217
16.1 Background Information on AppArmor Profiling.	218

17 Getting Started	219
17.1 Installing AppArmor.	219
17.2 Enabling and Disabling AppArmor.	220
17.3 Choosing the Applications to Profile.	221
17.4 Building and Modifying Profiles.	222
17.5 Updating Your Profiles.	224

18 Immunizing Programs	225
18.1 Introducing the AppArmor Framework.	226
18.2 Determining Programs to Immunize.	228
18.3 Immunizing cron Jobs.	229
18.4 Immunizing Network Applications.	229

19 Profile Components and Syntax	235
19.1 Breaking a AppArmor Profile into Its Parts.	236
19.2 Profile Types.	239
19.3 <code>#include</code> Statements.	241
19.4 Capability Entries (POSIX.1e).	242
19.5 Network Access Control.	243
19.6 Paths and Globbing.	243
19.7 File Permission Access Modes.	246
19.8 Execute Modes.	249
19.9 Resource Limit Control.	254
19.10 Auditing Rules.	255

20 AppArmor Profile Repositories	257
20.1 Using the Local Repository.	257

21 Building and Managing Profiles with YaST	259
--	------------

21.1	Adding a Profile Using the Wizard.	261
21.2	Manually Adding a Profile.	268
21.3	Editing Profiles.	269
21.4	Deleting a Profile.	274
21.5	Updating Profiles from Log Entries.	275
21.6	Managing AppArmor.	275
22	Building Profiles from the Command Line	279
22.1	Checking the AppArmor Module Status.	279
22.2	Building AppArmor Profiles.	281
22.3	Adding or Creating an AppArmor Profile.	281
22.4	Editing an AppArmor Profile.	282
22.5	Deleting an AppArmor Profile.	282
22.6	Two Methods of Profiling.	282
22.7	Important Filenames and Directories.	303
23	Profiling Your Web Applications Using ChangeHat	305
23.1	Apache ChangeHat.	306
23.2	Configuring Apache for mod_apparmor.	312
24	Confining Users with pam_apparmor	317
25	Managing Profiled Applications	319
25.1	Reacting to Security Event Rejections.	319
25.2	Maintaining Your Security Profiles.	320
26	Support	323
26.1	Updating AppArmor Online.	323
26.2	Using the Man Pages.	323
26.3	For More Information.	325
26.4	Troubleshooting.	325
26.5	Reporting Bugs for AppArmor.	332
27	AppArmor Glossary	335
A	Лицензии GNU	339
A.1	Универсальная Общественная Лицензия GNU (GNU General Public License).	339
A.2	GNU Free Documentation License.	343

Об этом руководстве

Это руководство представляет основные концепции безопасности системы в . Оно охватывает обширную часть документации об аутентификационных механизмах, доступных в GNU/Linux, таких как NIS или LDAP. В руководстве также освещаются такие вопросы безопасности как списки контроля доступа, шифрование и обнаружение вторжения. В третьей части "Сетевая безопасность" Вы узнаете как обезопасить свой компьютер с помощью брандмауера, маскардинга, а так же как настроить виртуальную частную сеть (VPN). Это руководство рассказывает также об использовании ПО для обеспечения безопасности доступного в системе: AppArmor (с помощью него Вы можете настраивать доступ для ПО к файлам).

Многие главы в этом руководстве содержат ссылки на ресурсы с дополнительной документацией. Все эти ресурсы и оригинальная документация, доступны как в системе, так и в сети.

Документацию, доступную для Вашего продукта, можно найти на странице: <http://www.suse.com/documentation> , а так же в следующих разделах.

1 Доступная документация

Мы предоставляем HTML и PDF-версии наших книг на разных языках. Для данного дистрибутива доступны следующие руководства для пользователей и администраторов:

Вступление (↑Вступление)

Руководство шаг за шагом проведет Вас через установку с DVD или из ISO-образа, даст краткое введение в окружения рабочего стола GNOME и KDE, включая некоторые ключевые приложения. Также познакомит с LibreOffice и его модулями для создания текста со сложным форматированием, работы с электронными таблицами или создания графики и презентаций.

Содержание (↑Содержание)

Даёт общее понимание работы , затрагивая задачи продвинутого системного администрирования. Его материал предназначен в первую очередь для системных администраторов и домашних пользователей, обладающих базовыми навыками администрирования. Содержит детальную информацию о про-

двинутых вариантах развертывания, администрирования, взаимодействия ключевых компонентов и настройке различных сетевых и файловых служб .

Руководство по безопасности (стр. i)

Описываются основные понятия системы безопасности, охватывающей как локальные, так и сетевые аспекты. Показывается, как использовать такие утилиты для обеспечения сетевой безопасности, как AppArmor (которая позволяет определить к каким файлам заданная программа будет иметь доступ на запись, чтение или выполнение) или система аудита, которая тщательно собирает информацию о событиях, так или иначе связанных с обеспечением надлежащего уровня безопасности системы.

System Analysis and Tuning Guide (↑System Analysis and Tuning Guide)

Руководство администратора по обнаружению проблем, их разрешение и оптимизация работы. В нем найдется информация о том, как проверить и оптимизировать работу системы с помощью специальных инструментов, эффективно управлять ее ресурсами. Также в нем содержится обзор общих проблем и их решений, а также дополнительные справочные материалы и обзор доступных ресурсов.

Виртуализация с KVM (↑Виртуализация с KVM)

Данное руководство предлагает краткое описание настройки и управления системой виртуализации на базе KVM (Kernel-based Virtual Machine) в . Также показывается, как управлять VM Guest с помощью libvirt и QEMU.

Большинство HTML-версий руководств в установленной системе можно найти по адресу `/usr/share/doc/manual` или в справочном центре используемого окружения рабочего стола. Последние обновления документации доступны по адресу <http://www.novell.com/documentation>, где можно загрузить в PDF или HTML-версии руководств для конкретного продукта.

2 Обратная связь

Некоторые из доступных каналов обратной связи:

Ошибки и запросы об улучшениях

Чтобы сообщить об ошибке или отправить запрос об улучшении, пожалуйста, используйте <https://bugzilla.novell.com/>. Чтобы сооб-

шить о найденной ошибке в документации отправьте отчет для компонента *Documentation* (Документация) соответствующего продукта.

Если вы плохо знакомы с Bugzilla, то для вас могут оказаться полезными эти статьи:

- http://ru.opensuse.org/opensUSE:Сообщить_об_ошибке
- http://ru.opensuse.org/opensUSE:Сообщить_об_ошибке_FAQ

Комментарии пользователей

Мы хотим услышать ваши комментарии и предложения об этом руководстве и другой документации, поставляемой с данным продуктом. Используйте поле ввода в нижней части на каждой страницы онлайн-документации или перейдите по ссылке <http://www.novell.com/documentation/feedback.html> и оставьте свой комментарий.

3 Условные обозначения

В данном руководстве используются следующие типографские соглашения:

- `/etc/passwd`: имена каталогов и файлов
- *заполнитель*: замена *заполнитель* на фактическое значение
- `PATH`: переменная окружения `PATH`
- `ls, --help`: команды, опции и параметры
- `user`: пользователи или группы
- `, + F1`: клавиша или клавиатурная комбинация; названия клавиш показаны в верхнем регистре, как на клавиатуре
- *Файл*, *Файл* > *Сохранить как*: пункты меню, кнопки
- *Танцующие пингвины* (Глава *Пингвины*, ↑Другое руководство): это ссылка на главу в другом руководстве.

4 О создании этого руководства

Эта книга была создана в Novdoc, основан на DocBook (смотрите <http://www.docbook.org>). Исходные XML-файлы проверяются программой `xmllint`, обрабатываются `xsltproc` и преобразовываются в XSL-FO с использованием специализированной версии таблиц стилей Нормана Уолша (Norman Walsh). Конечный PDF-файл отформатирован через XEP от RenderX. Инструменты с открытым исходным кодом и среда, используемая для создания этого руководства, доступны в пакете `susedoc`, поставляемом в составе .

5 Исходный код

Исходный код находится в открытом доступе. По следующему адресу доступны ссылки на загрузку и дополнительная информация http://ru.opensuse.org/Исходный_код.

6 Благодарности

Разработчики Linux сотрудничают с огромным числом добровольцев по всему миру, чтобы способствовать развитию Linux. Мы благодарны им за приложенные усилия — этот дистрибутив не существовал бы без них. Кроме того, мы благодарим Фрэнка Заппа (Frank Zappa) и Павара (Pawar). Особая благодарность, конечно же, выражается Линусу Торвальдсу (Linus Torvalds).

Спасибо всем кто принял участие в подготовке перевода данного руководства:

Александр Иванов
hrafn@hrafn.me

Александр Наумов
alexander_naumov@opensuse.org

Андрей Карепин
egdfree@opensuse.org

Антон Черкасов
linux-oid@opensuse.org

Борис Вассерман
natabor2004@gmail.com

Виктор Дубинюк
victor.dubiniuk@gmail.com

Динар Валеев
k0da@opensuse.org

Павел Астахов
pastakhov@yandex.ru

Have a lot of fun!

Ваша команда SUSE

Безопасность и конфиденциальность

Одной из главных характеристик UNIX/Linux систем является возможность обслуживать несколько пользователей (многопользовательские системы), а так же предоставлять им возможность выполнять несколько задач одновременно (многозадачность). Более того, операционная система прозрачна для сети. Пользователи часто даже не знают, находятся ли используемые ими данные и программы на локальном компьютере или доступны через сеть.

Так как системы многопользовательские, данные разных пользователей должны храниться отдельно, а так же должна быть обеспечена их безопасность и приватность. Безопасность данных была важна даже еще до того, как появилась возможность объединять компьютеры через сеть. Намного большее беспокойство вызывает несанкционированный доступ к информации, нежели ее потеря или даже поломка носителя информации (например жесткого диска).

Этот раздел фокусируется на проблемах конфиденциальности и на способах защиты личных пользовательских данных. Концепция безопасности включает в себя регулярно обновляемую, рабочую и проверяемую резервную копию, хранящуюся в безопасном месте. Без этих мер Вам придется потратить много времени, возвращая информацию— притом не только в случае поломки или дефекта носителя, но и в случае несанкционированного доступа.

1.1 Локальная и сетевая безопасность

Возможны несколько способов доступа к данным:

- личное общение с людьми, имеющим нужную информацию или доступ к данным на компьютере
- напрямую, путем физического доступа к терминалу
- через последовательный интерфейс
- используя сетевое соединение

В каждом из этих способов пользователь должен пройти проверку подлинности перед тем, как получит доступ к запрашиваемым им ресурсам или данным. На веб-сервер может накладываться меньше ограничений, но в любом случае Вы вряд ли захотите, чтобы он давал анонимному пользователю доступ к Вашим личным данным.

В приведенном выше списке самый первый способ использует наибольшее количество межличностного взаимодействия (например, когда Вы связываетесь с сотрудником банка, Вам требуется подтвердить, что Вы являетесь владельцем счета). Вас попросят предоставить подпись, ПИН-код или пароль, чтобы удостовериться, что Вы тот, за кого себя выдаете. В некоторых случаях простого упоминания уже известных Вам фрагментов и кусочков информации достаточно, чтобы расположить к себе осведомленного человека и вывести ее целиком. Ничего не подозревающую жертву можно подвести постепенному раскрытию все больших объемов информации. Среди хакеров это называется *социальной инженерией*. Единственной защитой от этого является подготовка людей и осознанное распространение информации. Перед взломом компьютерной системы атакующие часто пытаются использовать секретарей, обслуживающий персонал компании или даже членов семей ее сотрудников. Во многих случаях о такой основанной на социальной инженерии атаке становится известно гораздо позже.

Человек, желающий получить несанкционированный доступ к Вашим данным, может также попытаться использовать традиционный способ прямого доступа к Вашему компьютеру. Из этого следует, что система должна быть защищена от любых манипуляций, и никто не мог удалить, изменить или повредить ее компо-

ненты. Это относится и к резервным копиям, и даже к сетевым кабелям и кабелям питания. Также необходимо обезопасить этап загрузки, поскольку существует несколько широко распространенных клавиатурных комбинаций, способных вызвать нестандартное поведение. Защитите себя от этого, установив пароли на BIOS и системный загрузчик.

Устройства, подключаемые к последовательным портам повсеместно используются до сих пор. В отличие от сетевых интерфейсов, они не используют сетевых протоколов для взаимодействия с системой к которой подключены. Для обмена данными между устройствами используется обычный кабель или инфракрасный порт, по которому в обе стороны передаются обычные символы. Кабель является самым слабым звеном такой системы: передаваемые по нему данные могут быть перехвачены путем подключения к этому кабелю обычного старого принтера. Кроме принтера для перехвата можно использовать все что угодно, в зависимости от технической оснащённости атакующего.

Чтение локального файла требует дополнительных правил доступа в сравнении с открытием сетевого подключения к другому компьютеру. Существует четкое разграничение между локальной и сетевой безопасностью и черта проходит там, где данные инкапсулируются в пакеты для отправки куда-либо.

1.1.1 Локальная безопасность

Локальная безопасность начинается с физического окружения того места, где установлен Ваш компьютер. Устанавливайте свою машину в месте, безопасность которого соответствует Вашим потребностям и ожиданиям. Главная задача локальной безопасности — разделение пользователей таким образом, чтобы ни один из них не смог использовать привилегии или идентификационные данные другого. Это наиболее общее правило, которое необходимо соблюдать, и оно особенно касается пользователя `root`, обладающего привилегиями системного администратора. `root` может работать от имени любого другого локального пользователя не вводя его пароля и читать любые локальные файлы.

1.1.1.1 Пароли

Linux-система не хранит пароли в текстовом виде, используя для их валидации простое сравнение введенного текста с сохраненным образцом. Если бы это было так, то все аккаунты системы были бы под угрозой при получении кем-либо доступа к соответствующему файлу. Именно поэтому все сохраненные пароли

зашифрованы, перед сравнением введенный текст также шифруется, после чего сравниваются две зашифрованных строки. Этот метод будет более безопасным только в том случае, если зашифрованный пароль не может быть восстановлен в оригинальную текстовую строку.

Это достигается при помощи алгоритма особого рода, так же называемого *алгоритмом с потайным ходом*, поскольку он работает только в одном направлении. Злоумышленник, заполучив Ваш пароль в зашифрованном виде, не сможет восстановить его используя этот алгоритм повторно. Вместо этого ему придется перебирать все возможные комбинации символов до тех пор, пока он не получит комбинацию, которая после шифрования совпадет с зашифрованным паролем. При длине пароля в восемь символов ему придется перебрать значительное количество таких комбинаций.

В семидесятых годах прошлого века утверждалось, что данный метод безопаснее других, поскольку алгоритм работал довольно медленно и на шифрование одного пароля требовалось несколько секунд. Однако в настоящее время компьютеры стали достаточно мощными для шифрования сотен тысяч или даже миллионов паролей в секунду. Поэтому даже зашифрованные пароли не должны быть доступны обычному пользователю (обычные пользователи не должны иметь доступа к файлу `/etc/shadow`). Еще более важным является невозможность угадать пароль в случае, если файл паролей доступен по ошибке. Следовательно, «перевод» пароля вида «tantalize» в «t@nt@1lz3» не повышает уровня безопасности.

Недостаточно заменить нескольких букв в слове похожими на них цифрами (наподобие записи «tantalize» как «t@nt@1lz3»). Программы, использующие словари для взлома паролей, также могут производить такие замены. Гораздо лучший способ — составить слово не имеющее смысла ни для кого, кроме Вас лично, например из первых букв предложения или названия книги. Возьмем к примеру фразу о книге «Имя Розы» Умберто Эко - «The Name of the Rose» by Umberto Eco. В результате получится следующий безопасный пароль: «TNotRbUE9». Напротив, пароли наподобие «beerbuddy» или «jasmine76» могут быть легко подобра ны кем-либо, кто знает Вас хоть немного.

1.1.1.2 Процесс загрузки

Настройте свою систему так, чтобы ее нельзя было загрузить с дискеты или компакт-диска - либо физически удалив эти устройства, либо разрешив в BIOS загрузку только с жесткого диска и установив пароль на BIOS. Linux-системы

обычно запускаются при помощи системного загрузчика, позволяющего передать дополнительные параметры загружаемому ядру. Заблокируйте эту возможность для посторонних, установив дополнительный пароль в файле `/boot/grub/menu.lst` (Глава 6, *The Boot Loader GRUB* (↑Содержание)). Это необходимо для безопасности вашей системы: поскольку ядро стартует с привилегиями `root`, существует возможность получить привилегии `root` при старте системы.

1.1.1.3 Права доступа к файлам

Основным правилом является установка максимально строгих привилегий достаточных для выполнения конкретной задачи. Например, нет никакой необходимости использовать пользователя `root` для чтения и отправки электронной почты. Любой баг почтовой программы можно использовать для атаки, которая будет произведена с правами доступа этой программы. Изложенное выше правило сводит к минимуму потенциальный урон, нанесенный системе подобной атакой.

Права доступа ко всем файлам входящим в состав дистрибутива были выбраны очень тщательно. При установке дополнительного программного обеспечения или других файлов системный администратор должен быть очень внимателен, особенно при установке битов доступа. Опытные системные администраторы уделяющие внимание безопасности системы, всегда используют опцию `-l` команды `ls` для получения дополнительной информации о файлах и обнаружении неверно заданных привилегий. Неправильно установленные биты доступа не просто позволяют изменять или удалять файлы. Модифицированные файлы могут быть выполнены пользователем `root` или, если это конфигурационные файлы, программа может использовать эти файлы с привилегиями `root`, что значительно увеличивает уязвимость системы. Подобные атаки носят название "кукушкино яйцо", поскольку программа (яйцо) выполняется (появляется на свет из скорлупы) у другого пользователя (птицы), в точности как кукушка дурачит остальных птиц, подбрасывая им свои яйца.

Система включает в себя файлы `permissions`, `permissions.easy`, `permissions.secure` и `permissions.paranoid`, расположенные в директории `/etc`. Эти файлы нужны для задания особых привилегий, таких как разрешение всем пользователям изменять директории и установка бита `setuser ID` (программы с битом `setuser ID` запускаются не с привилегиями пользователя, а с привилегиями владельца данной программы, в большинстве случаев им является `root`). Администратор системы может использовать файл `/etc/permissions.local` для установки собственных настроек.

Чтобы задать какие из вышеперечисленных файлов используются программы конфигурации для установки битов доступа, выберите пункт *Локальная безопасность* в секции *Безопасность и пользователи YaST*. Для получения дополнительной информации обратитесь к комментариям `/etc/permissions` или справочному руководству по команде `chmod` (`man chmod`).

1.1.1.4 Баги переполнения буфера и строки формата

Следует быть особенно осторожным в случае, если программе необходимо обрабатывать данные, доступные пользователю для изменения. Это правило адресовано в первую очередь разработчику программы, а не рядовым пользователям. Разработчик должен быть уверен, что его приложение корректно обрабатывает данные и не пытается их сохранить в участки памяти недостаточного размера. Помимо этого, приложение должно проверять целостность данных, используя определенные для этих целей интерфейсы.

Переполнение буфера может произойти, если реальный размер буфера памяти не учитывается при записи в буфер. Возможна ситуация, когда эти данные (созданные пользователем) используют требуют больше памяти, чем ее доступно в буфере. В результате при некоторых обстоятельствах вместо обычной обработки введенных пользователем данных их часть, записанная за пределы буфера, может быть выполнена. Это может быть использовано для выполнения кода, заданного пользователем, а не программистом. Такой баг может иметь очень серьезные последствия, особенно при выполнении программ с особыми правами доступа (см Раздел 1.1.1.3, «Права доступа к файлам» (стр. 5)).

Баги формата строки немного отличаются, но суть их по-прежнему в том, что ввод пользователя способен ввести программу в заблуждение. В большинстве случаев эти ошибки могут быть использованы для программ, обладающими особыми привилегиями — `setuid` и `setgid` — таким образом, для защиты системы и данных Вы можете отменить соответствующие привилегии для программ. Повторимся, политика предоставления наименьших возможных привилегий является наилучшей (см. Раздел 1.1.1.3, «Права доступа к файлам» (стр. 5)).

Поскольку баги переполнения буфера и строки формата относятся к обработке пользовательских данных, они могут быть использованы не только пользователем, имеющим локальный доступ. Множество известных багов можно было воспроизвести как локально, так и по сети (например, используя особым образом

сформированную ссылку). Соответственно, баги строки формата и переполнения буфера следует классифицировать как относящиеся и к локальной и к сетевой безопасности.

1.1.1.5 Вирусы

Вопреки распространенному мнению, существуют вирусы работающие под Linux. Однако, известные вирусы были написаны своими авторами в качестве *доказательства концепции*, что техника работает по назначению. До настоящего времени ни один из этих вирусов не был обнаружен в *дикой природе*.

Вирусы не могут существовать и распространяться без своего хозяина. Хозяином может быть программа или важное хранилище системы, такое как основная загрузочная запись, для записи в которую код вируса должен иметь соответствующие привилегии. Благодаря своим многопользовательским возможностям, Linux позволяет ограничить доступ на запись к определенным файлам (это особенно важно для файлов системы). Таким образом, если Вы имеете привычку работать с привилегиями `root`, вы увеличиваете шансы системы быть зараженной вирусом. И напротив, следование упомянутому выше принципу наименьших возможных привилегий, значительно снижает такую возможность.

Кроме того, никогда не торопитесь запускать программу, скачанную с первого попавшегося интернет-сайта. RPM пакеты содержат криптографическую подпись, как знак того, что они были собраны с должным вниманием. Наличие вирусов является типичным признаком того, что администратор или пользователь не обладают достаточными знаниями о безопасности, поскольку им удалось подвергнуть риску систему, изначально спроектированную с высоким уровнем безопасности.

Не следует путать вирусы с червями, принадлежащими исключительно к миру сетей. Отличие состоит в том, что червь не нужен хозяин для распространения.

1.1.2 Сетевая безопасность

Сетевая безопасность важна для защиты от атак, источник которых находится за пределами сети. Типичная процедура входа в систему, требующая для авторизации имени пользователя и пароля, все так же является объектом локальной безопасности. В частном случае удаленного входа в систему следует отмечать два

аспекта безопасности. Все, что происходит до входа в систему, относится к сетевой безопасности, а все, что происходит после — к локальной.

1.1.2.1 Система X Window и авторизация в X

Как упоминалось в самом начале, сетевая прозрачность - одна из центральных характеристик систем семейства UNIX. X, оконная система UNIX систем, пользуется этой особенностью для достижения довольно внушительного эффекта. С помощью X абсолютно несложно подключиться к удаленному компьютеру и запустить программу с графическим интерфейсом, которая затем будет отправлена по сети и отображена на Вашем компьютере.

Когда клиенту X требуется отобразить удаленно используя X сервер, последний должен защищать свои ресурсы (экран) от несанкционированного доступа. Точнее, клиентская программа должна обладать строго заданными правами доступа. С помощью системы X Window это может быть реализовано двумя способами: доступ, основанный на хосте или доступ, основанный на cookie. Первый полагается на IP адрес хоста, где запущена клиентская программа. Для управления им используется программа `xhost`. `xhost` сохраняет IP адреса имеющих доступ клиентов, в базу данных X сервера. Однако полагаться на IP адреса для авторизации не слишком безопасно. К примеру, если на клиентском компьютере работает еще один пользователь, то он также получит доступ к X серверу — точно также, как любой, кто присвоит себе IP этого компьютера. Вследствие изложенных выше недостатков этот метод не будет детально описан в данном руководстве, однако Вы можете воспользоваться `man xhost` для получения более подробной информации.

В случае управления доступом при помощи cookie, генерируется строка символов, известная только X серверу и авторизованному пользователю, подобно любой идентификационной карте. Эта cookie сохраняется после логина в файле `.Xauthority` в домашнем каталоге пользователя и доступна любому X клиенту, желающему воспользоваться X сервером для отображения окна. Файл `.Xauthority` может быть исследован пользователем при помощи утилиты `xauth`. Если Вы переименуете `.Xauthority`, или удалите этот файл из Вашей домашней директории, то больше не сможете открыть новых окон или X клиентов.

SSH (безопасная оболочка) может быть использована для полного шифрования сетевого соединения и прозрачной переадресации его X серверу, без вмешательства пользователя в механизм шифрования. Также это называется X forwarding.

X forwarding достигается имитацией X сервера на серверной стороне и установкой переменной DISPLAY для оболочки на удаленном компьютере. Больше информации о SSH можно найти в главе Глава 12, *SSH: Безопасная работа в сети* (стр. 159).

ПРЕДУПРЕЖДЕНИЕ

Если Вы не считаете безопасным удаленный компьютер, к которому подключаетесь, не используйте X forwarding. Будучи включенным, он позволяет авторизоваться используя Ваше SSH соединения для проникновения на Ваш X сервер и произвести различные манипуляции (чтение или сниффинг, например того, что Вы печатаете на клавиатуре).

1.1.2.2 Баги переполнения буфера и строки формата

Как обсуждалось в Раздел 1.1.1.4, «Баги переполнения буфера и строки формата» (стр. 6), баги переполнения буфера и строки формата должны классифицироваться, как имеющие отношение и к локальной, и к сетевой безопасности. Подобно локальному варианту, переполнение буфера в сетевых программах преимущественно может быть использовано злоумышленником для получения привилегий `root`. Даже если эта попытка не будет успешна, атакующий может использовать баг для получения доступа к непривилегированному локальному аккаунту и дальнейшего поиска других уязвимостей, которые возможно присутствуют в системе.

Использование багов переполнения буфера и строки формата по сети — наиболее часто встречающаяся разновидность удаленных атак. Эксплоиты — программы, использующие вновь найденные уязвимости — часто публикуются в почтовых рассылках посвященных безопасности. Они могут быть использованы для попытки проникновения в систему без знания подробностей конкретной уязвимости. Опыт многих лет доказал, что доступность кода exploits помогла операционным системам стать более безопасными, очевидно благодаря тому, что именно они принуждают создателей операционных систем исправлять проблемы в своих программах. В свободном программном обеспечении каждый, кто имеет доступ к исходным кодам (поставляется со всеми доступными исходными кодами) и каждый, кто найдет уязвимость и код, ее эксплуатирующий, могут предложить патчи для исправления соответствующего бага.

1.1.2.3 Отказ в обслуживании

Целью атаки типа отказ в обслуживании (DoS) является блокирование программы-сервера или всей системы, что может быть достигнуто разными путями: повышенной нагрузкой на сервер, удержанием его в состоянии занятости путем отправки пакетов с мусором или использованием переполнения его буфера. Чаще всего единственной целью DoS атаки является прекращение обслуживания сервером. Тем не менее, поскольку сервер становится недоступным, появляется возможность также использовать другие виды атак: «человек посередине» (прослушивание, перехват TCP соединения, спуфинг) и подмена DNS.

1.1.2.4 «Человек посередине»

Любая сетевая атака, при которой атакующий находится между взаимодействующими устройствами, называется атакой типа «человек посередине» (a man-in-the-middle). Общим между ними является то, что жертва атаки обычно ничего не подозревает. Существует много вариантов — например, атакующий может принять запрос на соединение и затем перенаправить его адресату. В результате жертва непреднамеренно устанавливает соединение не с тем компьютером, поскольку другая сторона ведет себя как компьютер, с которым изначально предполагалось соединение.

Простейшая форма атаки «человек посередине» называется *сниффер* (атакующий «просто» прослушивает проходящий сетевой трафик). В качестве более сложного вида, атаки «человек посередине» может попробовать перехватить уже установленное соединение (это называется *hijacking*). Для этого атакующему необходимо некоторое время анализировать пакеты некоторое время, чтобы ему удалось предсказать последовательность чисел TCP принадлежащих соединению. Когда атакующий окончательно берет на себя роль целевой машины, жертвы узнают это, поскольку они получают сообщение говорящее о прекращении соединения из-за ошибки. Факт существования не защищенных от перехвата путем шифрования протоколов (производящих только обычную процедуру авторизации при установлении соединения) только упрощает задачу злоумышленника.

Спуфинг (spoofing) это атака при которой в пакетах подменяются первоначальные данные, обычно IP адрес. Наиболее активные формы атаки полагаются на рассылку таких поддельных пакетов (на Linux-машине, это может быть сделано только суперпользователем (`root`)).

Многие из упомянутых атак используются в сочетании с DoS. Если атакующий видит возможность внезапно положить определенный хост, даже на короткий промежуток времени, это может помочь ему в проведении активной атаки, т.к. этот хост не будет вмешиваться в трафик некоторое время.

1.1.2.5 Подмена DNS

Подмена DNS (также называемая атакой Каминского) производится путем повреждения целостности данных в кеше DNS сервера. Отвечая на запросы DNS сервера подмененными пакетами, можно заставить его послать определенные данные жертве, которая запрашивает информацию с этого сервера. Многие из серверов доверяют другим хостам, основываясь на именах или IP адресах. Атакующему необходимы знания о структуре доверительных отношений между хостами, для того, чтобы замаскироваться под авторитетный хост. Обычно для получения этой информации анализируются некоторые пакеты, полученные от сервера. Помимо этого, атакующему часто необходима хорошо спланированная по времени DoS атака на сервер имен. Для защиты используйте зашифрованные соединения с проверкой подлинности хоста, к которому происходит подключение.

1.1.2.6 Черви

Несмотря на очевидную разницу, червей часто путают с вирусами. В отличие от вируса, червя для распространения не требуется заражение находящейся на хосте программы. Они специализируются на максимально быстром распространении в сетевых структурах. Черви прошлого, такие как Ramen, Lion и Adore использовали известные дыры безопасности серверных программ, таких как bind8 и lprNG. Защититься от червей относительно легко. При условии что между обнаружением проблемы в безопасности и моментом, когда червь достигнет Вашего сервера, должно пройти некоторое время, шансы получить обновленную версию программы до этого момента весьма высоки. Поэтому важно регулярно устанавливать обновления безопасности.

1.2 Общие советы и хитрости безопасности

Для соблюдения безопасности важно придерживаться определенных правил. Следующий список правил может быть взят за основу:

- Принимайте и устанавливайте пакеты, содержащие обновления безопасности максимально быстро.
- Следите за последними проблемами безопасности:
 - opensuse-security-announce@opensuse.org это почтовая рассылка SUSE посвященная новостям безопасности. Этот источник предлагает из первых рук информацию об обновлениях пакетов и новых участниках команды безопасности из числа активных контрибьюторов. На эту рассылку Вы можете подписаться на странице <http://en.opensuse.org/Communicate/Mailinglists>.
 - Бюллетень безопасности SUSE доступен в виде новостной ленты по адресу http://www.novell.com/linux/security/suse_security.xml.
 - bugtraq@securityfocus.com одна из самых известных рассылок посвященных безопасности. Рекомендуется следить за этой рассылкой, в которую приходит от 15 до 20 сообщений в день. Подробнее о ней можно узнать здесь <http://www.securityfocus.com>.
- Обсуждайте любые интересующие Вас проблемы безопасности в нашей рассылке opensuse-security@opensuse.org.
- Согласно правилу использования наименьших необходимых для работы прав доступа, избегайте выполнения своих ежедневных задач от пользователя `root`. Это уменьшает риск получения «кукушкиного яйца» или вируса и защищает Вас от Ваших собственных ошибок.
- По возможности всегда используйте зашифрованное соединение для работы на удаленной машине. Использование `ssh` (secure shell) в качестве замены `telnet`, `ftp`, `rsh`, и `rlogin` должно стать повседневной практикой.
- Избегайте методов аутентификации основанных исключительно на IP адресах.
- Пытайтесь содержать в актуальном состоянии самые важные пакеты системы, относящиеся к сети и подпишитесь на соответствующие рассылки безопасности, чтобы получать уведомления о новых версиях этих программ (`bind`, `postfix`, `ssh`, etc.) Это же касается программ, относящихся к локальной безопасности.
-

Измените файл `/etc/permissions` для оптимизации прав доступа к существенным для безопасности Вашей системы файлам. Если Вы удалите у программы бит `setuid` вполне может оказаться, что она не сможет работать так, как задумано. С другой стороны в большинстве случаев такая программа должна считаться потенциальной проблемой для безопасности. Это правило также применимо к доступным для всех на запись директориям и файлам.

- Запретите абсолютно все сетевые сервисы, которые не являются необходимыми для работы Вашего сервера. Это делает Вашу систему менее уязвимой. Открытые порты с сокетами в статусе `LISTEN`, можно обнаружить программой `netstat`. Что касается опций, рекомендуется использовать `netstat -ap` или `netstat -anp`. Опция `-p` позволяет узнать, какой процесс занимает порт под каким именем.

Сравнивайте результаты `netstat` с результатами работы внешнего сканера портов. Для этого прекрасно подойдет `nmap`, которая не только проверит порты Вашего компьютера, а также сделает некоторые заключения о службах, которые их используют. Однако сканирование портов может быть интерпретировано как акт агрессии, поэтому не производите его без ведома администратора системы. И наконец помните, что важно сканировать не только TCP порты, но и UDP порты (опции `-sS` и `-sU`).

- Для надежного контроля целостности файлов Вашей системы используйте программу `AIDE` (Advanced Intrusion Detection Environment), доступную в . Во избежание подделки зашифруйте созданную `AIDE` базу данных. Более того, храните резервную копию этой базы данных подальше от своего компьютера, на внешнем недоступном по сети носителе.
- Будьте внимательны при установке стороннего программного обеспечения, Известен случай, когда хакер встроил троянского коня в архив пакета с программой, к счастью это быстро обнаружили. Устанавливайте бинарные пакеты только в случае если у Вас нет сомнений в сайте с которого Вы их скачали.

RPM пакеты SUSE подписаны `gpg`. Ключ, использованный SUSE для подписи:

```
ID:9C800ACA 2000-10-19 SUSE Package Signing Key <build@suse.de>  
Key fingerprint = 79C1 79B2 E1C8 20C1 890F 9994 A84E DAE8 9C80 0ACA
```

Команда `rpm --checksig package.rpm` показывает верна ли контрольная сумма и подпись пакета, который Вы собираетесь установить. Вы можете

найти ключ на первом CD дистрибутива и на большинстве серверов ключей по всему миру.

- Регулярно проверяйте бекапы пользовательских и системных файлов. Учтите, что если Вы не проверите работоспособность произведенного бекапа, он может оказаться бесполезным в критической ситуации.
- Проверяйте логи. При возможности напишите небольшой скрипт для поиска подозрительных записей. Возможно это не такая уж простая задача. В конце концов, только Вам известно, какие записи подозрительны, а какие - нет.
- Используйте `tcp_wrapper` для ограничения доступа к определенным службам Вашей машины и управления IP адресами, которым разрешено подключение к конкретным службам. Для подробной информации о `tcp_wrapper` обратитесь к справочному руководству по `tcpd` и `hosts_access` (`man 8 tcpd`, `man hosts_access`).
- Используйте `SuSEfirewall` для усиления безопасности обеспечиваемой `tcpd` (`tcp_wrapper`).
- Создавайте свои меры безопасности так, чтобы они были избыточными: сообщение, выводимое дважды, лучше, чем отсутствие сообщения вообще.
- Если Вы используете гибернацию с сохранением на диск, рассмотрите возможность шифрования файла гибернации с помощью скрипта `configure-suspend-encryption.sh`. Эта программа создает ключ, копирует его в `/etc/suspend.key` и изменяет `/etc/suspend.conf` для шифрования файла гибернации.

1.3 Использование Центрального адреса для сообщений о безопасности

Если Вы обнаружите проблему связанную с безопасностью (пожалуйста, сначала проверьте наличие обновлений для соответствующего пакета), напишите имейл security@suse.de. Пожалуйста, включите подробное описание проблемы и номер версии соответствующего пакета. SUSE пришлет Вам ответ настолько

быстро, насколько это возможно. Мы рекомендуем шифровать сообщения при помощи pgp. Pgp ключ SUSE:

```
ID:3D25D3D9 1999-03-06 SUSE Security Team <security@suse.de>  
Key fingerprint = 73 5F 2E 99 DF DB 94 C4 8F 5A A3 AE AF 22 F2 D5
```

Этот ключ также доступен для загрузки по адресу <http://www.suse.com/support/security/contact.html>.

Часть I. Аутентификация

Авторизация с помощью PAM

При авторизации Linux использует PAM (pluggable authentication modules - подключаемые модули авторизации) в качестве прослойки между пользователем и приложением. Модули PAM доступны для всей системы, поэтому они могут быть запрошены любым приложением. Эта глава описывает механизм работы и настройку модулей авторизации.

2.1 Что такое PAM?

Системным администраторам и программистам часто требуется ограничить доступ к некоторым частям системы или функциям конкретного приложения. Если бы не было PAM, приложения необходимо было бы адаптировать к каждому новому механизму авторизации: LDAP, Samba, Kerberos. Такая адаптация также требовалась бы при появлении любого другого механизма авторизации. Эта процедура требует значительных временных ресурсов и подвержена ошибкам. Единственный путь для устранения этих недостатков - выделить авторизационный механизм из приложения и делегировать авторизацию централизованно управляемым модулям. После этого, при необходимости использовать новую схему аутентификации, ее достаточно адаптировать или написать подходящий *модуль PAM* для этой схемы.

Концепция PAM включает в себя:

- *Модули PAM*, которые представляют собой набор разделяемых библиотек для использования определенными механизмами авторизации.

- *Стек модулей* с одним и более модулями PAM.
- *Службу* PAM, которой требуется авторизация с использованием модулей PAM или стека модулей. Как правило, в качестве имени службы используется имя соответствующего приложения - например, `login` или `su`. Имя службы `other` зарезервировано для правил по умолчанию.
- *Аргументы модуля*, с помощью которых можно управлять выполнением конкретного модуля PAM.
- Механизм, определяющий *результат* выполнения конкретного модуля PAM. Положительный результат приводит к выполнению следующего модуля PAM. Способ обработки отрицательного значения зависит от конфигурации: от «не важно, продолжить» до «прекратить немедленно» и промежуточных значений.

2.2 Структура файла конфигурации PAM

Есть два способа конфигурации PAM:

Конфигурация с использованием одного файла (`/etc/pam.conf`)

Настройки каждой службы хранятся в `/etc/pam.conf`. Однако, из соображений сопровождения и юзабилити, данная схема конфигурации не используется в .

Конфигурация с использованием директории (`/etc/pam.d/`)

Каждая служба (или программа), которая использует механизм PAM, хранит свой файл настроек в директории `/etc/pam.d/`. Например, служба `sshd` находится в файле `/etc/pam.d/sshd`.

Файлы в `/etc/pam.d/` определяют модули PAM используемые для авторизации. Каждый файл состоит из строк, определяющих службу и каждая строка содержит до четырех компонентов:

```
TYPE
CONTROL
MODULE_PATH
MODULE_ARGS
```

Эти компоненты имеют следующее значение:

TYPE

Определяет тип службы. Модули PAM обрабатываются как стеки. Различные типы модулей используются для разных целей. Например один модуль производит проверку пароля, второй - размещения, с которого запрошен доступ, и еще один считывает относящиеся к пользователю настройки. PAM известно о четырех различных типах модулей:

`auth`

Проверяет подлинность пользователя, обычно запрашивая пароль. Однако для этого также можно использовать карту-ключ или биометрические данные (например сканирование отпечатков пальцев или радужной оболочки глаза).

`account`

Модули этого типа проверяют, есть ли у пользователя общие привилегии для доступа к запрашиваемой службе. Например, подобная проверка должна выполняться для запрета входа в систему пользователей с истекшим сроком действия аккаунта.

`password`

Целью данного типа модулей является разрешение на изменение авторизационного токена. В большинстве случаев им является пароль.

`session`

Модули этого типа отвечают за управление и конфигурацию пользовательских сессий. Они запускаются до авторизации и после нее для учета попыток входа в систему и конфигурации пользовательского окружения (почтовых аккаунтов, домашних директорий, ограничений системы и т.д.)

CONTROL

Управляет поведением PAM модуля. Для каждого из модулей возможны следующие флаги управления:

`required`

Модуль с этим флагом должен быть успешно обработан для продолжения авторизации. В случае отказа модуля с флагом `required` обраба-

тываются остальные модули с этим флагом и только после этого пользователь получает сообщение о неудачной попытке авторизации.

`requisite`

Модули с этим флагом тоже должны быть обработаны успешно, так же как у модули с флагом `required`. Однако в случае неудачи модуль с этим флагом немедленно сообщает пользователю об отказе, и остальные модули не выполняются. В случае успеха другие модули обрабатываются последовательно, так же как и в случае модулей с флагом `required`. Флаг `requisite` может быть использован для базовой проверки наличия определенных условий необходимых для корректной авторизации.

`sufficient`

После успешного завершения работы модуля с этим флагом, запросившее авторизацию приложение получает немедленное сообщение об успехе и дальнейшая обработка модулей прекращается, при условии, что все предыдущие модули с флагом `required` выполнены успешно. Отказ модуля с флагом `sufficient` не имеет прямых последствий - все последующие модули выполняются в обычном порядке.

`optional`

Успешное выполнение или отказ модуля с этим флагом не имеет прямых последствий. Это может быть полезно для модулей, которые просто отображают уведомление (например, уведомляют пользователя о новом сообщении), не предпринимая никаких дальнейших действий.

`include`

При установке этого флага файл, указанный в качестве аргумента, вставляется в данное место.

MODULE_PATH

Содержит полное имя файла модуля PAM. Его не требуется указывать явно, если модуль находится в директории по умолчанию `/lib/security` (для 64-битных платформ, поддерживаемых , в директории `/lib64/security`).

MODULE_ARGS

Содержит разделенный пробелами список опций для управления модулем PAM, такие как `debug` (включает отладку) или `nullok` (позволяет использовать пустые пароли).

Кроме этого, существуют глобальные файлы настроек для модулей PAM в директории `/etc/security`, которые определяют точное поведение этих модулей (примерами являются файлы `pam_env.conf` и `time.conf`). Каждое приложение, которое использует модуль PAM, в действительности вызывает некоторый набор функций PAM, которые обрабатывают информацию в различных файлах конфигурации и возвращают результат запросившему его приложению.

Для облегчения создания и поддержки модулей PAM, были созданы общие дефолтные файлы конфигурации для типов модулей `auth`, `account`, `password` и `session`. Они извлечены из конфигурации PAM для каждого приложения. Обновления глобальных модулей конфигурации PAM в `common-*` распространяются таким образом на файлы конфигурации PAM, и администратору не требуется обновлять каждый файл конфигурации PAM по отдельности.

Глобальные файлы конфигурации PAM обслуживаются с помощью утилиты `pam-config`. Эта утилита автоматически добавляет новые модули в конфигурацию, изменяя конфигурацию существующих модулей или удаляет модули (или опции) из конфигурации. Ручные изменения при обслуживании конфигурации PAM сведены к минимуму или не требуются вовсе.

ПРИМЕЧАНИЕ: Смешанные 64-битные и 32-битные инсталляции

При использовании 64-битной операционной системы возможно также включать окружение для запуска 32-битных приложений. В этом случае убедитесь, что Вы установили обе версии PAM модулей.

2.3 Конфигурация `sshd` с использованием PAM

В качестве примера рассмотрим конфигурацию PAM для `sshd`:

Пример 2.1 Конфигурация PAM для `sshd` (`/etc/pam.d/sshd`)

```
#%PAM-1.0
auth      requisite      pam_nologin.so ❶
auth      include        common-auth ❷
account   requisite      pam_nologin.so ❷
account   include        common-account ❸
password  include        common-password ❸
session   required       pam_loginuid.so ❹
```

```
session include common-session ❸
```

- ❶ Объявляет версию этого конфигурационного файла - PAM 1.0. Это просто соглашение, но его можно использовать в будущем для проверки версии.
- ❷ Проверяет, существует ли `/etc/nologin`. Если да - логин разрешен только пользователю `root`.
- ❸ Ссылается на конфигурационные файлы четырех типов модулей: `common-auth`, `common-account`, `common-password` и `common-session`. Эти четыре файла содержат дефолтные настройки для каждого типа модуля.
- ❹ Устанавливает атрибут `uid` для процесса, который прошел авторизацию.

Включая эти файлы вместо добавления каждого модуля по отдельности в соответствующую конфигурацию PAM, Вы автоматически получаете обновляемую конфигурацию PAM в случае, если администратор сменит дефолтные настройки. Ранее после изменений в PAM или после установки нового приложения Вам приходилось бы изменять конфигурационные файлы для всех приложений вручную. Сейчас настройка PAM производится централизованно и все изменения автоматически наследуются конфигурацией каждой службы PAM.

Первый подключаемый файл (`common-auth`) вызывает три модуля типа `auth`: `pam_env.so`, `pam_gnome_keyring.so` и `pam_unix2.so`. См. Пример 2.2, «Дефолтная конфигурация секции `auth` (`common-auth`)» (стр. 24).

Пример 2.2 Дефолтная конфигурация секции `auth` (`common-auth`)

<code>auth</code>	<code>required</code>	<code>pam_env.so</code>	❶
<code>auth</code>	<code>optional</code>	<code>pam_gnome_keyring.so</code>	❷
<code>auth</code>	<code>required</code>	<code>pam_unix2.so</code>	❸

- ❶ `pam_env.so` загружает `/etc/security/pam_env.conf` для установки переменных окружения в соответствии со значениями в этом файле. Он может быть использован для установки корректного значения переменной `DISPLAY`, поскольку модулю `pam_env` известно, откуда производится логин.
- ❷ Этот модуль автоматически разблокирует ключи GNOME при необходимости.
- ❸ `pam_unix2`, проверяет логин и пароль пользователя, используя `/etc/passwd` и `/etc/shadow`.

Весь стек модулей `auth` обрабатывается до получения `sshd` любой информации о результате авторизации. Поскольку все модули стека имеют управляющий

флаг `required`, они должны быть успешно обработаны до получения `sshd` сообщения об успехе. Если один из модулей вернет отрицательный результат, весь стек модулей будет обработан до конца и только затем `sshd` будет уведомлен об отрицательном результате.

После успешного завершения всех модулей типа `auth` будет обработано следующее выражение. В нашем случае Пример 2.3, «Дефолтная конфигурация секции `account` (`common-account`)» (стр. 25). `common-account` содержит всего один модуль, `pam_unix2`. Если `pam_unix2` сообщит, что пользователь существует, `sshd` будет уведомлен об этом и будет обработан следующий стек модулей (`password`), показанный в Пример 2.4, «Дефолтная конфигурация секции `password` (`common-password`)» (стр. 25).

Пример 2.3 Дефолтная конфигурация секции `account` (`common-account`)

```
account required          pam_unix2.so
```

Пример 2.4 Дефолтная конфигурация секции `password` (`common-password`)

```
password requisite      pam_pwcheck.so  nullok cracklib
password                optional      pam_gnome_keyring.so  use_authtok
password                required      pam_unix2.so         use_authtok nullok
```

Итак, конфигурация ПАМ для `sshd` включает в себя только выражение `include`, ссылающееся на дефолтную конфигурацию модулей `password`, которая находится в `common-password`. Эти модули должны завершиться успешно (управляющие флаги `requisite` и `required`) при запросе приложением смены токена авторизации.

Изменение пароля или другого токена авторизации требует проверки безопасности. Она производится при помощи модуля `pam_pwcheck`. Модуль `pam_unix2` используемый далее, подхватывает любые новые и старые пароли у `pam_pwcheck`, так что пользователю не требуется повторная авторизация после смены пароля. Эта процедура делает невозможным обход проверок, производимых модулем `pam_pwcheck`. В случае, если нужно настроить типы `account` или `auth` на блокировку аккаунтов после истечения срока действия пароля, модули `password` тоже должны быть использованы.

Пример 2.5 Дефолтная конфигурация секции `session` (`common-session`)

```
session required      pam_limits.so
session required      pam_unix2.so
session optional      pam_apparmor.so
session optional      pam_umask.so
```

```
session optional          pam_gnome_keyring.so      auto_start only_if=gdm,lxdm
```

Последними вызываются модули типа `session` (встроенные в файл `common-session`) для настройки сессии пользователя в соответствии с конкретным пользователем. Модуль `pam_limits` загружает файл `/etc/security/limits.conf`, который может определять ограничения на использование определенных ресурсов системы. Модуль `pam_unix2` обрабатывается снова. Модуль `pam_umask` может быть использован для определения маски битов доступа создаваемых файлов. Поскольку этот модуль задан с флагом `optional`, отказ этого модуля не повлияет на успешное выполнение всего стека модулей сессии. Модули `session` вызываются во второй раз при выходе пользователя из системы.

2.4 Настройка модулей PAM

Некоторые из модулей PAM можно конфигурировать. Файлы настроек находятся в директории `/etc/security`. Эта секция кратко описывает файлы конфигурации имеющие отношение к примеру с `sshd` — `pam_env.conf` и `limits.conf`.

2.4.1 `pam_env.conf`

`pam_env.conf` может быть использован для определения стандартного пользовательского окружения, которое устанавливается при вызове модуля `pam_env`. Он позволяет установить переменные окружения используя следующий синтаксис:

```
VARIABLE [DEFAULT=value] [OVERRIDE=value]
```

VARIABLE

Имя переменной окружения.

```
[DEFAULT=<value>]
```

Значение по умолчанию *value*, которое хочет установить администратор.

```
[OVERRIDE=<value>]
```

Значения, которые могут быть запрошены и установлены `pam_env`, перекрывая дефолтное значение.

Типичный пример использования `pam_env` — изменение переменной `DISPLAY`, которая изменяется в случае удаленного входа в систему. Это описано в Пример 2.6, «`pam_env.conf`» (стр. 27).

Пример 2.6 `pam_env.conf`

```
REMOTEHOST      DEFAULT=localhost OVERRIDE=@{PAM_RHOST}  
DISPLAY         DEFAULT=${REMOTEHOST}:0.0 OVERRIDE=${DISPLAY}
```

Первая строка устанавливает значение переменной `REMOTEHOST` в `localhost`, которое используется, если `pam_env` не может определить другое значение. В свою очередь, переменная `DISPLAY` содержит значение `REMOTEHOST`. Дополнительную информацию можно найти в комментариях к файлу `/etc/security/pam_env.conf`.

2.4.2 `pam_mount.conf`

Задача `pam_mount` — примонтировать домашние каталоги пользователей во время входа в систему и отмонтировать их во время выхода. Этот модуль используется для окружения, в котором центральный файловый сервер хранит все домашние директории пользователей. Используя этот метод нет необходимости монтировать `/home` полностью, с домашними директориями всех пользователей. Вместо этого монтируется только домашний каталог того пользователя, который находится в процессе входа в систему.

После установки `pam_mount`, шаблон файла `pam_mount.conf.xml` доступен в `/etc/security`. Описание различных его элементов можно найти в справочной системе `man 5 pam_mount.conf`.

Базовую настройку этого модуля можно произвести с помощью YaST. Выберите *Сетевые службы > Членство в домене Windows > Настройки эксперта* чтобы добавить файловый сервер; подробнее смотрите Раздел “Configuring Clients” (Глава 15, *Samba*, ↑Содержание).

2.4.3 `limits.conf`

Ограничения для пользователя или группы могут быть установлены в файле `limits.conf`, который используется модулем `pam_limits`. Этот файл позволяет установить жесткие ограничения, нарушение которых не допускается, и

мягкие ограничения, которые можно нарушить временно. Подробную информацию о синтаксисе и опциях можно найти в комментариях к файлу `/etc/security/limits.conf`.

2.5 Настройка PAM при помощи `pam-config`

Утилита `pam-config` поможет Вам настроить глобальные файлы конфигурации PAM (`/etc/pam.d/common-*-pc`), а также несколько других приложений. Для отображения списка поддерживаемых модулей используйте команду `pam-config --list-modules`. Команда `pam-config` используется для сопровождения файлов конфигурации PAM. С ее помощью можно добавлять модули в Вашу конфигурацию PAM, удалять их, или модифицировать их опции. При изменении глобальных файлов конфигурации PAM не требуется ручная настройка PAM для каждого отдельного приложения.

Простой пример использования `pam-config` выглядит так:

1 Автоматическое создание новой конфигурации PAM в стиле Unix.

Давайте при помощи `pam-config` создадим простейшую конфигурацию, которую Вы сможете расширить позже. Команда `pam-config --create` создает простую конфигурацию для авторизации в UNIX. Существующие файлы конфигурации будут перезаписаны, а их резервные копии сохранены как `*.pam-config-backup`.

2 Добавление нового метода авторизации.

Добавление нового метода авторизации (например, LDAP) в Ваш стек модулей PAM сводится к простой команде `pam-config --add --ldap`. При необходимости LDAP добавляется во все `common-*-pc` файлы конфигурации PAM.

3 Включение отладочной информации для тестирования.

Чтобы убедиться в работоспособности новой процедуры авторизации, включим отладку для всех операций, связанных с PAM. С помощью команды `pam-config --add --ldap-debug` отладку можно включить для всех операций PAM связанных с LDAP. Отладочную информацию можно найти в файле `/var/log/messages`.

4 Проверка конфигурации.

Перед окончательным запуском новой конфигурации PAM, проверьте, содержит ли она все настройки, которые Вы хоте-

ли в нее добавить. Команда `pam-config --query --module` отображает тип и параметры для запрашиваемого модуля PAM.

- 5 Отключение отладочной информации.** Теперь, когда мы окончательно удовлетворены конфигурацией, отключим отладку. Команда `pam-config --delete --ldap-debug` отключает отладку для LDAP авторизации. Если Вы включали отладку для других модулей, используйте аналогичные команды, чтобы выключить и их.

Подробную информацию о команде `pam-config` и ее опциях можно получить в справочном руководстве о команде `pam-config(8)`.

2.6 Ручная настройка PAM

Если Вы предпочитаете создавать и поддерживать файлы конфигурации PAM вручную, убедитесь, что `pam-config` запрещена для этих файлов.

Когда Вы создаете файлы конфигурации PAM с нуля используя команду `pam-config --create`, она создает символические ссылки с файлов `common-*` на файлы `common-*-pc`. Удаление этих символических ссылок эффективно для отключения `pam-config`, так как `pam-config` оперирует только файлами `common-*-pc` и эти файлы не изменятся при отсутствии символических ссылок.

2.7 Дальнейшие инструкции

После установки системы в директории `/usr/share/doc/packages/pam` можно найти дополнительную документацию:

README

В этой директории есть общий файл README. Поддиректория `modules` содержит файлы README для доступных модулей PAM.

Linux-PAM. Руководство системного администратора

Этот документ содержит все, что системный администратор должен знать о PAM. В нем обсуждается широкий спектр тем, от синтаксиса конфигурационных файлов до аспектов безопасности PAM.

Руководство для создателей модулей Linux-PAM

Этот документ обобщает данную тему с точки зрения разработчика, описывая, как создавать модули PAM соответствующие стандартам.

Linux-PAM. Руководство разработчика приложений

Этот документ содержит все, что нужно разработчику, который хочет использовать библиотеки PAM в своем приложении.

Справочное руководство PAM

PAM, как и каждый из его модулей, поставляется со справочным руководством, предоставляющим неплохой обзор функциональности соответствующего компонента.

Использование NIS

Поскольку множество UNIX систем используют общие сетевые ресурсы, возникла необходимость общей идентификации пользователей и групп на всех компьютерах внутри сети. Сеть должна быть прозрачной для пользователей: их окружение не должно зависеть от того компьютера, который они используют в данный момент. Этого можно достигнуть используя службы NIS и NFS. NFS распределяет файловые системы по сети и описывается в Глава 14, *Sharing File Systems with NFS* (↑Содержание).

NIS (Network Information Service, Информационная служба сети) можно описать как схожую с базой данных службу, которая предоставляет доступ по сети к содержимому файлов `/etc/passwd`, `/etc/shadow` и `/etc/group`. NIS также может быть использована для других целей (например, для доступа к содержимому файлов `/etc/hosts` или `/etc/services`), однако данное применение здесь не рассматривается. NIS часто называют *YP*, поскольку она работает как «yellow pages - жёлтые страницы» сети.

3.1 Настройка NIS серверов

Для распространения информации NIS по сети, необходима инсталляция одного сервера (*главного*) для обслуживания всех клиентов или подчиненных NIS серверов, запрашивающих эту информацию у главного и доставляющих её своим клиентам.

- Для настройки одного NIS сервера в своей сети обратитесь к Раздел 3.1.1, «Настройка главного сервера NIS» (стр. 32).

- Если Вашему главному NIS серверу необходимо поставлять данные подчиненным серверам, настройте его как описано в Раздел 3.1.1, «Настройка главного сервера NIS» (стр. 32), а затем установите подчиненные сервера в подсетях согласно Раздел 3.1.2, «Настройка подчиненного NIS-сервера» (стр. 37).

3.1.1 Настройка главного сервера NIS

Для настройки главного сервера NIS в Вашей сети:

- 1 Проверьте, установлен ли модуль YaST для настройки NIS сервера. Для этого запустите YaST и выберите *Программное обеспечение > Управление программным обеспечением*. Поищите пакет `yast2-nis-server` и установите его если требуется.
- 2 Запустите *YaST > Сетевые службы > Сервер NIS*.
- 3 Если Вам нужен всего один NIS-сервер в сети или это будет главный сервер для подчиненных NIS-серверов, выберите *Установка и настройка главного NIS-сервера*. YaST установит требуемые пакеты.

ПОДСКАЗКА

Если программы NIS-сервера уже установлены на Вашем компьютере, запустите создание главного сервера NIS нажав *Создать главный NIS-сервер*.

Рисунок 3.1 *Настройка NIS-сервера*



4 Укажите основные настройки NIS:

4a Введите имя домена NIS.

4b Укажите, должен ли этот узел быть также и NIS-клиентом (позволять пользователям входить в систему и получать данные от NIS-сервера) выбрав *Этот узел также является NIS-клиентом*.

4c Если данный NIS-сервер должен выполнять функции главного для подчиненных NIS-серверов в других подсетях, выберите *Есть действующий подчиненный сервер NIS*.

Опция *Быстрое распределение отображений* полезна только в комбинации с *Есть действующий подчиненный сервер NIS*. Она ускоряет передачу отображений подчиненным серверам.

4d Выберите *Разрешить изменение паролей*, если хотите позволить пользователям в Вашей сети (и локальным, и управляемым NIS-сервером) изменять свои пароли на NIS-сервере (с помощью команды

yppasswd). Ее активация делает доступными опции *Разрешить изменение GECOS полей* и *Разрешить изменение оболочки входа в систему*. «GECOS» означает, что пользователи смогут также изменять свои настройки имен и адресов командой `ypchfn`. «Оболочка» позволит пользователям изменять свою оболочку по умолчанию командой `ypchsh` (например, для переключения с `bash` на `sh`). Новая оболочка может быть выбрана только из записей файла `/etc/shells`.

- 4е** Выберите *Открыть порт в брандмауэре* чтобы YaST изменил настройки брандмауэра для работы NIS-сервера.

Рисунок 3.2 *Настройка главного сервера*

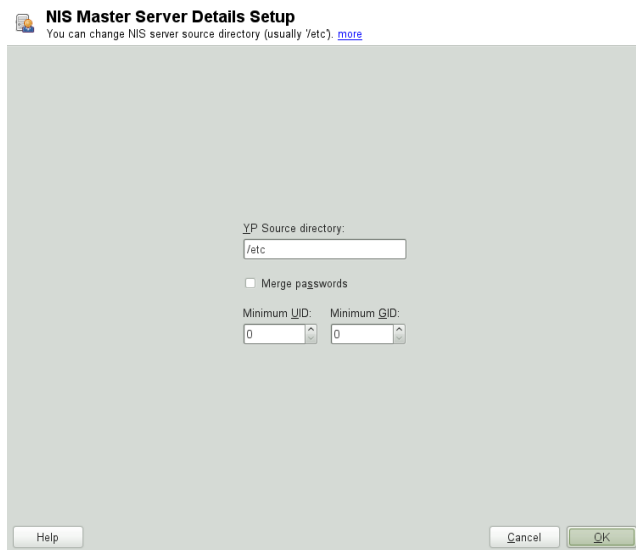


- 4ф** Выйдите из этого диалога при помощи кнопки *Далее* или нажмите *Прочие глобальные настройки* для изменения дополнительных настроек.

Прочие глобальные настройки позволяют изменить исходный каталог NIS-сервера (по умолчанию `/etc`). Также, на этой странице можно объединить пароли. Чекбокс должен быть отмечен, если Вы хотите создать базу данных пользователей из системных файлов аутентификации: `/etc/passwd`, `/etc/shadow` и `/etc/group`. Также задайте минимальные идентификаторы пользователя и группы, которые могут

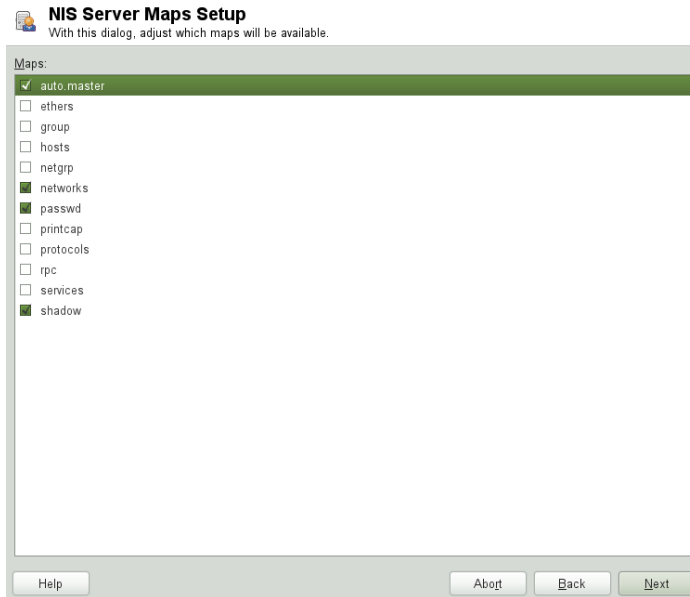
быть предложены NIS. Нажмите *OK* для подтверждения настроек и возврата на предыдущий экран.

Рисунок 3.3 *Изменение директории и синхронизация файлов с NIS-сервером*



- 5 Если Вы выбрали опцию *Есть действующий подчиненный сервер NIS*, введите имена подчиненных хостов и нажмите *Далее*. Если нет - этот шаг будет пропущен.
- 6 Далее появится диалог настройки базы данных. Укажите *Отображения*, части базы данных, которые будут переданы NIS-сервером клиенту. Как правило, настройки по умолчанию вполне адекватны. Нажмите *Далее*.
- 7 Выберите отображения, доступные клиенту и нажмите *Далее* для продолжения.

Рисунок 3.4 Установка отображений NIS-сервера

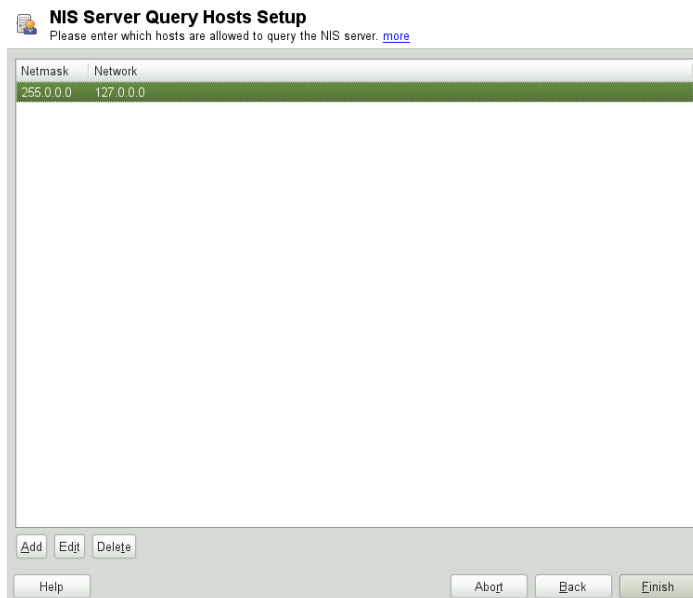


- 8** Укажите, каким хостам разрешено обращение к NIS-серверу. Вы можете добавлять, редактировать, или удалять хосты. Определите из каких сетей можно посылать запросы к NIS-серверу. Обычно это Ваша внутренняя сеть. В этом случае, необходимы следующие две записи:

```
255.0.0.0      127.0.0.0
0.0.0.0        0.0.0.0
```

Первая запись разрешает соединение с текущего хоста, который является NIS-сервером. Вторая разрешает всем хостам посылать запросы к серверу.

Рисунок 3.5 Установка узлов запроса сервера NIS



- 9 Нажмите *Завершить* для сохранения изменений и выхода из диалога конфигурации.

3.1.2 Настройка подчиненного NIS-сервера

Чтобы настроить дополнительные *подчиненные серверы* в Вашей сети:

- 1 Запустите *YaST > Сетевые службы > Сервер NIS*.
- 2 Выберите *Установка и настройка подчиненного NIS-сервера* и нажмите *Далее*.

ПОДСКАЗКА

Если программы NIS-сервера уже установлены на Вашем компьютере, запустите создание главного сервера NIS нажав *Создать подчиненный NIS сервер*.

3 Произведите настройку подчиненного NIS-сервера:

3a Введите домен NIS.

3b Введите имя хоста или IP адрес главного сервера.

3c Выберите *Этот узел также является NIS-клиентом*, если хотите разрешить вход пользователей на этот сервер.

3d Измените настройки брандмауэра *Открыть порт в брандмауэре*.

3e Нажмите *Далее*.

4 Укажите, каким хостам разрешено обращение к NIS-серверу. Вы можете добавлять, редактировать, или удалять хосты. Определите все сети, запросы из которых к NIS-серверу разрешены. Если они разрешены от всех сетей, используйте следующую конфигурацию:

255.0.0.0	127.0.0.0
0.0.0.0	0.0.0.0

Первая запись разрешает соединение с текущего хоста, который является NIS-сервером. Вторая разрешает всем хостам посылать запросы к серверу.

5 Нажмите *Завершить*, чтобы сохранить изменения и выйти из диалога конфигурации.

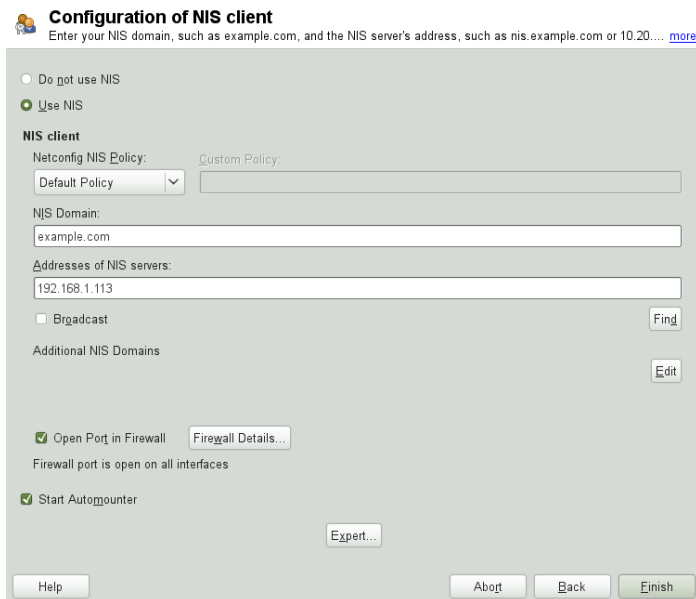
3.2 Настройка NIS-клиентов

Для использования NIS на рабочей станции:

1 Запустите *YaST > Сетевые службы > Клиент NIS*.

- 2 Выберите *Использовать NIS*.
- 3 Введите домен NIS. обычно это имя домена, предоставленной Вам администратором или статический IP адрес, полученный по DHCP. Для информации о DHCP обратитесь к Глава 12, *DHCP* (↑Содержание).

Рисунок 3.6 Установка домена и адреса NIS-сервера



- 4 Введите адреса Ваших NIS-серверов, разделяя их пробелами. Если адрес NIS-сервера Вам неизвестен, нажмите *Найти*, чтобы найти адреса NIS-серверов в Вашем домене с помощью YaST. Время поиска зависит от величины Вашей локальной сети. *Широковещательный* запрашивает NIS-сервер в локальной сети после того как указанные серверы не ответили.
- 5 В зависимости от Вашей локальной инсталляции, возможно Вам потребуется automounter. При выборе этой опции дополнительное программное обеспечение будет установлено при необходимости.
- 6 Если Вы не хотите предоставлять другим хостам информацию о том, какой сервер используется Вашим клиентом, откройте настройки *Эксперт* и запретите *Отвечать удаленным узлам*. Выбрав *Сломанный сервер*, клиенту будет

разрешено получать ответы от сервера через непривилегированный порт. Для дальнейшей информации обратитесь к `man ypbind`.

- 7** Нажмите *Завершить*, чтобы сохранить их и вернуться в Центр Управления YaST. Конфигурация Вашего NIS-клиента завершена.

LDAP — Сервис директорий

Облегченный протокол доступа к каталогам (LDAP) — это набор протоколов, созданный для доступа и поддержания информационных каталогов. LDAP может использоваться для различных целей, например, управление пользователями и группами, управление системной конфигурацией или управление адресами. В этом разделе дается краткое описание как работает OpenLDAP и как управлять данными LDAP с помощью YaST.

В сетевом окружении важно сохранить ценную информацию структурированной и быстро доступной. Это может быть сделано с помощью такого сервиса каталогов, как общие желтые страницы, хранящие информацию доступной в отлично структурированной форме, приспособленной для быстрого поиска.

В идеальном случае, центральный сервер хранит данные в каталоге и распределяет их всем клиентам по определенному протоколу. Данные структурированы в таком виде, который позволяет широкому кругу приложений получить доступ к ним. Таким образом, отпадает необходимость в хранении клиентом каждого календаря или базы электронной почты - вместо этого будет доступен центральный репозиторий. Использование открытого и стандартизированного протокола, как LDAP, гарантирует возможность получения доступа к информации для различных приложений.

Каталог в этом контексте — это тип базы данных, оптимизированного на чтения и поиск:

- Делает возможным множественный доступ для чтения, доступ на запись ограничен небольшим числом модификаций, сделанных администратором. Обыч-

ные базы данных оптимизированы для доступа к большому объему возможных данных за короткий промежуток времени.

- Поскольку доступ на запись может быть выполнен в ограниченном виде, сервис каталогов используется по большей части для администрирования неизменяемой, статической информации. При работе с часто меняющимися данными, особенно с такими наборами данных, как, например, банковские счета, согласованность данных имеет первостепенное значение. Если сумма должна быть вычтена из одного счета для добавления к другому, то такие операции должны происходить одновременно, в течение одной *транзакции*, чтобы гарантировать корректность полученных данных. Традиционные реляционные базы данных обычно имеют очень сильный акцент на целостности данных, такой как поддержка ссылочной целостности транзакций. С другой стороны, краткосрочные несоответствия, как правило, приемлемы для LDAP. Так, часто нет необходимости в таких жестких требованиях к согласованности данных.

Дизайн сервиса каталогов, как LDAP, не подразумевает поддержки сложных модификаций и механизмов запросов. Все приложения, обращающиеся к этому сервису, должны получать доступ к нужным данным легко и быстро.

4.1 LDAP против NIS

Администраторы Unix-систем традиционно используют сервис NIS (Network Information Service, Информационная служба сети) для разрешения имен и предоставления данных по сети. Данные конфигурации содержатся в файлах `group`, `hosts`, `mail`, `netgroup`, `networks`, `passwd`, `printcap`, `protocols`, `rpc` и `services` в каталоге `/etc`, которые предоставляются всем пользователям в сети. Эти файлы могут поддерживаться без особых усилий, так как они состоят из простого текста. Обработка большого количества данных, однако, становится все более и более сложной из-за отсутствия структурирования. NIS разрабатывался для Unix платформ и, потому, не очень подходит для централизованного управления в гетерогенных сетях.

В отличие от NIS, сервис LDAP не ограничен только сетями UNIX. Серверы Windows (начиная с 2000) поддерживают LDAP, как сервис каталогов. Прикладные задачи, описанные выше, дополнительно поддерживаются в не-Unix сетях.

Принцип LDAP может быть применен к структуре любых данных, которая может централизованно администрироваться. Примеры нескольких приложений:

- Применение в качестве замены сервиса NIS
- Перенаправление почты (postfix, sendmail)
- Адресные книги для таких почтовых клиентов, как Mozilla, Evolution и Outlook
- Администрирование зоны, написанной для сервера имен Bind9
- Аутентификация пользователей совместно с Samba в гетерогенных сетях

Этот список может быть расширен, поскольку LDAP более гибкая вещь, чем NIS. Ясная иерархическая структура облегчает администрирование большого количества данных, так как они могут быть найдены значительно проще.

4.2 Структура дерева каталога LDAP

Для лучшего понимания, как работает LDAP и как хранятся данные, очень важно понимать, каким образом данные организованы на сервере и как эта структура позволяет обеспечить быстрый доступ к необходимым данным. Для успешной установки LDAP необходимо ознакомиться с используемой терминологией. Эта секция позволяет понять основной вывод дерева и описывает терминологию в контексте LDAP. Эту секцию можно пропустить, если вы уже имеете некоторое представление о работе LDAP и просто хотите научиться установке LDAP-окружения в . Прочитайте Раздел 4.3, «Конфигурирование сервера LDAP с помощью YaST» (стр. 46) или Раздел 4.7, «Конфигурация сервера LDAP вручную» (стр. 66).

Каталог LDAP имеет структуру дерева. Все записи (называемые объектами) каталога имеют определенную позицию в этой иерархии. Эта иерархия называется *Информационным деревом справочника* (DIT, Directory Information Tree). Полный путь к необходимой записи, который однозначно идентифицирует ее, называется *характерное имя* (distinguished name) или DN. Единый узел вдоль пути к этой записи называется (relative distinguished name) или RDN.

Отношения элементов в пределах дерева LDAP хорошо видны на рисунке Рисунок 4.1, «Структура каталога LDAP» (стр. 44).

Рисунок 4.1 Структура каталога LDAP

Полная диаграмма является вымышленным информационным деревом каталога. Изображены записи на третьем уровне. Каждая запись соответствует одному прямоугольнику на картинке. Полное *distinguished name* для вымышленного пользователя `Geeko Linux` будет, в данном случае, `cn=Geeko Linux, ou=doc, dc=example, dc=com`. Оно создается путем добавления RDN `cn=Geeko Linux` к DN предыдущей записи `ou=doc, dc=example, dc=com`.

Типы объектов, которые хранятся в DIT, в общем случае определяются, следуя *Схеме* (Schema). Тип объекта определяется *классом объекта* (object class). Класс объекта определяет, какие свойства связанного объекта должны или могут быть назначены. Схема, в свою очередь, должна содержать определения всех классов объекта и свойства, использующиеся в нужном прикладном сценарии. Существует несколько общих схем (RFC 2252 и 2256). LDAP RFC определяют несколько наиболее часто используемых схем (см., например, RFC4519). Кроме того, существует много других схем (например, замена Samba, NIS и т.д.). При этом можно самому создавать схемы или использовать несколько дополняющих друг друга схем (если это требуется окружением, в котором будет эксплуатироваться LDAP-сервер).

В Таблица 4.1, «Общий блок используемых классов объектов и атрибутов» (стр. 44) показан небольшой обзор классов объекта из `core.schema` и `inetorgperson.schema`, использующихся в примере, включая свойства и действительные значения свойств.

Таблица 4.1 Общий блок используемых классов объектов и атрибутов

Класс объекта	Значение	Пример записи	Необходимые свойства
dcObject	<i>domainComponent</i> (именованные компоненты домена)	example	dc

Класс объек- та	Значение	При- мер за- писи	Необ- ходи- мые свой- ства
organizationalUnit	<i>organizationalUnit</i> (организа- ционная единица)	doc	ou
inetOrgPerson	<i>inetOrgPerson</i> (персональные данные)	Geeko Linux	sn and cn

В Пример 4.1, «Выдержка из schema.core» (стр. 45) показана часть из дирек-
тивы схемы с объяснениями.

Пример 4.1 *Выдержка из schema.core*

```
attributetype (2.5.4.11 NAME ( 'ou' 'organizationalUnitName') ❶
    DESC 'RFC2256: organizational unit this object belongs to' ❷
    SUP name ) ❸

...
objectclass ( 2.5.6.5 NAME 'organizationalUnit' ❹
    DESC 'RFC2256: an organizational unit' ❺
    SUP top STRUCTURAL ❻
    MUST ou ❼
    MAY (userPassword $ searchGuide $ seeAlso $ businessCategory ❽
        $ x121Address $ registeredAddress $ destinationIndicator
        $ preferredDeliveryMethod $ telexNumber
        $ teletexTerminalIdentifier $ telephoneNumber
        $ internationalISDNNumber $ facsimileTelephoneNumber
        $ street $ postOfficeBox $ postalCode $ postalAddress
        $ physicalDeliveryOfficeName
        $ st $ l $ description) )
...

```

Тип свойства `organizationalUnitName` и переданный класс объекта `organizationalUnit` в данном случае служат примером.

- ❶ Имя свойства, его уникальный OID (*идентификатор объекта*, object identifier (цифровой)) и аббревиатура свойства.
- ❷ Краткое описание свойства с помощью DESC. Соответствующий RFC, на котором основано данное определение, также упоминается в этой строке.
- ❸ SUP указывает на соподчинённый тип свойства, к которому принадлежит это свойство.

- ④ Описание класса объекта `organizationalUnit`, как и в описании свойства, с `OID` и имени класса объекта.
- ⑤ Краткое описание класса объекта.
- ⑥ Запись `SUP top` указывающая, что этот класс объекта не зависит от другого класса объекта.
- ⑦ Начиная с `MUST` перечисляются все типы свойств, которые должны использоваться в связке с объектом типа `organizationalUnit`.
- ⑧ Начиная с `MAY` перечисляются все типы свойств, которые позволено связывать с этим классом объекта.

Очень хорошим введением с описанием использования схем можно найти в документации к `openLDAP`. После установки, она доступна в `/usr/share/doc/packages/openldap2/guide/admin/guide.html`.

4.3 Конфигурирование сервера LDAP с помощью YaST

Используйте `YaST` для первоначальной настройки сервера `LDAP`. Типичные случаи использования серверов `LDAP` включают в себя: управление аккаунтами пользователей и задание настроек почты, серверов `DNS` и `DHCP`.

Рисунок 4.2 YaST: Настройка сервера LDAP

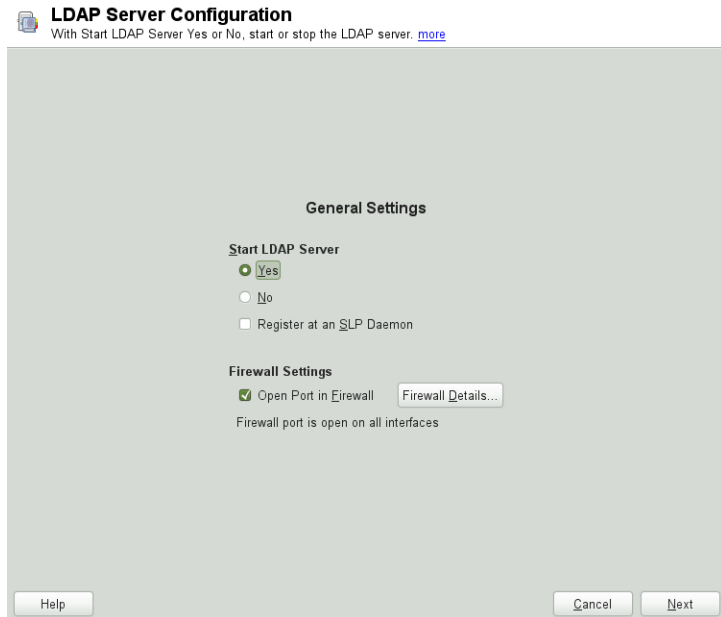



Рисунок 4.3 YaST: Сервер LDAP — Новая база данных

 **New Database**
Choose the Database from hdb and bdb. [more](#)

Basic Database Settings

Database Type:

Base DN:

Administrator DN:
 ☒ Append Base DN

LDAP Administrator Password:

Validate Password:

Database Directory:

☒ Use this database as the default for OpenLDAP clients

Для установки сервера LDAP для управления аккаунтами пользователей, потребуется установить пакеты `yast2-ldap-server` и `openldap2`. Сделайте следующее:

- 1 Запустите YaST от имени `root` и выберите *Сетевые службы > Сервер LDAP*, чтобы вызвать диалог первоначальной настройки.
- 2 Задайте параметры вашего LDAP-сервера в разделе *Общие настройки* (их можно будет изменить позднее) — см. Рисунок 4.2, «YaST: Настройка сервера LDAP» (стр. 47):
 - 2a Укажите необходимость автоматического запуска LDAP-сервера во время загрузки системы.
 - 2b Если необходимо, чтобы сервер LDAP предоставлял свои возможности через SLP, выберите *Регистрироваться в демоне SLP*.
 - 2c Задайте *Настройки брандмауэра*.
 - 2d Нажмите *Далее*.

- 3 Выберите тип сервера: обособленный сервер, основной сервер репликации или вторичный сервер.
- 4 Выберите опции безопасности (*Настройки TLS*).

Строго рекомендуется *Включить TLS*. Дополнительную информацию см. в Шаг 4 (стр. 51).

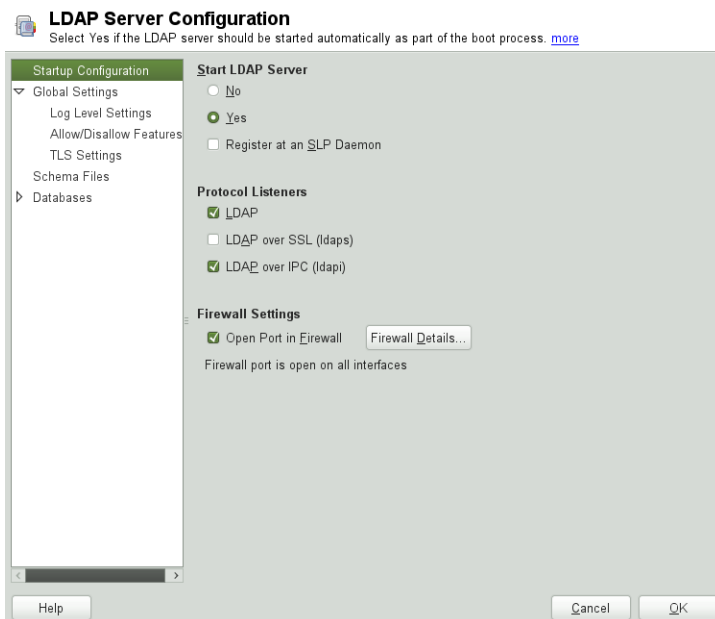
ПРЕДУПРЕЖДЕНИЕ: Шифрование паролей

Активация TLS обеспечивает шифрование паролей передаваемых по сети. Если эта опция не активна, то пароли будут передаваться в не зашифрованном виде.

Так же рассмотрите возможность применения сертификатов и SSL.

- 5 Подтвердите *Основные настройки базы данных* введя *Пароль администратора LDAP* и нажмите *Далее* — см. Рисунок 4.2, «YaST: Настройка сервера LDAP» (стр. 47).
- 6 Проверьте *Итог настройки сервера LDAP* и нажмите *Готово*, чтобы закрыть мастер настройки.

Рисунок 4.4 YaST: Настройка сервера LDAP



Для изменения параметров конфигурации запустите модуль Сервер LDAP и в левой панели выберите *Общие настройки*, чтобы перейти к этому подпункту — см. Рисунок 4.4, «YaST: Настройка сервера LDAP» (стр. 50):

- 1 Подпункт *Настройки уровня журнала* позволяет настроить уровень регистрируемых в журнале событий (уровень подробности) сервера LDAP. Добавьте или удалите из предопределенного списка параметры ведения журнала в соответствии с вашими потребностями. Чем больше опций будет включено, тем быстрее будут разрастаться лог-файлы.
- 2 Укажите какие типы соединений серверу разрешено принимать в *Разрешить/запретить возможности*. Среди них:

Запросы связи LDAPv2

Эта опция позволяет запросы на соединение (bind requests) от клиентов, использующих предыдущую версию протокола (LDAPv2).

Анонимная связь при непустых реквизитах доступа

Обычно сервер LDAP запрещает любые попытки аутентификации с пустыми регистрационными данными (DN или пароль). Включение этой оп-

ции, однако, делает возможным соединение с паролем и без DN для установки анонимного соединения.

Не прошедшая аутентификацию связь при непустом DN

Включение этой опции делает возможным соединение без аутентификации (анонимно), используя DN, но без пароля.

Неаутентифицированные операции обновления для обработки

Включение этой опции позволяет делать неаутентифицированные (анонимные) модификации. Доступ ограничивается согласно ACL и другими правилами.

- 3** *Разрешить/запретить возможности* также позволяет установить флаги сервера. Среди них:

Запретить приём анонимных запросов связи

Сервер больше не будет принимать анонимные соединения. Заметьте, что это не запрещает анонимный доступ к каталогам.

ше не будет принимать анонимные соединения. Заметьте, что это не запрещает анонимный доступ к

Отключить аутентификацию Simple Bind

Полное отключение аутентификации Simple Bind.

Отключить перевод сессии в анонимное состояние перед операциями StartTLS

Сервер больше не будет принудительно переводить аутентифицированное соединение обратно в анонимное состояние при получении операции StartTLS.

Запретить StartTLS, если аутентифицировано

Сервер будет отклонять операцию StartTLS для уже аутентифицированных соединений.

- 4** Для настройки безопасного соединения между клиентом и сервером перейдите к подпункту *Настройки TLS*:

4a Отметьте *Включить TLS* для активации шифрования по протоколам TLS и SSL клиент-серверных соединений.

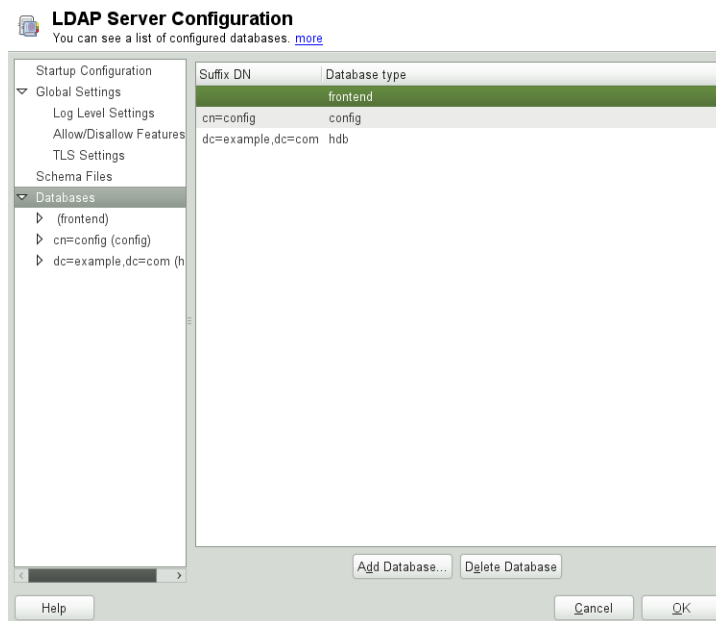
4b Нажмите *Импорт сертификата*, чтобы указать точный путь его расположения или активируйте *Использовать общий сертификат сервера*.

Если пункт *Использовать общий сертификат сервера* не доступен, так как сертификат не был создан во время установки, то нажмите *Запустить модуль управления СА*. Дополнительную информацию см. в Раздел 15.2, «YaST Modules for CA Management» (стр. 202).

Включить файл схемы в конфигурации сервера можно выбрав подпункт *Файлы схем* в левой части диалога. По умолчанию указанные файлы схемы относятся к серверу, являющемуся для YaST источником данных об учетных записях пользователей.

YaST поддерживает традиционные файлы схем (как правило, с именем, заканчивающимся на `.schema`) или LDIF-файлы содержащие описание схемы в LDIF формате OpenLDAP.

Рисунок 4.5 YaST: Сервер LDAP — Настройка базы данных



Чтобы настроить базы данных вашего LDAP-сервера, проделайте следующее:

- 1 Выберите пункт *Базы данных* в левой части диалога.
- 2 Нажмите *Добавить базу данных*, чтобы добавить базу данных.

3 Введите необходимые данные:

Основной DN

Введите основной DN вашего сервера LDAP.

DN администратора

Укажите DN администратора отвечающего за этот сервер. Если выбрать *Добавить базовый DN*, то достаточно указать только `cn` администратора — система подхватит остальные значения автоматически.

Пароль администратора LDAP

Введите пароль администратора базы данных.

Использовать эту базу данных по умолчанию для клиентов OpenLDAP

Выберите эту опцию, если нужно.

4 В следующем диалоге задайте параметры репликации.

5 В следующем диалоге включите возможность политик паролей для обеспечения дополнительной безопасности для сервера LDAP:

5a Выберите *Включить политику паролей*, чтобы задать политики для паролей.

5b Активируйте *Хэшировать текстовые пароли*, чтобы при изменении или создании пароля, его представление в открытом виде хэшировалось перед записью в базу данных.

5c Пункт *Выявить состояние "Учётная запись заблокирована"* обеспечивает вывод соответствующего сообщения об ошибке для запросов к заблокированным аккаунтам.

ПРЕДУПРЕЖДЕНИЕ: Блокирование учетных записей в средах с повышенными требованиями к безопасности

Не используйте опцию *Выявить состояние "Учётная запись заблокирована"*, если ваше окружение имеет повышенные требования к безопасности, потому что сообщение «Учётная запись заблокирована» предоставляет важную с точки зрения безопасности информацию, которая может эксплуатироваться потенциальными злоумышленниками.

5d Введите DN объекта политики по умолчанию. Для использования DN, отличного от предложенного YaST, укажите здесь нужное значение. Иначе принимаются значения, установленные по умолчанию.

6 Завершите конфигурирование базы данных, нажав *Готово*.

Если политики паролей не были выбирали, то сервер, в данный момент, готов к запуску. В противном случае, продолжите конфигурирование политик паролей. Если был выбран объект политики паролей, которого не существует, то YaST создаст его:

1 Введите пароль сервера LDAP. В навигационной панели ниже *Базы данных* откройте объект вашей базы и активируйте пункт *Настройка политики пароля*.

2 Убедитесь, что пункт *Включить политику паролей* активирован. Затем нажмите *Редактировать политику*.

3 Сформируйте политику изменения паролей:

3a Определите количество паролей, хранящихся в истории. Сохраненные пароли не могут использоваться пользователем несколько раз.

3b Определите, будут ли пользователи иметь возможность менять свои пароли и будет ли им нужно задать новый пароль, после его сброса администратором. Дополнительно, при изменении пароля можно затребовать ввод старого пароля.

3c Определите, какую сложность должен иметь пароль. Задается минимальная длина, меньше которой пароль не принимается. Если выбрано *Принимать не проверяемые пароли*, то пользователи смогут использовать зашифрованные пароли, но в этом случае их проверка не может быть выполнена. Если выбрано *Принимать только проверенные пароли*, то пароли, не удовлетворяющие всем критериям сложности, будут отклоняться.

4 Определите политику устаревания паролей:

4a Определите минимальный срок действия паролей (время, которое должно пройти до появления возможности изменить текущий пароль) и максимальный возраст пароля.

- 4b** Определите временной промежуток используемый для вывода предупреждения о истечении времени действия пароля перед его фактическим устареванием.
 - 4c** Установите количество выводимых предупреждений об окончании срока действия пароля перед тем, как он станет недействительным.
- 5** Определите политики блокировок:
- 5a** Разрешите блокировку паролей.
 - 5b** Укажите количество неудачных попыток, по исчерпанию которых пароль блокируется.
 - 5c** Укажите, на какое время будет блокироваться пароль.
 - 5d** Укажите, как долго неудачные попытки ввода паролей будут храниться в кэше.
- 6** Сохраните сделанные настройки политик паролей, нажав *ОК*.

Чтобы отредактировать ранее созданную базу данных, выберите ее DN в дереве слева. В правой части окна, YaST покажет диалог, подобный тому, который использовался при создании новой базы данных (с главным отличием в том, что запись основного DN неактивна и не может быть изменена).

После того, как все необходимые настройки в конфигурацию сервера LDAP будут внесены, нажмите *Готово*. Ваш сервер LDAP готов к работе. Для тонкой настройки используйте механизм динамического конфигурирования OpenLDAP.

Механизм динамического конфигурирования OpenLDAP хранит настройки в самой базе данных LDAP. Она состоит из набора `.ldif` файлов в каталоге `/etc/openldap/slapd.d`. Доступа к этим файлам напрямую не требуется. Для доступа к настройкам можно воспользоваться модулем YaST Сервер LDAP (пакет `yast2-ldap-server`) или клиентом LDAP с помощью команд: `ldapmodify` или `ldapsearch`. Дополнительную информацию об этом механизме см. в Руководстве администратора OpenLDAP.

4.4 Конфигурирование клиента LDAP с помощью YaST

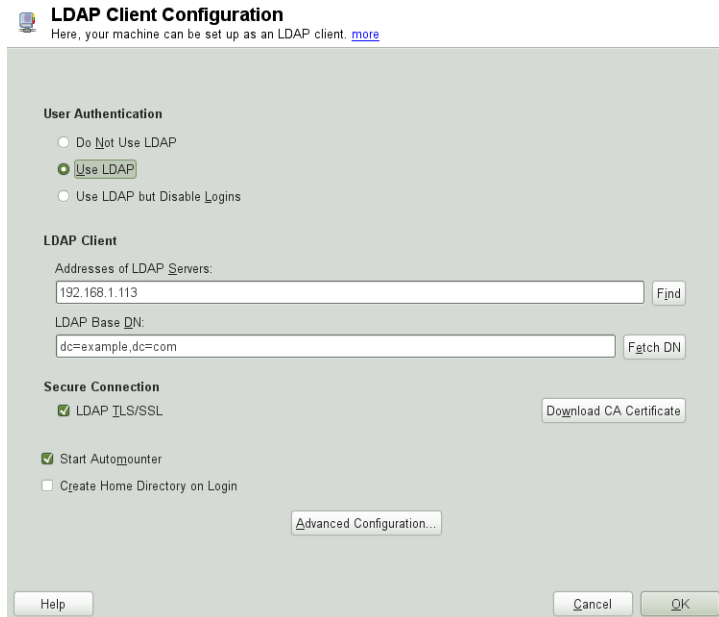
YaST включает в себя модуль для установки управления пользователями через LDAP. Если эта возможность не была выбрана во время установки, то запустите модуль, выбрав *Сетевые службы > Клиент LDAP*. YaST автоматически произведет изменения для PAM и NSS, необходимые для LDAP, и установит необходимые файлы. Просто подключите клиента к серверу и пусть YaST управляет пользователями через LDAP. Эти действия описаны в Раздел 4.4.1, «Стандартная процедура настройки» (стр. 56).

Используйте Клиент LDAP для последующего задания настроек в модулях YaST по настройке групп и пользователей. Сюда относится задание настроек по умолчанию для новых пользователей и групп, а также количество и характер атрибутов, назначенных пользователю или группе. Управление пользователями через LDAP позволяет применять гораздо больше атрибутов к пользователям и группам, чем традиционные решения. Это описано в Раздел 4.4.2, «Конфигурирование модулей YaST управления пользователями и группами» (стр. 60).

4.4.1 Стандартная процедура настройки

Стандартный диалог настройки клиента LDAP (см. Рисунок 4.6, «YaST: Конфигурация клиента LDAP» (стр. 57)) открывается во время установки, если выбрано управление пользователями через LDAP или если выбрать *Сетевые службы > Клиент LDAP* в Центре управления YaST во время установки.

Рисунок 4.6 YaST: Конфигурация клиента LDAP



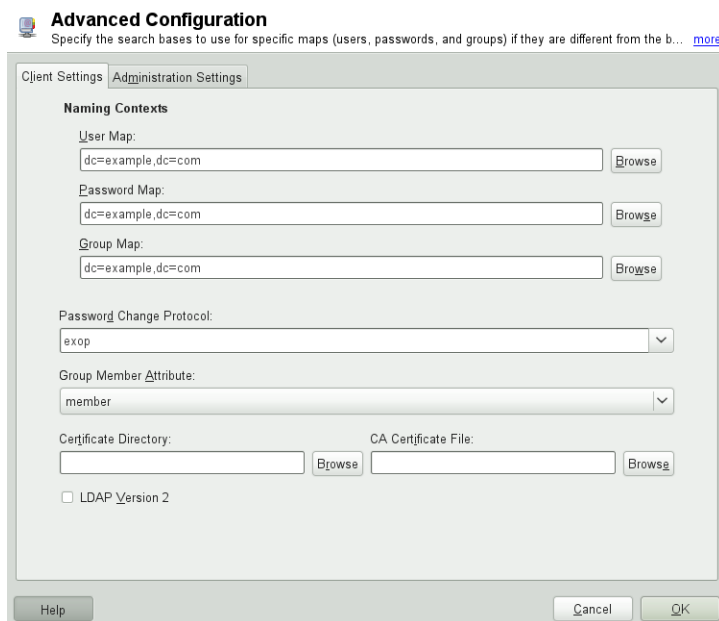
Для аутентификации и управления пользователей через сервер OpenLDAP выполните следующие действия:

- 1 Нажмите *Использовать LDAP*, для активации авторизации через LDAP. Выберите *Использовать LDAP, но отключить вход в систему*, если нужно использовать LDAP для проверки подлинности, но, при этом, другие пользователи не могли авторизоваться через этот клиент.
- 2 Введите IP-адрес используемого сервера LDAP.
- 3 Введите *Базовый DN LDAP* для выбора базы поиска на сервере LDAP. Для получения базового DN автоматически нажмите *Запрос DN*. После этого YaST проверит каждую базу данных LDAP на сервера, указанном выше. Выберите необходимый базовый DN из результатов поиска, предоставленных YaST.
- 4 Если сервер требует, чтобы соединение защищалось с помощью TLS или SSL, то выберите *LDAP TLS/SSL*. Нажмите *Загрузить сертификат CA*, чтобы загрузить сертификат в PEM-формате по указанному URL.

- 5 Выберите *Запустить automounter* для монтирования удаленных директорий на клиенте, такие как внешний каталог /home.
- 6 Выберите *Создать домашний каталог при входе в систему* для автоматического создания домашнего каталога пользователя при первом входе в систему.
- 7 Нажмите *ОК*, чтобы применить настройки.

Для изменения данных на сервер, в качестве администратора, нажмите *Дополнительная настройка*. Следующее диалоговое окно разделено на две вкладки. См. Рисунок 4.7, «YaST: Дополнительная настройка» (стр. 58).

Рисунок 4.7 YaST: Дополнительная настройка



- 1 На вкладке *Настройки клиента* измените следующие параметры, если это необходимо:
 - 1a Если база поиска для пользователей, паролей и групп отличается от глобальной базы поиска, указанной в *Базовый DN LDAP*, введите нужные значения в *Отображение пользователей*, *Отображение паролей* и *Отображение групп*.

- 1b** Укажите протокол используемый для смены пароля. Стандартный метод, который используется по умолчанию — `crypt` — используются хэши паролей, созданные `crypt`. За более подробной информацией об этой и других опциях обратитесь к `man`-странице `passwd`.
- 1c** Укажите группу LDAP, которая будет использоваться в *Атрибут члена группы*. Значение по умолчанию — `member`.
- 1d** Если для проверки сертификата требуется защищенное соединение, то укажите его расположение в PEM-формате в *Файл сертификата CA*. Или укажите каталог с сертификатом.
- 1e** Если сервер LDAP по прежнему использует LDAPv2, то укажите использовать эту версию протокола выбрав *LDAP Версия 2*.
- 2** В *Настройки администратора* измените следующие параметры:
- 2a** Установите базу для хранения данных управления пользователями через *Основной DN настроек*.
- 2b** Введите соответствующее значение для *DN администратора*. Это DN должно быть идентично значению `rootdn`, указанному в `/etc/openldap/slapd.conf`, чтобы дать возможность простым пользователям манипулировать данными, хранящимися на сервере LDAP. Введите полное имя DN (например, `cn=Administrator,dc=example,dc=com`) или активируйте *Добавлять основной DN*, чтобы основной DN автоматически добавлялся при вводе `cn=Administrator`.
- 2c** Отметьте *Создать конфигурационные объекты по умолчанию* для создания основных объектов конфигурации на сервере, чтобы разрешить управление пользователями через LDAP.
- 2d** Если клиентская машина должна служить файловым сервером для домашних каталогов пользователей сети, то отметьте *Домашние каталоги на этой машине*.
- 2e** Используйте раздел *Политика пароля* для выбора, добавления, изменения или удаления параметров настройки использующихся политик па-

ролей. Конфигурация политики паролей с помощью YaST является одной из частей установки сервера LDAP.

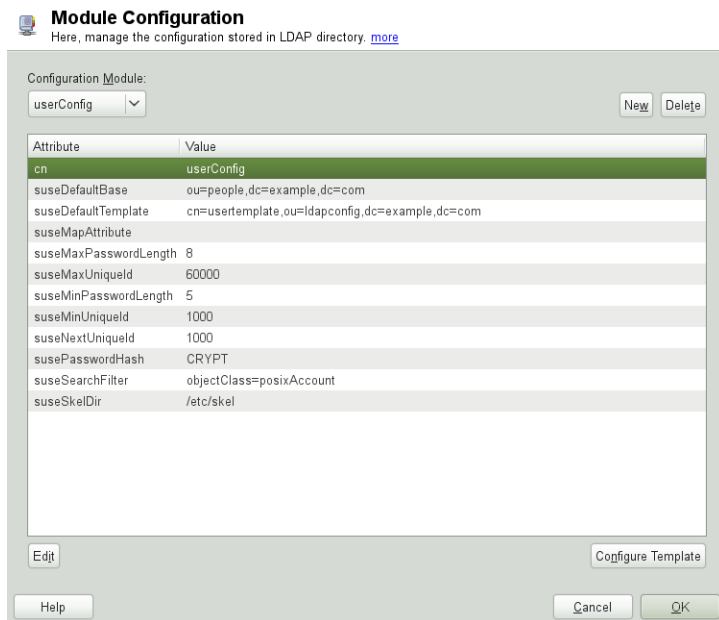
- 2f** Нажмите *ОК*, чтобы выйти из *Дополнительная настройка*, затем *Готово* для принятия настроек.

Используйте *Настройка параметров управления пользователями* для редактирования записей на сервере LDAP. Доступ к модулям конфигурации предоставляется согласно ACL и ACI, находящимся на сервере. Следуйте указаниям описанным в Раздел 4.4.2, «Конфигурирование модулей YaST управления пользователями и группами» (стр. 60).

4.4.2 Конфигурирование модулей YaST управления пользователями и группами

Используйте YaST Клиент LDAP для адаптации модуля управления пользователями и группами и расширения их по необходимости. Определите шаблон со значениями по умолчанию для конкретных атрибутов, чтобы сделать процедуру регистрацию данных более простой. Преднастройки, сделанные здесь, хранятся, как объекты LDAP в каталоге LDAP. Регистрация данных пользователей совершается с помощью обычных модулей YaST для управления пользователями и группами. Зарегистрированные данные хранятся в виде объектов LDAP на сервере.

Рисунок 4.8 YaST: Конфигурация модулей



Диалог конфигурации модулей (Рисунок 4.8, «YaST: Конфигурация модулей» (стр. 61)) позволяет создать новые модули, выбирать и модифицировать уже существующие модули, а также создавать и изменять шаблоны для таких модулей.

Для создания нового модуля конфигурации проделайте следующее:

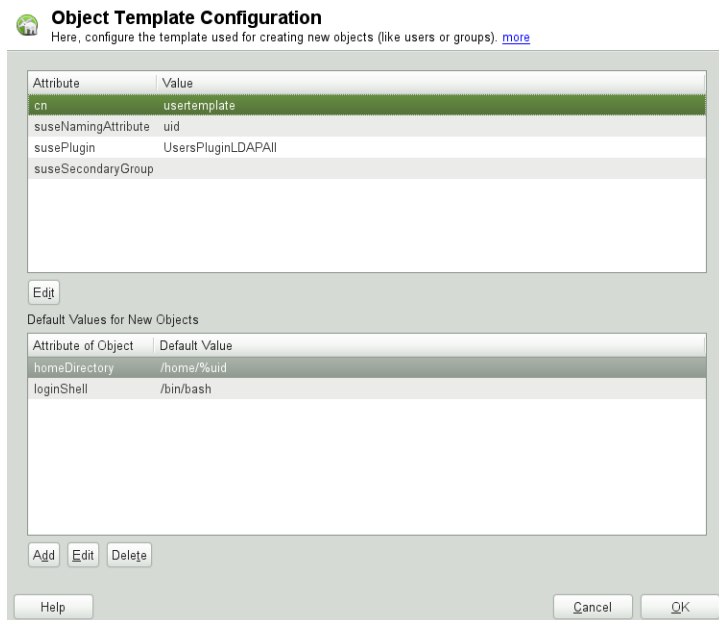
- 1 В *Настройка клиента LDAP* нажмите *Дополнительная настройка*, затем откройте вкладку *Настройки администратора*. Нажмите *Настройка параметров управления пользователями* и укажите полномочия сервера LDAP.
- 2 Нажмите *Создать* и выберите тип создаваемого модуля. Для модуля конфигурации пользователя выберите `suseUserConfiguration`, а для конфигурации группы — `suseGroupConfiguration`.
- 3 Укажите имя нового шаблона (например, `userConfig`). Затем содержимое будет выведено в виде таблицы с перечислением всех атрибутов, разрешенных в этом модуле, с установленными им значениями.

- 4 Оставьте предустановленные значения или укажите собственные выбрав соответствующий атрибут и нажав *Изменить* введите новое значение. Переименуйте модуль, просто изменив атрибут `cn`. Для удаления выбранного модуля модуля нажмите *Удалить*.
- 5 После нажатия на *ОК* новый модуль будет добавлен в меню выбора.

В модуль YaST для администрирования пользователей и групп уже встроены шаблоны с разумными стандартными значениями. Чтобы отредактировать шаблон, связанный с модулем конфигурации, сделайте следующее (Рисунок 4.9, «YaST: Конфигурирование шаблона объекта» (стр. 62)):

- 1 В диалоге *Конфигурация модуля* нажмите *Настройка шаблона*.
- 2 Задайте нужные значения для общих атрибутов, относящихся к этому шаблону, или оставьте их пустыми. Пустые атрибуты удаляются на сервере LDAP.
- 3 Измените, удалите или добавьте новые значения по умолчанию для новых объектов (объекты конфигурирования пользователей и групп в дереве LDAP).

Рисунок 4.9 YaST: Конфигурирование шаблона объекта



Объедините шаблон с его модулем, установив в атрибута модуля `susedefaulttemplate` значение DN соответствующего шаблона.

ПОДСКАЗКА

Значения по умолчанию для атрибутов можно создать используя другие атрибуты используя переменную вместо абсолютного значения. Например, при создании нового пользователя указав `cn=%sn %givenName` значения атрибутов `sn` и `givenName` поставятся автоматически.

Как только все модули и шаблоны настроены правильно и готовы к работе, новые группы и пользователи могут быть зарегистрированы обычным способом через YaST.

4.5 Конфигурация пользователей и групп LDAP в YaST

Фактическая регистрация данных пользователей и групп лишь слегка отличается от процедуры без использования LDAP. Следующие инструкции относятся к администрированию пользователей. Процесс администрирования групп аналогичен.

- 1 Перейдите к администрированию пользователей в YaST: *Безопасность и пользователи > Управление пользователями и группами*.
- 2 Используйте меню *Задать фильтр*, чтобы ограничиться только пользователями LDAP, введя пароль Root DN.
- 3 Нажмите *Добавить* и укажите параметры для нового пользователя. Откроется диалог с четырьмя вкладками:

3а Укажите имя пользователя, его логин и пароль на вкладке *Информация о пользователе*.

3б На вкладке *Подробности* уточните членство в группах, оболочку входа и домашний каталог нового пользователя. При необходимости, измените значения по умолчанию на нужные значения. Значения по умолчанию так же, как и настройки пароля, могут быть определены

с помощью процедуры, описанной в Раздел 4.4.2, «Конфигурирование модулей YaST управления пользователями и группами» (стр. 60).

3c Измените или оставьте настройки по умолчанию на вкладке *Настройки пароля*.

3d Выберите вкладку *Дополнения*, затем выберите плагин LDAP и нажмите *Запуск* для настройки дополнительных атрибутов LDAP нового пользователя (см. Рисунок 4.10, «YaST: Дополнительные настройки LDAP» (стр. 64)).

4 Нажмите *OK* для сохранения настроек и выхода из модуля конфигурирования пользователя.

Рисунок 4.10 *YaST: Дополнительные настройки LDAP*

Additional LDAP Settings
Here, see the table of all allowed attributes for the current LDAP entry that were not set in previous dialogs. [more](#)

Attribute	Value
cn	Tux Geeko
givenName	Tux
sn	Geeko
audio	
businessCategory	
carLicense	
departmentNumber	
displayName	
employeeNumber	
employeeType	
homePhone	
homePostalAddress	
initials	
jpegPhoto	
labeledURI	
mail	
manager	
mobile	
o	
pager	
photo	

Edit

Help Cancel OK

При открытии формы администрирования пользователей выводятся *Опции LDAP*. И предоставляется возможность применять фильтры поиска в LDAP для выбора доступных пользователей или перейти к модулю для конфигурирования пользователей и групп LDAP, выбрав *Конфигурация пользователей и групп LDAP*.

4.6 Просмотр дерева каталогов LDAP

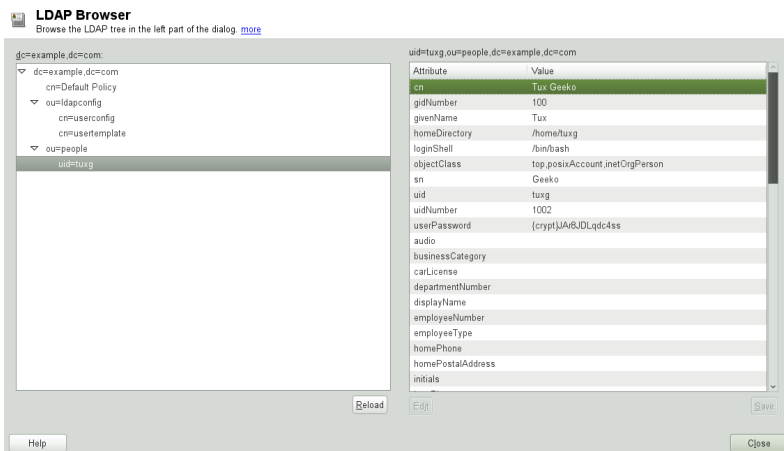
Для просмотра дерева каталогов LDAP и всех его записей удобно использовать Обозреватель LDAP в YaST:

- 1 Зайдите под пользователем `root`.
- 2 Запустите *YaST > Сетевые службы > Обозреватель LDAP*.
- 3 Введите адрес сервера LDAP, DN администратора и пароль для корневого DN этого сервера (если нужен доступ к данным, хранящимся на сервере не только на чтение, но и на запись).

Или не вводите пароль и выберите *Анонимный доступ*.

Во вкладке *Дерево LDAP* отображается содержимое каталога LDAP, к которому подключена ваша машина. Нажмите на элемент, чтобы раскрыть его содержимое.

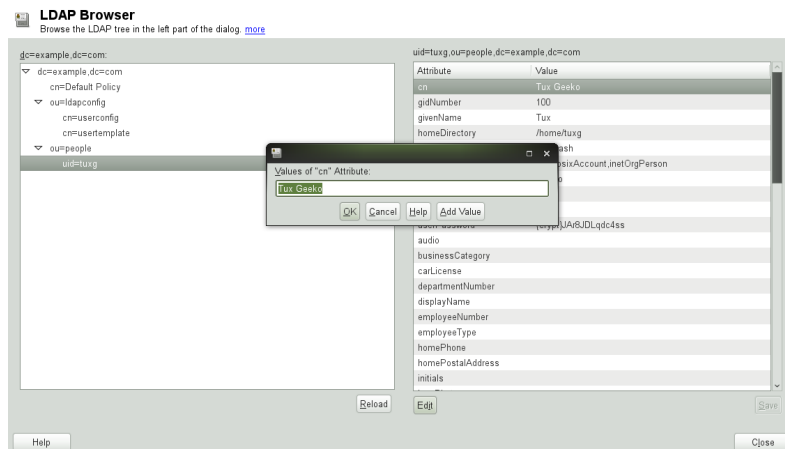
Рисунок 4.11 Просмотр дерева каталогов LDAP



- 4 Для более подробного просмотра записей выберите ее во вкладке *Дерево LDAP* и перейдите на вкладку *Данные элемента*.

Будут отображены все атрибуты и значения, относящиеся к этой записи.

Рисунок 4.12 Просмотр данных



- 5 Чтобы изменить значение любого из этих атрибутов, выберите его и нажмите *Редактировать*. Введите новое значение и нажмите *Сохранить*. Будет запрошен пароль от корневого DN.
- 6 Выйдите из браузера LDAP нажав *Заккрыть*.

4.7 Конфигурация сервера LDAP вручную

YaST использует базу данных OpenLDAP динамической конфигурации (back-config) для хранения настроек серверов LDAP. Для получения дополнительной информации о механизме динамической конфигурации см. map-страницу slapd-config(5) или Руководство администратора OpenLDAP 2.4 доступного в файле /usr/share/doc/packages/openldap2/guide/admin/guide.html после установки пакета openldap2.

ПОДСКАЗКА: Обновление старой установки OpenLDAP

YaST больше не использует файл `/etc/openldap/slapd.conf` для хранения конфигурации OpenLDAP. В случае обновления системы, копия оригинального файла `/etc/openldap/slapd.conf` будет сохранена под именем `/etc/openldap/slapd.conf.YaSTsave`.

Чтобы получить доступ к серверной конфигурации в удобной форме, можно воспользоваться внешней SASL-аутентификацией. Например, с помощью команды `ldapsearch` запущенной от имени пользователя `root` можно полностью просмотреть конфигурацию `slapd`:

```
ldapsearch -Y external -H ldapi:/// -b cn=config
```

4.7.1 Запуск и остановка серверов

Как только сервер LDAP полностью настроен и все необходимые записи сделаны согласно образцу, описанному в Раздел 4.8, «Управление данными в каталоге LDAP» (стр. 67), запустите сервер с правами пользователя `root`, введя команду `rcldap start`. Для остановки сервера вручную, введите `rcldap stop`. Запросить статус запущенного сервера можно с помощью команды `rcldap status`.

Редактор уровня запуска YaST можно использовать для запуска и остановки сервера при загрузке и остановке системы соответственно. Также есть возможность создать символические ссылки для скриптов старта и остановки с помощью команды `insserv`.

4.8 Управление данными в каталоге LDAP

OpenLDAP предоставляет серию инструментов для администрирования данными в каталоге LDAP. Четыре наиболее важных инструмента для добавления, удаления, поиска и изменения набора данных кратко описаны ниже.

4.8.1 Вставка данных в каталог LDAP

Если конфигурация вашего сервера LDAP выполнена корректно и готова к использованию (имеются соответствующие записи для `suffix`, `directory`, `rootdn`, `rootpw` и `index`), то можно приступить к вводу записей. Для этой задачи OpenLDAP предоставляет команду `ldapadd`. Если возможность — добавляйте объекты в базу данных в связке (по вполне практичным причинам). Для этого в LDAP присутствует поддержка обработки данных в формате LDIF (LDAP data interchange format — Формат обмена данными LDAP). Файл LDIF — это простой тестовый файл, содержащий произвольное количество пар атрибутов и значений. Файл LDIF для создания структуры приведен, например, на Рисунок 4.1, «Структура каталога LDAP» (стр. 44) мог бы выглядеть, как в Пример 4.2, «Пример файла LDIF» (стр. 68).

ВАЖНО: Кодировка файлов LDIF

LDAP работает с UTF-8 (Юникод). Умляюты должны кодироваться правильно. В противном случае, избегайте использования умляутов и других специальных символов или используйте `iconv` для перевода кодировки в UTF-8.

Пример 4.2 *Пример файла LDIF*

```
# The Organization
dn: dc=example,dc=com
objectClass: dcObject
objectClass: organization
o: Example dc: example

# The organizational unit development (devel)
dn: ou=devel,dc=example,dc=com
objectClass: organizationalUnit
ou: devel

# The organizational unit documentation (doc)
dn: ou=doc,dc=example,dc=com
objectClass: organizationalUnit
ou: doc

# The organizational unit internal IT (it)
dn: ou=it,dc=example,dc=com
objectClass: organizationalUnit
ou: it
```

Сохраните файл с суффиксом `.ldif`, затем отправьте его на сервер с помощью следующей команды:

```
ldapadd -x -D dn_of_the_administrator -W -f file.ldif
```

Ключ `-x`, в данном случае, отключает аутентификацию с помощью SASL. Ключ `-D` указывает пользователя от имени которого иницируется данное действие. Действительный DN администратора, введенный здесь, должен быть таким же, как в файле `slapd.conf`. В данном примере, это `cn=Administrator,dc=example,dc=com`. Ключ `-W` предотвращает ввод пароля в командной строке (открытым текстом) и активирует отдельное приглашение для ввода пароля. С помощью ключа `-f` указывается имя файла. Детали работы `ldapadd` можно посмотреть в Пример 4.3, «`ldapadd` и `example.ldif`» (стр. 69).

Пример 4.3 *ldapadd и example.ldif*

```
ldapadd -x -D cn=Administrator,dc=example,dc=com -W -f example.ldif
```

```
Enter LDAP password:
adding new entry "dc=example,dc=com"
adding new entry "ou=devel,dc=example,dc=com"
adding new entry "ou=doc,dc=example,dc=com"
adding new entry "ou=it,dc=example,dc=com"
```

Данные конкретного пользователя могут быть подготовлены в отдельном файле LDIF. В Пример 4.4, «Данные LDIF для пользователя Tux» (стр. 69) пользователь Tux добавляется в новый каталог LDAP.

Пример 4.4 *Данные LDIF для пользователя Tux*

```
# coworker Tux
dn: cn=Tux Linux,ou=devel,dc=example,dc=com
objectClass: inetOrgPerson
cn: Tux Linux
givenName: Tux
sn: Linux
mail: tux@example.com
uid: tux
telephoneNumber: +49 1234 567-8
```

Файл LDIF может содержать произвольное количество объектов. Возможно сразу отправить полные ветви каталога на сервер или только его часть, как показано в примере для конкретных объектов. Если требуется менять отдельные данные достаточно часто, то рекомендуется использовать для этого отдельную часть одиночного объекта.

4.8.2 Модификация данных в каталоге LDAP

Для модификации набора данных предоставляется команда `ldapmodify`. Наиболее простой способ сделать это — изменить соответствующий файл LDIF, а затем отправить его на сервер. Так, для того, чтобы изменить номер телефона коллеги Tux с +8 1234 5 67-08 на +8 1234 5 67-10 отредактируйте LDIF-файл, как показано в Пример 4.5, «Изменение файла tux.ldif» (стр. 70).

Пример 4.5 *Изменение файла tux.ldif*

```
# coworker Tux
dn: cn=Tux Linux,ou=devel,dc=example,dc=com
changetype: modify
replace: telephoneNumber
telephoneNumber: +8 1234 5 67-10
```

Импортируйте измененный файл в каталог LDAP с помощью следующей команды:

```
ldapmodify -x -D cn=Administrator,dc=example,dc=com -W -f tux.ldif
```

Или отправьте измененные атрибуты напрямую с помощью команды `ldapmodify`:

1 Запустите `ldapmodify` и введите пароль:

```
ldapmodify -x -D cn=Administrator,dc=example,dc=com -W
Enter LDAP password:
```

2 Введите измененные данные (внимательно следите за правильностью синтаксиса) в порядке приведенном ниже:

```
dn: cn=Tux Linux,ou=devel,dc=example,dc=com
changetype: modify
replace: telephoneNumber
telephoneNumber: +8 1234 5 67-10
```

Более подробную информацию о команде `ldapmodify` и его синтаксисе смотрите на соответствующей man-странице.

4.8.3 Поиск и чтение данных из каталога LDAP

OpenLDAP представляет инструмент командной строки для поиска и чтения данных внутри каталога LDAP — `ldapssearch`. Простой запрос можно выполнить следующим образом:

```
ldapsearch -x -b dc=example,dc=com "(objectClass=*)"
```

Ключ `-b` задает базу для поиска (раздел дерева, внутри которого будет производиться поиск). В данном случае, это `dc=example,dc=com`. Чтобы выполнить более подробный поиск в конкретном подразделе каталога LDAP (например, только в пределах раздела `devel`), укажите его `ldapsearch` с помощью ключа `-b`. Ключ `-x` запрашивает активацию простой авторизации. Строка `(objectClass=*)` объявляет, что необходимо прочесть все объекты, содержащиеся в каталоге. Эта опция может использоваться для проверки, что все записи были указаны правильно и сервер отвечает корректно, например, после создания нового дерева каталогов. Дополнительную информацию об использовании этой команды можно найти на [man-странице `ldapsearch\(1\)`](#).

4.8.4 Удаление данных из каталога LDAP

Удалить нежелательные записи можно с помощью команды `ldapdelete`. Её синтаксис схож с синтаксисом описанных выше команд. Например, чтобы удалить всю информацию о связанной с записью `Tux Linux`, запустите следующую команду:

```
ldapdelete -x -D cn=Administrator,dc=example,dc=com -W cn=Tux \
Linux,ou=devel,dc=example,dc=com
```

4.9 Дополнительная информация

Более сложные темы, вроде конфигурирования SASL или установки реплицирующего сервера LDAP, который распределяет нагрузку на множество подчиненных, намеренно не вошли в этот раздел. Детальная информация об этих темах может быть найдена в Руководство администратора OpenLDAP 2.4 (стр. 72).

Веб-сайт проекта openLDAP предоставляет исчерпывающую информацию для начинающих и продвинутых пользователей.

OpenLDAP Faq-O-Matic

Очень богатая коллекция с вопросами и ответами на них по поводу установки, настройки и использования openLDAP. Посетите <http://www.openldap.org/faq/data/cache/1.html>.

Быстрый старт

Краткие пошаговые инструкции по установке вашего первого сервера LDAP. Ищите их на <http://www.openldap.org/doc/admin24/quickstart.html> или в установленной системе в разделе 2 файла `/usr/share/doc/packages/openldap2/guide/admin/guide.html`.

Руководство администратора OpenLDAP 2.4

Детальное введение во все важные аспекты конфигурации LDAP, в том числе, контроль доступа и шифрование. Смотрите на <http://www.openldap.org/doc/admin24/> или в установленной системе `/usr/share/doc/packages/openldap2/guide/admin/guide.html`.

Введение в LDAP

Детальное общее введение в основные принципы LDAP : <http://www.redbooks.ibm.com/redbooks/pdfs/sg244986.pdf>.

Печатная литература о LDAP:

- *LDAP System Administration* by Gerald Carter (ISBN 1-56592-491-6)
- *Understanding and Deploying LDAP Directory Services* by Howes, Smith, and Good (ISBN 0-672-32316-8)

Последний справочный материал по теме LDAP — соответствующие документы RFC (запрос комментариев), с 2251 по 2256.

Active Directory Support

Active Directory* (AD) is a directory-service based on LDAP, Kerberos, and other services that is used by Microsoft Windows to manage resources, services, and people. In an MS Windows network, AD provides information about these objects, restricts access to them, and enforces policies. lets you join existing AD domains and integrate your Linux machine into a Windows environment.

5.1 Integrating Linux and AD Environments

With a Linux client (configured as an Active Directory client) that is joined to an existing Active Directory domain, benefit from various features not available on a pure Linux client:

Browsing Shared Files and Folders with SMB

Both Nautilus (the GNOME file manager) and Konqueror (its KDE counterpart) support browsing shared resources through SMB.

Sharing Files and Folders with SMB

Both Nautilus (the GNOME file manager) and Konqueror (its KDE counterpart) support sharing folders and files as in Windows.

Accessing and Manipulating User Data on the Windows Server

Through Nautilus and Konqueror, users are able to access their Windows user data and can edit, create, and delete files and folders on the Windows server. Users can access their data without having to enter their password multiple times.

Offline Authentication

Users are able to log in and access their local data on the Linux machine even if they are offline or the AD server is unavailable for other reasons.

Windows Password Change

This port of AD support in Linux enforces corporate password policies stored in Active Directory. The display managers and console support password change messages and accept your input. You can even use the Linux `passwd` command to set Windows passwords.

Single-Sign-On through Kerberized Applications

Many applications of both desktops are Kerberos-enabled (*kerberized*), which means they can transparently handle authentication for the user without the need for password reentry at Web servers, proxies, groupware applications, or other locations.

A brief technical background for most of these features is given in the following section.

5.2 Background Information for Linux AD Support

Many system components need to interact flawlessly in order to integrate a Linux client into an existing Windows Active Directory domain. Рисунок 5.1, «Active Directory Authentication Schema» (стр. 74) highlights the most prominent ones. The following sections focus on the underlying processes of the key events in AD server and client interaction.

Рисунок 5.1 *Active Directory Authentication Schema*

To communicate with the directory service, the client needs to share at least two protocols with the server:

LDAP

LDAP is a protocol optimized for managing directory information. A Windows domain controller with AD can use the LDAP protocol to exchange directory information with the clients. To learn more about LDAP in general and about the open source port of it, OpenLDAP, refer to Глава 4, *LDAP — Сервис директорий* (стр. 41).

Kerberos

Kerberos is a third-party trusted authentication service. All its clients trust Kerberos's authorization of another client's identity, enabling kerberized single-sign-on (SSO) solutions. Windows supports a Kerberos implementation, making Kerberos SSO possible even with Linux clients.

The following client components process account and authentication data:

Winbind

The most central part of this solution is the winbind daemon that is a part of the Samba project and handles all communication with the AD server.

NSS (*Name Service Switch*)

NSS routines provide name service information. Naming service for both users and groups is provided by `nss_winbind`. This module directly interacts with the winbind daemon.

PAM (*Pluggable Authentication Modules*)

User authentication for AD users is done by the `pam_winbind` module. The creation of user homes for the AD users on the Linux client is handled by `pam_mkhomedir`. The `pam_winbind` module directly interacts with winbindd. To learn more about PAM in general, refer to Глава 2, *Авторизация с помощью PAM* (стр. 19).

Applications that are PAM-aware, like the login routines and the GNOME and KDE display managers, interact with the PAM and NSS layer to authenticate against the Windows server. Applications supporting Kerberos authentication (such as file managers, Web browsers, or e-mail clients) use the Kerberos credential cache to access user's Kerberos tickets, making them part of the SSO framework.

5.2.1 Domain Join

During domain join, the server and the client establish a secure relation. On the client, the following tasks need to be performed to join the existing LDAP and Kerberos

SSO environment provided by the Window domain controller. The entire join process is handled by the YaST Domain Membership module, which can be run during installation or in the installed system. The steps involved are:

- 1** The Windows domain controller, providing both LDAP and KDC (Key Distribution Center) services, is located.
- 2** A machine account for the joining client is created in the directory service.
- 3** An initial ticket granting ticket (TGT) is obtained for the client and stored in its local Kerberos credential cache. The client needs this TGT to get further tickets allowing it to contact other services, like contacting the directory server for LDAP queries.
- 4** NSS and PAM configurations are adjusted to enable the client to authenticate against the domain controller.

During client boot, the winbind daemon is started and retrieves the initial Kerberos ticket for the machine account. winbindd automatically refreshes the machine's ticket to keep it valid. To keep track of the current account policies, winbindd periodically queries the domain controller.

5.2.2 Domain Login and User Homes

The login managers of GNOME and KDE (GDM and KDM) have been extended to allow the handling of AD domain login. Users can choose to log into the primary domain the machine has joined or to one of the trusted domains with which the domain controller of the primary domain has established a trust relationship.

User authentication is mediated by a number of PAM modules as described in Раздел 5.2, «Background Information for Linux AD Support» (стр. 74). The `pam_winbind` module used to authenticate clients against Active Directory or NT4 domains is fully aware of Windows error conditions that might prohibit a user's login. The Windows error codes are translated into appropriate user-readable error messages that PAM gives at login through any of the supported methods (GDM, KDM, console, and SSH):

`Password has expired`

A message stating that the password has expired and needs to be changed is displayed. The system prompts for a new password and informs the user if the

new password does not comply with corporate password policies (for example the password is too short, too simple, or already in the history). If a user's password change fails, the reason is shown and a new password prompt is given.

Account disabled

The user sees an error message stating that the account has been disabled and to contact the system administrator.

Account locked out

The user sees an error message stating that the account has been locked and to contact the system administrator.

Password has to be changed

The user can log in but receives a warning that the password needs to be changed soon. This warning is sent three days before that password expires. After expiration, the user cannot log in.

Invalid workstation

When a user is restricted to specific workstations and the current machine is not among them, a message appears that this user cannot log in from this workstation.

Invalid logon hours

When a user is only allowed to log in during working hours and tries to log in outside working hours, a message informs the user that logging in is not possible at that time.

Account expired

An administrator can set an expiration time for a specific user account. If that user tries to log in after expiration, the user gets a message that the account has expired and cannot be used to log in.

During a successful authentication, `pam_winbind` acquires a ticket granting ticket (TGT) from the Kerberos server of Active Directory and stores it in the user's credential cache. It also renews the TGT in the background, requiring no user interaction.

supports local home directories for AD users. If configured through YaST as described in Раздел 5.3, «Configuring a Linux Client for Active Directory» (стр. 78), user homes are created at the first login of a Windows (AD) user into the Linux client. These home directories look and feel entirely the same as standard Linux user home directories and work independently of the AD

domain controller. Using a local user home, it is possible to access a user's data on this machine (even when the AD server is disconnected) as long as the Linux client has been configured to perform offline authentication.

It is also possible to mount server home directories automatically; for more information, see Раздел “Configuring Clients” (Глава 15, *Samba*, ↑Содержание).

5.2.3 Offline Service and Policy Support

Users in a corporate environment must have the ability to become roaming users (for example, to switch networks or even work disconnected for some time). To enable users to log in to a disconnected machine, extensive caching was integrated into the winbind daemon. The winbind daemon enforces password policies even in the offline state. It tracks the number of failed login attempts and reacts according to the policies configured in Active Directory. Offline support is disabled by default and must be explicitly enabled in the YaST Domain Membership module.

When the domain controller has become unavailable, the user can still access network resources (other than the AD server itself) with valid Kerberos tickets that have been acquired before losing the connection (as in Windows). Password changes cannot be processed unless the domain controller is online. While disconnected from the AD server, a user cannot access any data stored on this server. When a workstation has become disconnected from the network entirely and connects to the corporate network again later, acquires a new Kerberos ticket as soon as the user has locked and unlocked the desktop (for example, using a desktop screen saver).

5.3 Configuring a Linux Client for Active Directory

Before your client can join an AD domain, some adjustments must be made to your network setup to ensure the flawless interaction of client and server.

DNS

Configure your client machine to use a DNS server that can forward DNS requests to the AD DNS server. Alternatively, configure your machine to use the AD DNS server as the name service data source.

NTP

To succeed with Kerberos authentication, the client must have its time set accurately. It is highly recommended to use a central NTP time server for this purpose (this can also be the NTP server running on your Active Directory domain controller). If the clock skew between your Linux host and the domain controller exceeds a certain limit, Kerberos authentication fails and the client is logged in using the weaker NTLM (NT LAN Manager) authentication. For more details about using active directory for time synchronization, see Процедура 5.1, «Joining an AD Domain» (стр. 80).

DHCP

If your client uses dynamic network configuration with DHCP, configure DHCP to provide the same IP and hostname to the client. If possible, use static IP addresses.

Firewall

To browse your network neighborhood, either disable the firewall entirely or mark the interface used for browsing as part of the internal zone.

To change the firewall settings on your client, log in as `root` and start the YaST firewall module. Select *Interfaces*. Select your network interface from the list of interfaces and click *Change*. Select *Internal Zone* and apply your settings with *OK*. Leave the firewall settings with *Next > Accept*. To disable the firewall, just set *Service Start* to *Manually* and leave the firewall module with *Next > Accept*.

AD Account

You cannot log in to an AD domain unless the AD administrator has provided you with a valid user account for that domain. Use the AD username and password to log in to the AD domain from your Linux client.

Join an existing AD domain during installation (or by later activating SMB user authentication with YaST in the installed system).

ПРИМЕЧАНИЕ

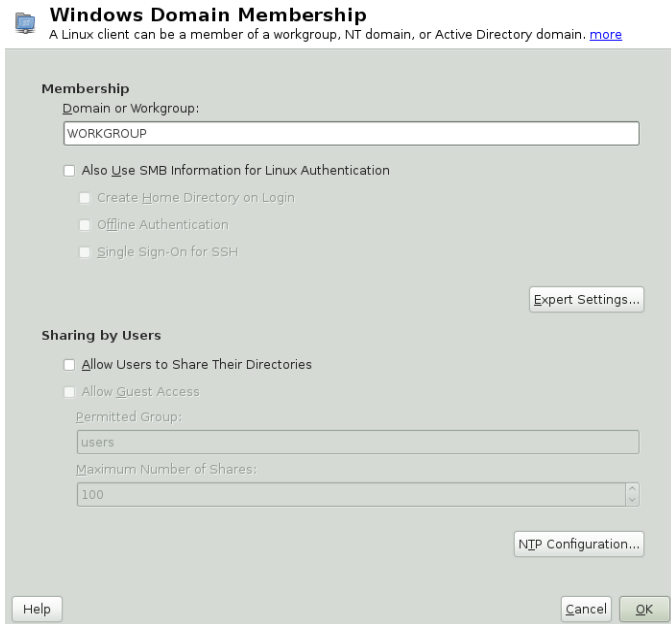
Currently only a domain administrator account, such as `Administrator`, can join into Active Directory.

To join an AD domain in a running system, proceed as follows:

Процедура 5.1 Joining an AD Domain

- 1 Log in as `root` and start YaST.
- 2 Start *Network Services > Windows Domain Membership*.
- 3 Enter the domain to join at *Domain or Workgroup* in the *Windows Domain Membership* screen (see Рисунок 5.2, «Determining Windows Domain Membership» (стр. 80)). If the DNS settings on your host are properly integrated with the Windows DNS server, enter the AD domain name in its DNS format (`mydomain.mycompany.com`). If you enter the short name of your domain (also known as the pre-Windows 2000 domain name), YaST must rely on NetBIOS name resolution instead of DNS to find the correct domain controller. To select from a list of available domains instead, use *Browse* to list the NetBIOS domains, then select the desired domain.

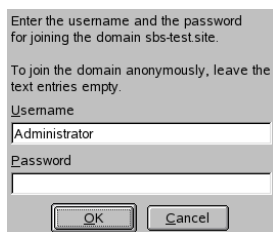
Рисунок 5.2 Determining Windows Domain Membership



- 4 Check *Also Use SMB Information for Linux Authentication* to use the SMB source for Linux authentication.

- 5 Check *Create Home Directory on Login* to automatically create a local home directory for your AD user on the Linux machine.
- 6 Check *Offline Authentication* to allow your domain users to log in even if the AD server is temporarily unavailable, or if you do not have a network connection.
- 7 Select *Expert Settings*, if you want to change the UID and GID ranges for the Samba users and groups. Let DHCP retrieve the WINS server only if you need it. This applies when some of your machines are resolved by the WINS system only.
- 8 Configure NTP time synchronization for your AD environment by selecting *NTP Configuration* and entering an appropriate server name or IP address. This step is obsolete if you have already entered the appropriate settings in the standalone YaST NTP configuration module.
- 9 Click *Finish* and confirm the domain join when prompted for it.
- 10 Provide the password for the Windows administrator on the AD server and click *OK* (see Рисунок 5.3, «Providing Administrator Credentials» (стр. 81)).

Рисунок 5.3 *Providing Administrator Credentials*



Enter the username and the password for joining the domain sbs-test.site.

To join the domain anonymously, leave the text entries empty.

Username
Administrator

Password

OK Cancel

After you have joined the AD domain, you can log in to it from your workstation using the display manager of your desktop or the console.

5.4 Logging In to an AD Domain

Provided your machine has been configured to authenticate against Active Directory and you have a valid Windows user identity, you can log in to your machine using

the AD credentials. Login is supported for both desktop environments (GNOME and KDE), the console, SSH, and any other PAM-aware application.

BAKHO: Offline Authentication

supports offline authentication, allowing you to remain logged in to your client machine even if the client machine is disconnected from the network.

5.4.1 GDM and KDM

To authenticate a GNOME client machine against an AD server, proceed as follows:

- 1 Select the domain.
- 2 Enter your Windows username and press **Enter**.
- 3 Enter your Windows password and press **Enter**.

To authenticate a KDE client machine against an AD server, proceed as follows:

- 1 Select the domain.
- 2 Enter your Windows username.
- 3 Enter your Windows password and press **Enter**.

If configured to do so, creates a user home directory on the local machine on the first login of each AD authenticated user. This allows you to benefit from the AD support of while still having a fully functional Linux machine at your disposal.

5.4.2 Console Login

You can not only log in to the AD client machine using a graphical front-end, but also by using the text-based console login or by using SSH.

To log in to your AD client from a console, enter *DOMAIN\user* at the `login:` prompt and provide the password.

To remotely log in to your AD client machine using SSH, proceed as follows:

- 1 At the login prompt, enter:

```
ssh DOMAIN\user@hostname
```

The \ domain and login delimiter is escaped with another \ sign.

- 2 Provide the user's password.

5.5 Changing Passwords

has the ability to help a user choose a suitable new password that meets the corporate security policy. The underlying PAM module retrieves the current password policy settings from the domain controller, informing the user about the specific password quality requirements a user account typically has, by means of a message on login. Like its Windows counterpart, presents a message describing:

- Password history settings
- Minimum password length requirements
- Minimum password age
- Password complexity

The password change process cannot succeed unless all requirements have been successfully met. Feedback about the password status is given both through the display managers and the console.

GDM and KDM provide feedback about password expiration and prompt for new passwords in an interactive mode. To change passwords in the display managers, just provide the password information when prompted.

To change your Windows password, you can use the standard Linux utility, `passwd`, instead of having to manipulate this data on the server. To change your Windows password, proceed as follows:

- 1 Log in at the console.
- 2 Enter `passwd`.
- 3 Enter your current password when prompted.

- 4** Enter the new password.
- 5** Reenter the new password for confirmation. If your new password does not comply with the policies on the Windows server, you are prompted for another password.

To change your Windows password from the GNOME desktop, proceed as follows:

- 1** Click the *Computer* icon on the left edge of the panel.
- 2** Select *Control Center*.
- 3** From the *Personal* section, select *Change Password*.
- 4** Enter your old password.
- 5** Enter and confirm the new password.
- 6** Leave the dialog with *Close* to apply your settings.

To change your Windows password from the KDE desktop, proceed as follows:

- 1** Select *Personal Settings* from the main menu.
- 2** Select *Security & Privacy*.
- 3** Click *Password & User Account*.
- 4** Click *Change Password*.
- 5** Enter your current password.
- 6** Enter and confirm the new password and apply your settings with *OK*.
- 7** Leave the *Personal Settings* with *File > Quit*.

Сетевая аутентификация при помощи Kerberos

Открытая сетевая архитектура не предоставляет способа удостовериться в том, что рабочая станция идентифицирует пользователей должным образом, за исключением обычного механизма паролей. В большинстве случаев пользователь должен вводить пароль при каждом обращении к службе внутри сети. Kerberos предоставляет метод аутентификации, с помощью которого пользователь регистрируется только один раз, после чего ему доступны все сетевые ресурсы до конца сессии. Следующие требования должны быть соблюдены для безопасности сети:

- Все пользователи должны подтверждать свою подлинность и никто не должен использовать чужие данные.
- Убедиться, что каждый сервер сети также подтверждает свою подлинность. Иначе злоумышленник может подменить сервер и получить доступ к отправляемой ему конфиденциальной информации. Эта концепция названа *взаимной аутентификацией*, потому что клиент аутентифицируется у сервера и наоборот.

Kerberos поможет Вам соблюсти эти условия предлагая аутентификацию с устойчивым шифрованием. Здесь мы обсудим только основные принципы Kerberos. За подробной технической инструкцией обратитесь к документации Kerberos.

6.1 Терминология Kerberos

Следующий глоссарий содержит основные определения Kerberos.

верительные данные

Пользователи или клиенты должны представить какие-либо верительные данные, разрешающие им запрашивать службы. Kerberos известно два вида верительных данных: билеты и аутентификаторы.

билет

Билет - это верительные данные, используемые клиентом для аутентификации на сервере, у которого клиент запрашивает службу. Он содержит имя сервера, имя клиента, адрес клиента в интернете, дату, время жизни и произвольный ключ сессии. Все эти данные зашифрованы с использованием ключа сервера.

аутентификатор

Объединяясь с билетом, аутентификатор используется для доказательства того, что клиент предоставивший билет действительно тот, за кого он себя выдает. аутентификатор создается с использованием имени клиента, IP-адреса рабочей станции и текущего времени рабочей станции. Все это зашифровано ключом сессии, известным только клиенту и соответствующему серверу. В отличие от билета, аутентификатор может быть использован только один раз. Клиент может создавать аутентификаторы самостоятельно.

принципал

Kerberos принципал это уникальная сущность (пользователь или служба), которой можно ассигновать билет. Принципал состоит из следующих компонентов:

- **Основа (primary)** — первая часть принципала, которая может совпадать с именем пользователя, в случае использования пользователя в качестве принципала.
- **Экземпляр (instance)** — некая необязательная информация, характеризующая основу. Эта строка отделена от исходной символом /.
- **Область (realm)** — определяет вашу область Kerberos. Обычно область соответствует доменному имени, написанному в верхнем регистре.

взаимная аутентификация

Kerberos проверяет подлинность как пользователя, так и клиента. Они разделяют между собой ключ сессии, который используется для безопасного взаимодействия.

ключ сессии

Ключи сессии — это временные защищенные ключи, созданные Kerberos. Они известны клиенту и используются для шифрования данных, передаваемых между клиентом и сервером, для которого он запросил и получил билет.

повторное воспроизведение

Почти все сообщения, посылаемые по сети, могут быть подслушаны, украдены и отосланы повторно. В контексте Kerberos будет очень опасно, если атакующему удастся получить Ваш запрос на обслуживание содержащий Ваш билет и аутентификатор. Атакующий может попытаться послать их повторно (*replay*), чтобы представиться Вами. Однако, в Kerberos реализовано несколько механизмов чтобы решить эту проблему.

сервер и служба

Служба используется для обозначения определенного действия, которое нужно выполнить. Стоящий за этим действием процесс называется *сервером*.

6.2 Как работает Kerberos

Kerberos часто называют сторонней доверенной службой аутентификации, это означает, что все клиенты доверяют Kerberos решение о подлинности друг друга. Kerberos хранит базу данных всех своих пользователей и их защищенных ключей.

Чтобы убедиться в правильной работе Kerberos, запустите сервер аутентификации и предоставляющий билеты сервер на выделенной машине. Убедитесь, что только администратор имеет физический и сетевой доступ к этой машине. Снизьте количество запущенных на нем (сетевых) служб до абсолютного минимума — не запускайте даже `sshd`.

6.2.1 Первый контакт

Ваш первый контакт с Kerberos напоминает стандартную процедуру сетевого логина. Введите Ваше имя пользователя. Эта часть информации и имя службы, вы-

дающей билеты, посылаются серверу аутентификации (Kerberos). Если сервер аутентификации знает Вас, он генерирует случайный ключ сессии для дальнейшего использования между Вашим клиентом и предоставляющим билеты сервером. Далее сервер аутентификации готовит билет для сервера, предоставляющего билеты. Билет содержит следующую информацию (все зашифровано ключом сессии, который знают только сервер аутентификации и сервер, предоставляющий билеты):

- Имена обоих, клиента и сервера, предоставляющего билеты
- Текущее время
- Время жизни, назначенное этому билету
- IP-адрес клиента
- Вновь созданный ключ сессии

Этот билет посылается клиенту вместе с ключом сессии, опять-таки в зашифрованном виде, но на этот раз защищенным ключом клиента. Защищенный ключ известен только Kerberos и клиенту, потому что он создается на основе пароля пользователя. После получения этого ответа клиентом, у Вас запрашивается пароль. Этот пароль преобразуется в ключ, которым можно расшифровать пакет, отправленный сервером аутентификации. Пакет «разворачивается» и пароль с ключом стираются из памяти рабочей станции. Пока время жизни билета, используемого для получения других билетов не истечет, Ваша рабочая станция может подтверждать свою подлинность.

6.2.2 Запрос службы

Для запроса службы у любого сервера сети, клиентское приложение должно подтвердить подлинность перед сервером. Поэтому приложение генерирует аутентификатор. Аутентификатор состоит из следующих компонентов:

- Принципал клиента
- IP-адрес клиента
- Текущее время
- Контрольная сумма (выбранная клиентом)

Вся эта информация зашифрована с использованием ключа сессии, которую клиент уже получил от специального сервера. Аутентификатор и билет для сервера посылаются серверу. Сервер использует свою копию ключа сессии, чтобы расшифровать аутентификатор, который предоставляет всю необходимую информацию о клиенте, запрашивающем службу, и сравнивает ее с информацией билета. Сервер также проверяет, принадлежат ли аутентификатор и билет одному и тому же клиенту.

Если не предпринять мер безопасности на стороне сервера, этот шаг может быть идеальной целью для атак повторного воспроизведения. Кто-нибудь может попытаться переслать запрос, украденный ранее из сети. Для предотвращения этого, сервер не принимает запросов с билетом и временной меткой уже принятыми ранее. В дополнение к этому, запросы с временной меткой сильно отличающейся от времени получения запроса, игнорируются.

6.2.3 Взаимная аутентификация

Аутентификация Kerberos может быть использована в обоих направлениях. Она касается не только удостоверения своей подлинности клиентом. Сервер тоже должен аутентифицировать себя перед клиентом, запрашивая его службу. Таким образом, он тоже посылает аутентификатор. Он добавляет единицу к контрольной сумме клиентского аутентификатора и шифрует результат разделяемым им с клиентом ключем сессии. Клиент принимает этот ответ, как доказательство подлинности сервера и они начинают взаимодействие.

6.2.4 Получение билетов — соединение с серверами

Билеты создаются для использования с конкретным сервером. Это означает, что при запросе к новой службе Вам нужен новый билет. Kerberos реализует механизм получения билетов для отдельных серверов. Эта служба называется «службой получения билетов». Служба получения билетов использует протоколы доступа описанные ранее. Каждый раз, когда приложению нужен новый билет, оно соединяется с сервером получения билетов. Этот запрос состоит из следующих компонентов:

- Запрашиваемый принципал
- Билет на получение билета

- Аутентификатор

Как и любой другой сервер, сервер получения билетов проверяет билет на получение билета и аутентификатор. Если он считает их достоверными, то создает новый ключ сессии для использования между клиентом и новым сервером. Затем создается билет для нового сервера, который содержит следующую информацию:

- Принципал клиента
- Принципал сервера
- Текущее время
- IP-адрес клиента
- Вновь созданный ключ сессии

Новый билет имеет время жизни, равное оставшемуся времени жизни билета на получение билета или времени жизни по умолчанию для этой службы. Из двух этих значений выбирается наименьшее. Клиент получает от службы получения билетов этот билет и ключ сессии, зашифрованные тем же ключом, которым был зашифрован билет на получение билета. Клиент может расшифровать ответ без использования пароля пользователя, при запросе новой службы. Таким образом, Kerberos может получать билет за билетом, не беспокоя пользователя.

6.2.5 Совместимость с Windows 2000

Windows 2000 содержит реализацию Kerberos 5 от Microsoft. использует MIT реализацию Kerberos 5, полезная информация и руководство можно найти в документации MIT Раздел 6.5, «Дополнительная информация» (стр. 113).

6.3 Пользовательский взгляд на Kerberos

В идеальном случае, единственный контакт пользователя с Kerberos происходит во время входа в систему. Процесс входа систему включает создание билета на получение билета. При выходе все пользовательские билеты Kerberos уничтожа-

ются автоматически, что затрудняет кому-либо возможность представиться этим пользователем. Автоматическое истечение билетов может привести к несколько неловкой ситуации, когда пользовательская сессия длится дольше, чем максимальная продолжительность жизни билета на выдачу билета (разумное ее значение равно 10 часам). Однако пользователь может получить новый билет на получение билета, запустив `kinit`. Введите пароль снова и Kerberos получит доступ к требуемым службам без дополнительной аутентификации. Для получения списка всех билетов, полученных для Вас Kerberos, запустите `klist`.

краткий список всех приложений, использующих аутентификацию Kerberos Эти приложения можно найти в директории `/usr/lib/mit/bin` и `/usr/lib/mit/sbin` после установки пакета `krb5-apps-clients`. Все они имеют полную функциональность своих UNIX и Linux братьев, плюс дополнительный бонус прозрачной авторизации, управляемой Kerberos:

- `telnet`, `telnetd`
- `rlogin`
- `rsh`, `rcp`, `rshd`
- `ftp`, `ftpd`
- `ksu`

Вам больше не нужно вводить свой пароль для использования этих приложений, поскольку Kerberos уже подтвердил Вашу подлинность. `ssh` и собранный с поддержкой Kerberos, может даже перенаправлять все полученные для одной рабочей станции билеты на другую. Если Вы используете `ssh` для входа на другую рабочую станцию, `ssh` проверяет, чтобы зашифрованное содержание билетов исправлялось в соответствии с ситуацией. Простого копирования билетов между рабочими станциями недостаточно, поскольку билеты содержат информацию, относящуюся к рабочей станции (IP-адрес). XDM, GDM и KDM также предлагают поддержку Kerberos. Дальнейшую информацию о сетевых приложениях Kerberos можно найти в *Руководстве пользователя по Kerberos V5 UNIX*, доступному по адресу <http://web.mit.edu/kerberos>.

6.4 Инсталляция и администрирование Kerberos

Окружение Kerberos состоит из нескольких различных компонентов. Центр распределения ключей (KDC - key distribution center) содержит центральную базу данных со всеми данными Kerberos. Все клиенты полагаются на KDC в надлежащей сетевой аутентификации. Как KDC, так и клиенты нуждаются в настройке для соответствия Вашей конфигурации:

Общая подготовка

Проверьте свою конфигурацию сети и убедитесь, что она соответствует минимальным требованиям, описанным в Раздел 6.4.1, «Сетевая топология Kerberos» (стр. 92). Выберите подходящую область для Вашей конфигурации Kerberos, см. Раздел 6.4.2, «Выбор областей Kerberos» (стр. 93). Тщательно настройте компьютер, предназначенный для KDC, с применением максимальных мер безопасности, Раздел 6.4.3, «Настройка KDC» (стр. 94). Настройте надежный источник времени в сети, чтобы удостовериться, что все билеты содержат валидные временные метки, см. Раздел 6.4.4, «Настройка синхронизации времени» (стр. 95).

Основная конфигурация

Настройте KDC и клиентов согласно Раздел 6.4.5, «Настройка KDC» (стр. 96) и Раздел 6.4.6, «Конфигурация клиентов Kerberos» (стр. 99). Разрешите удаленное администрирование для Вашей службы Kerberos, чтобы Вам не нужен был физический доступ к компьютеру с KDC, см. Раздел 6.4.7, «Настройка удаленного администрирования Kerberos» (стр. 104). Создайте принципалов служб для каждой службы в Вашей области, см. Раздел 6.4.8, «Creating Kerberos Service Principals» (стр. 106).

Включение аутентификации Kerberos

Различные службы в Вашей сети могут использовать Kerberos. Чтобы добавить проверку пароля Kerberos приложениям, использующим PAM, выполните действия, описанные в Раздел 6.4.9, «Включение поддержки PAM в Kerberos» (стр. 108). Чтобы настроить SSH или LDAP на аутентификацию через Kerberos, следуйте Раздел 6.4.10, «Настройка SSH для аутентификации с помощью Kerberos» (стр. 108) и Раздел 6.4.11, «Использование LDAP и Kerberos» (стр. 109).

6.4.1 Сетевая топология Kerberos

Любое окружение с Kerberos для полноценной работы должно отвечать следующим требованиям:

- Настройте DNS сервер для разрешения имен внутри Вашей сети, чтобы клиенты и серверы могли найти друг друга. Обратитесь к Глава 11, *The Domain Name System* (↑Содержание) за информацией по настройке DNS.
- Настройте сервер времени в Вашей сети. Использование точных временных меток существенно для работы Kerberos, поскольку валидные билеты Kerberos должны содержать корректные временные метки. Обратитесь к Глава 13, *Time Synchronization with NTP* (↑Содержание) за информацией по настройке NTP.
- Настройте центр распространения ключей (KDC) как центральную часть архитектуры Kerberos. Он содержит базу данных Kerberos. Используйте максимально высокую политику безопасности на этой машине для предотвращения любых атак, которые могут подвергнуть риску всю Вашу сетевую инфраструктуру.
- Настройте клиентские машины на использование аутентификации Kerberos.

Следующее изображение поясняет простой пример сети с минимумом компонентов, необходимых для построения инфраструктуры Kerberos. В зависимости от размеров и топологии Вашей сети, Ваша инсталляция может отличаться.

Рисунок 6.1 *Сетевая топология Kerberos*

ПОДСКАЗКА: Настройка маршрутизации подсети

Для конфигурации подобной Рисунок 6.1, «Сетевая топология Kerberos» (стр. 93), настройте маршрутизацию между двумя подсетями (192.168.1.0/24 и 192.168.2.0/24). Обратитесь к Раздел “Configuring Routing” (Глава 9, *Basic Networking*, ↑Содержание) за дальнейшей информацией о настройке маршрутизации с помощью YaST.

6.4.2 Выбор областей Kerberos

Домен, в котором работает инсталляция Kerberos называется областью и идентифицируется именем, таким как `EXAMPLE.COM` или просто `ACCOUNTING`. Kerberos чувствителен к регистру, поэтому `example.com` является областью, отличной от `EXAMPLE.COM`. Используйте регистр, который Вам нравится. Од-

нако, общепринятая практика — использовать верхний регистр для имен областей.

Хорошая идея использовать Ваше имя домена из DNS (или поддомена, такого как ACCOUNTING.EXAMPLE.COM). Как показано ниже, Вы можете облечить себе жизнь как администратору, если настроите клиенты Kerberos на поиск KDC и других служб Kerberos с помощью DNS. При этом очень полезно указать имя области как поддомен Вашего доменного имени DNS.

в отличие от пространства имен DNS, Kerberos не иерархичен. Вы не можете подчинить области с именем EXAMPLE.COM, две «подобласти» с именами DEVELOPMENT и ACCOUNTING, так, чтобы подчиненные области наследовали принципалов у EXAMPLE.COM. Вместо этого, Вы получите три разных области, для которых Вам придется настроить межобластную аутентификацию, чтобы пользователи одной области могли взаимодействовать с серверами и пользователями другой области.

Для простоты давайте предположим, что Вы настраиваете всего одну область для всей организации. До конца этой секции для нее во всех примерах будет использоваться имя EXAMPLE.COM.

6.4.3 Настройка KDC

Первый шаг, необходимый для использования Kerberos — это компьютер, который будет работать как центр распространения ключей или, для краткости, KDC. Этот компьютер содержит всю базу данных пользователей Kerberos с паролями и другой информацией.

KDC это наиболее важная часть вашей инфраструктуры безопасности: если кто-либо проникнет в нее, все учетные записи пользователей и вся защищенная Kerberos часть сети будет скомпрометирована. Взломщик с доступом к базе данных Kerberos может представиться любым принципалом из этой базы. Сделайте настройки безопасности для этого компьютера максимально строгими:

- 1 Поестите этот сервер в физически безопасное место, такое как закрытая серверная комната, к которой имеет доступ ограниченное количество людей.
- 2 Не запускайте на ней сетевых приложений за исключением KDC. Это относится и к серверам и к клиентам: например, KDC не должен импортировать файловые системы по NFS или использовать DHCP для получения сетевой конфигурации.

- 3 Установите на него минимальное количество приложений, затем проверьте список установленных пакетов и удалите все ненужные пакеты. Это относится к серверам, таким как `inetd`, `portmap` и `cups`, а также ко всему работающему под X. Даже установка SSH сервера должна рассматриваться как потенциальная уязвимость.
- 4 На этой машине не требуется графического входа в систему, поскольку X сервер это потенциальная угроза безопасности. Kerberos предоставляет собственный интерфейс администрирования.
- 5 Настройте `/etc/nsswitch.conf` для использования только локальных файлов для поиска пользователей и групп. Измените строки для `passwd` и `group` следующим образом:

```
passwd:      files
group:       files
```

Измените файлы `passwd`, `group` и `shadow` в `/etc`, удалив строки начинающиеся с символа `+` (они нужны для поиска NIS).
- 6 Запретите все аккаунты пользователей, кроме `root`, заменив в `/etc/shadow` хеши паролей на символ `*` или `!`.

6.4.4 Настройка синхронизации времени

Для успешного использования Kerberos, убедитесь, что все системные часы внутри Вашей организации синхронизируются в определенном пределе. Это важно, поскольку Kerberos защищен против повторной отправки верительных данных. Атакующий может иметь возможность отслеживать верительные данные Kerberos по сети и повторно использовать их для атаки сервера. Kerberos применяет несколько видов защиты, чтобы предотвратить это. Одним из них являются временные метки билетов. Сервер, получивший билет с временной меткой, которая отличается от текущего времени, не принимает этот билет.

Kerberos имеет некоторый люфт при сравнении временных меток. Тем не менее, часы компьютера могут быть неточны при отсчете времени — часто приходится слышать о получасовом отклонении от точного времени, набранном за неделю. По этой причине, настройте все компьютеры сети на синхронизацию с центральным источником времени.

Самым простым способом будет установка на один из компьютеров сервера NTP и синхронизация всех своих часов клиентами с этим сервером. Это можно сделать либо запустив NTP демон в режиме клиента на всех этих компьютерах или при помощи ежедневного запуска `ntpdate` на всех клиентах (это решение скорее всего будет работать только для небольшого количества клиентов). KDC тоже должен синхронизироваться с общим источником времени. Поскольку запуск NTP демона на этом компьютере будет представлять угрозу безопасности, хорошей идеей будет запуск `ntpdate` посредством записи в кроне. Что бы настроить клиент NTP, следуйте инструкциям Раздел “Configuring an NTP Client with YaST” (Глава 13, *Time Synchronization with NTP*, ↑Содержание).

Другой способом безопасно использовать службу времени с NTP-демоном - закрепить один источник точного времени за выделенным NTP-сервером и другой источник точного времени за KDC.

Можно также изменить максимальное отклонение, допускаемое Kerberos при проверке временных меток. Это значение (называемое *разброс часов - clock skew*) может быть установлено в файле `krb5.conf` как описано в «Изменения разброса часов» (стр. 104).

6.4.5 Настройка KDC

Эта секция поясняет начальную настройку и установку KDC, включая создание административного принципала. Эта процедура состоит из нескольких шагов:

- 1 Установите RPM-пакеты** На машине, предназначенной для KDC, установите следующие пакеты приложений: `krb5`, `krb5-server` и `krb5-client`.
- 2 Измените файлы конфигурации** Файлы конфигурации `/etc/krb5.conf` и `/var/lib/kerberos/krb5kdc/kdc.conf` должны быть изменены в соответствии с Вашим сценарием. Эти файлы содержат всю информацию о KDC.
- 3 Создайте базу данных Kerberos** Kerberos хранит базу идентификаторов всех принципалов и их защищенные ключи, необходимые для аутентификации. Обратитесь к Раздел 6.4.5.1, «Setting Up the Database» (стр. 97) за подробной информацией.
- 4 Измените файлы ACL: добавьте администраторов** Расположенной на KDC базой данных Kerberos можно управлять удаленно. Для предотвращения

подделки базы данных неавторизованными принципами, Kerberos использует списки управления доступом. Вы должны явно разрешить удаленный доступ для администрирующего принципа чтобы он мог управлять этой базой данных. Файл ACL Kerberos находится в `/var/lib/kerberos/krb5kdc/kadm5.acl`. Обратитесь к Раздел 6.4.7, «Настройка удаленного администрирования Kerberos» (стр. 104) за подробной информацией.

- 5 Измените базу данных Kerberos: добавьте администраторов** Вам нужен как минимум один администрирующий принципал для запуска и управления Kerberos. Этот принципал должен быть добавлен перед запуском KDC. Обратитесь к Раздел 6.4.5.2, «Создание принципалов» (стр. 98) за подробностями.
- 6 Запустите демон Kerberos** После того, как KDC установлен и настроен, запустите демон Kerberos для обслуживания Kerberos Вашей области. Подробности можно найти в Раздел 6.4.5.3, «Запуск KDC» (стр. 99).
- 7 Создайте для себя принципала** Вам нужен принципал для себя. Подробное описание доступно в Раздел 6.4.5.2, «Создание принципалов» (стр. 98).

6.4.5.1 Setting Up the Database

Ваш следующий шаг - инициализировать базу данных, в которой Kerberos хранит всю информацию о принципах. Задайте мастер-ключ, который будет использован для защиты Вашей базы данных от случайного раскрытия (например, при создании ее резервной копии). Мастер ключ создается из пароля и хранится в файле, называемом спрятанный (англ. stash) файл. Поэтому Вам не требуется вводить пароль при каждом перезапуске KDC. Убедитесь, что Вы выбрали хороший пароль, например фразу из книги, открытой на случайной странице.

Когда Вы сохраняете резервные копии базы данных Kerberos на ленту, (`/var/lib/kerberos/krb5kdc/principal`), не включайте в резервную копию спрятанный файл (находящийся в `/var/lib/kerberos/krb5kdc/.k5.EXAMPLE.COM`). Иначе, каждый, кто имеет доступ к резервной копии, сможет дешифровать базу данных. Поэтому храните копию пароля в безопасном месте, на случай, если Вам придется восстанавливать базу данных после сбоя.

Для создания спрятанного файла и базы данных, запустите:

```
kdb5_util create -r EXAMPLE.COM -s
```

Вы увидите следующую информацию:

```
Initializing database '/var/lib/kerberos/krb5kdc/principal' for realm
'EXAMPLE.COM',
master key name 'K/M@EXAMPLE.COM'
You will be prompted for the database Master Password.
It is important that you NOT FORGET this password.
Enter KDC database master key: ❶
Re-enter KDC database master key to verify: ❷
```

❶ Введите мастер-пароль.

❷ Введите пароль снова.

Для проверки используйте команду `list`:

```
kadmin.local
```

```
kadmin> listprincs
```

Вы увидите несколько принципов в базе данных, которые предназначены для внутреннего пользования Kerberos:

```
K/M@EXAMPLE.COM
kadmin/admin@EXAMPLE.COM
kadmin/changepw@EXAMPLE.COM
krbtgt/EXAMPLE.COM@EXAMPLE.COM
```

6.4.5.2 Создание принципов

Создайте для себя двух принципов Kerberos: обычного принципа для ежедневной работы и другого - для администрирования Kerberos. Допустим, Ваше имя пользователя `geeko`, значит Вам необходимо выполнить следующие действия:

```
kadmin.local
```

```
kadmin> ank geeko
```

Вы увидите следующее:

```
geeko@EXAMPLE.COM's Password: ❶
Verifying password: ❷
```

❶ Type `geeko's password`.

❶ Type `geeko's password` again.

Далее создайте другого принципа с именем `geeko/admin` набрав `ank geeko/admin` в `kadmin`. Суффикс `admin` после Вашего имени пользователя — это *роль*. Используйте эту роль при администрировании базы данных Kerberos. Пользователь может иметь несколько ролей различного назначения. Роли — это различные аккаунты с похожими именами.

6.4.5.3 Запуск KDC

Запустите демоны KDC и `kadmin`. Для ручного запуска демонов введите `rckrb5kdc start` и `rkadmind start`. Также удостоверьтесь, что KDC и `kadmind` запускаются автоматически при перезапуске сервера с помощью команд `insserv krb5kdc` и `insserv kadmind` или используйте Настройки уровня запуска YaST.

6.4.6 Конфигурация клиентов Kerberos

После того, как развернута вспомогательная инфраструктура (DNS, NTP) и KDC настроен и запущен, настройте клиентские компьютеры. Для настройки Kerberos клиента Вы можете использовать YaST либо один из двух методов ручной настройки, описанных ниже.

6.4.6.1 Конфигурация клиента Kerberos при помощи YaST

Вместо ручного изменения файлов конфигурации при настройке клиента Kerberos, позвольте YaST сделать это за Вас. Вы можете произвести настройку во время инсталляции, либо на уже установленной системе. Она выполняется следующим образом:

- 1 Войдите в систему как `root` и выберите *Сетевые службы > Клиент Kerberos*.
- 2 Выберите *Использовать Kerberos*.
- 3 Для настройки клиента Kerberos, использующего DNS выполните следующие действия:
 - 3а Включите *Использовать DNS для получения данных настроек во время работы* и проверьте *Основные настройки Kerberos* отображенные на экране.

ПРИМЕЧАНИЕ: Использование поддержки DNS

Опция *Использовать DNS* не может быть выбрана, если DNS сервер не предоставляет таких данных.

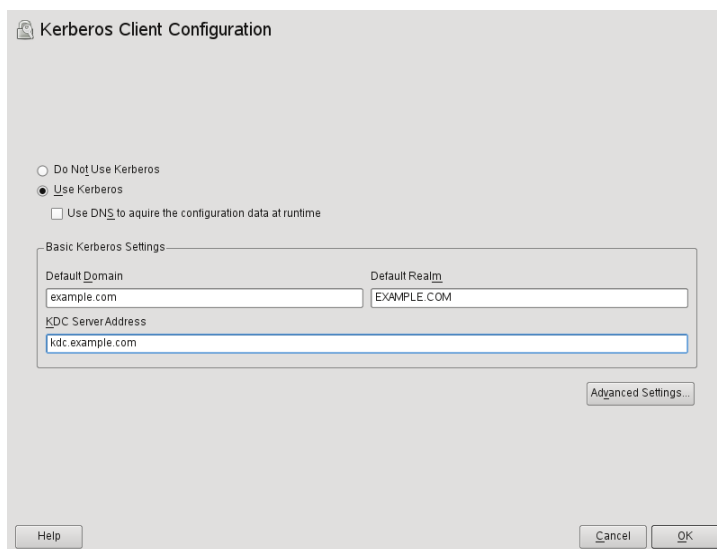
3b Нажмите *Дополнительные настройки* для конфигурирования параметров билетов, поддержки OpenSSH, синхронизации времени и экспертных настроек PAM.

4 Для настройки статического клиента Kerberos:

4a Установите в *Домен по умолчанию*, *Область по умолчанию* и *Адрес сервера KDC* значения, соответствующие Вашей конфигурации.

4b Нажмите *Дополнительные настройки* для конфигурирования параметров билетов, поддержки OpenSSH, синхронизации времени и экспертных настроек PAM.

Рисунок 6.2 YaST: Базовая настройка клиента Kerberos



Для конфигурирования параметров билетов в диалоге *Дополнительные настройки*:

- Укажите *Время действия по умолчанию* и *Обновляемое время действия по умолчанию* в днях, часах или минутах, (используя единицы измерения *d*, *h* или *t*, без разделителя между значением и единицей измерения).
- Для переадресации Вашей полной идентификации (для использования Ваших билетов на других хостах), выберите значение в поле *Переадресация*.

- Разрешите передачу некоторых билетов, выбрав значение в поле *Проксирование*.
- Включите поддержку аутентификации Kerberos для своего OpenSSH клиента, отметив соответствующее поле. После этого клиент будет использовать билеты Kerberos для авторизации на сервере SSH.
- Отключите использование аутентификации Kerberos для некоторых пользовательских аккаунтов введя в поле *Минимальный UID* значение необходимое для активации авторизации. Например, Вы можете пожелать исключить администратора системы (`root`).
- Используйте *Разброс часов*, чтобы установить значение максимального отклонения между временными метками и системным временем Вашей машины.
- Для синхронизации системного времени с NTP сервером, Вы также можете настроить NTP клиент, выбрав *Настройка NTP*, которая откроет диалог YaST Расширенная настройка NTP, описываемый в Раздел “Configuring an NTP Client with YaST” (Глава 13, *Time Synchronization with NTP*, ↑Содержание). После завершения конфигурации, YaST выполнит все необходимые изменения и клиент Kerberos будет готов к использованию.

Рисунок 6.3 YaST: Дополнительные настройки клиента Kerberos

Advanced Kerberos Client Configuration

PAM Settings Expert PAM Settings PAM Services

Ticket Attributes

Default Lifetime: 1d

Default Renewable Lifetime: 1d

Forwardable: All services

Proxiable: No services

☐ Kerberos Support for OpenSSH Client

☒ Ignore Unknown Users

Minimum UID: 1

Clock Skew: 300

NTP Configuration...

Configure User Data

Help Cancel OK

За подробной информацией о конфигурации вкладок *Настройки эксперта РАМ* и *Службы РАМ*, обратитесь к официальной документации Раздел 6.5, «Дополнительная информация» (стр. 113) и man-странице `man 5 krb5.conf`, входящей в пакет `krb5-doc`.

6.4.6.2 Ручная настройка клиентов Kerberos

Существует два основных подхода настройки Kerberos: статическая конфигурация с использованием файла `/etc/krb5.conf` или динамическая конфигурация при помощи DNS. Используя конфигурацию DNS, приложения Kerberos пытаются обнаружить службы KDC используя записи DNS. Для статической конфигурации, добавьте адрес Вашего KDC сервера в `krb5.conf` (и обновляйте файл при переносе KDC или других изменениях в Вашей области).

Конфигурация, основанная на DNS, является более гибкой и сокращает время на настройку каждого клиента. Однако, она требует, чтобы имя Вашей области либо совпадало с Вашим доменом DNS или его поддоменом. Настройка Kerberos через DNS также создает небольшую угрозу безопасности — атакующий может значительно нарушить работу Вашей инфраструктуры, с помощью DNS (например, отключив сервер имен подменив записи DNS и т.д.). Однако, в самом худшем случае это вызовет отказ в обслуживании. Подобный сценарий атаки применим и к статической конфигурации, если только Вы не введете IP-адреса в файл `krb5.conf` вместо имен хостов.

Статическая конфигурация

Один из способов настройки Kerberos заключается в редактировании `/etc/krb5.conf`. Файл, установленный по умолчанию, содержит различные примеры записей. Удалите их все перед началом настройки. `krb5.conf` состоит из нескольких секций (строф), каждая определяется именем в квадратных скобках вот [так].

Для настройки Ваших клиентов Kerberos, добавьте следующую строфу в `krb5.conf` (где `kdc.example.com` — имя хоста KDC):

```
[libdefaults]
    default_realm = EXAMPLE.COM

[realms]
    EXAMPLE.COM = {
        kdc = kdc.example.com
        admin_server = kdc.example.com
```

```
}
```

Строка `default_realm` устанавливает область по умолчанию для приложений Kerberos. если у Вас областей несколько, просто добавьте дополнительные выражения в секцию `[realms]`.

Также в этот файл нужно добавить выражение, которое будет сообщать приложениям, как отобразить имена хостов на область. Например, при подключении к удаленному компьютеру, библиотеке Kerberos требуется знать, в какой области размещен этот хост. Это должно быть задано в секции `[domain_realms]`:

```
[domain_realm]
    .example.com = EXAMPLE.COM
    www.foobar.com = EXAMPLE.COM
```

Это сообщает библиотеке, что все хосты в доменах DNS `example.com` находятся в области Kerberos `EXAMPLE.COM`. В дополнение, один внешний хост с именем `www.foobar.com` должен тоже считаться членом области `EXAMPLE.COM`.

Конфигурация при помощи DNS

Конфигурация Kerberos на основе DNS плотно использует записи SRV. Обратитесь к (*RFC2052*) *DNS RR for specifying the location of services* на сайте <http://www.ietf.org>.

Имя записи SRV, с точки зрения Kerberos, всегда задано в формате `_service._proto.realm`, где `realm` — это область Kerberos. Доменные имена в DNS не чувствительны к регистру, поэтому чувствительные к регистру области Kerberos realms будут нарушены при использовании данного метода конфигурации. `_service` — это имя службы (например, при попытке связаться с KDC или службой паролей используются различные имена). `_proto` может быть либо `_udp`, либо `_tcp`, но не все службы поддерживают оба протокола.

Часть данных ресурса SRV состоит из значения приоритета, веса, номера порта и имени хоста. Приоритет определяет порядок, в котором должны перебираться хосты (меньшие значения указывают на более высокий приоритет). Значение веса необходимо для поддержки балансирования нагрузки между серверами с равным приоритетом. Возможно, Вам не потребуется их использовать, в этом случае просто установите оба значения в ноль.

В настоящее время MIT Kerberos ищет следующие имена при поиске служб:

_kerberos

Определяет размещение демона KDC (сервера аутентификации и выдачи билетов). Обычно выглядит так:

```
_kerberos._udp.EXAMPLE.COM. IN SRV 0 0 88 kdc.example.com.  
_kerberos._tcp.EXAMPLE.COM. IN SRV 0 0 88 kdc.example.com.
```

_kerberos-adm

Определяет размещение службы удаленного администрирования. Типичные записи выглядят так:

```
_kerberos-adm._tcp.EXAMPLE.COM. IN SRV 0 0 749 kdc.example.com.
```

Поскольку `kadmind` не поддерживает UDP, записи `_udp` быть не должно.

Как и при использовании файла статической конфигурации, существует механизм, сообщаящий клиентам о принадлежности заданного хоста к области `EXAMPLE.COM`, даже если он не входит в домен `DNS example.com`. Это можно сделать прикрепив запись `TXT` к `_kerberos.hostname`, как показано здесь:

```
_kerberos.www.foobar.com. IN TXT "EXAMPLE.COM"
```

Изменения разброса часов

Разброс часов — это чувствительность при приеме билетов, временные метки которых не точно соответствуют текущему времени системы. Обычно, разброс часов устанавливается в 300 секунд (пять минут). Это означает, что билет может иметь временную метку опережающую часы сервера на пять минут или отстающую от них на столько же.

При использовании NTP для синхронизации всех хостов, Вы можете уменьшить это значение до одной минуты. Разброс часов задается в файле `/etc/krb5.conf` следующим образом:

```
[libdefaults]  
    clockskew = 60
```

6.4.7 Настройка удаленного администрирования Kerberos

Чтобы добавлять и удалять принципалов из базы данных Kerberos, не имея прямого доступа к консоли KDC, укажите серверу администрирования Kerberos какие действия разрешены каким принципалам отредактировав `/var/lib/`

kerberos/krb5kdc/kadm5.acl. Файл ACL (списка контроля доступа) позволяет Вам задавать привилегии с четким их контролем. За подробностями обратитесь к man-странице `man 8 kadmin`.

Сейчас просто утановите себе привилегию администрировать базу данных добавив слкдующую строку в файл:

```
geeko/admin *
```

Замените имя пользователя `geeko` на свое. перезапустите `kadmin` для применения изменений.

У вас должна появится возможность выполнять задачи удаленного администрирования Kerberos используя утилиту `kadmin`. Сначала получите билет для своей администраторской роли и используйте его для подключения к серверу `kadmin`:

```
kadmin -p geeko/admin
Authenticating as principal geeko/admin@EXAMPLE.COM with password.
Password for geeko/admin@EXAMPLE.COM:
kadmin: getprivs
current privileges: GET ADD MODIFY DELETE
kadmin:
```

Используя команду `getprivs`, проверьте, какие у Вас привилегии. Список выше отображает полный набор привилегий.

В качестве примера, измените принципала `geeko`:

```
kadmin -p geeko/admin
Authenticating as principal geeko/admin@EXAMPLE.COM with password.
Password for geeko/admin@EXAMPLE.COM:

kadmin: getprinc geeko
Principal: geeko@EXAMPLE.COM
Expiration date: [never]
Last password change: Wed Jan 12 17:28:46 CET 2005
Password expiration date: [none]
Maximum ticket life: 0 days 10:00:00
Maximum renewable life: 7 days 00:00:00
Last modified: Wed Jan 12 17:47:17 CET 2005 (admin/admin@EXAMPLE.COM)
Last successful authentication: [never]
Last failed authentication: [never]
Failed password attempts: 0
Number of keys: 2
Key: vno 1, Triple DES cbc mode with HMAC/sha1, no salt
Key: vno 1, DES cbc mode with CRC-32, no salt
Attributes:
Policy: [none]

kadmin: modify_principal -maxlife "8 hours" geeko
Principal "geeko@EXAMPLE.COM" modified.
kadmin: getprinc joe
```

```
Principal: geeko@EXAMPLE.COM
Expiration date: [never]
Last password change: Wed Jan 12 17:28:46 CET 2005
Password expiration date: [none]
Maximum ticket life: 0 days 08:00:00
Maximum renewable life: 7 days 00:00:00
Last modified: Wed Jan 12 17:59:49 CET 2005 (geeko/admin@EXAMPLE.COM)
Last successful authentication: [never]
Last failed authentication: [never]
Failed password attempts: 0
Number of keys: 2
Key: vno 1, Triple DES cbc mode with HMAC/sha1, no salt
Key: vno 1, DES cbc mode with CRC-32, no salt
Attributes:
Policy: [none]
kadmin:
```

Это изменит максимальное время жизни билета, установив его равным восьми часам. Подробную информацию о команде `kadmin` и доступных опциях смотрите в пакете `krb5-doc`, по ссылке <http://web.mit.edu/kerberos/www/krb5-1.8/krb5-1.8.3/doc/krb5-admin.html#Kadmin%20Options>, или на man-странице `man 8 kadmin`.

6.4.8 Creating Kerberos Service Principals

До настоящего момента мы обсуждали только аутентификацию пользователей. Однако, службы совместимые с Kerberos обычно также должны аутентифицировать себя перед клиентами. Таким образом, в базе данных Kerberos должны существовать специальные служебные принципы для каждой службы работающей данной области. Например, если `ldap.example.com` предоставляет службу LDAP, вам нужен служебный принцип, `ldap/ldap.example.com@EXAMPLE.COM`, для аутентификации этой службы перед всеми клиентами.

Соглашение именования для служебных принципов *служба/имя_хоста@ОБЛАСТЬ*, где *имя_хоста* — полное имя хоста.

Действительные дескрипторы служб:

Дескриптор службы	Служба
host	Telnet, RSH, SSH

Дескриптор службы	Служба
nfs	NFSv4 (с поддержкой Kerberos)
HTTP	HTTP (с аутентификацией Kerberos)
imap	IMAP
pop	POP3
ldap	LDAP

Служебные принципы сходны с пользовательскими, но имеют существенные отличия. Основное отличие между ними заключается в том, что ключ пользовательского принципа защищен паролем — когда пользователь получает билет на предоставление билета от KDC, ему нужно ввести свой пароль для того, чтобы Kerberos смог расшифровать этот билет. Системному администратору было бы очень неудобно получать новые билеты для демона SSH каждые восемь часов или около того.

Вместо этого, ключ для расшифровки первого билета служебного принципа получается администратором от KDC всего один раз и хранится в локальном файле с именем *keytab*. Службы, такие как демон SSH, читают этот ключ и, при необходимости, используют его для получения нового билета автоматически. Файл *keytab* по умолчанию находится в `/etc/krb5.keytab`.

Для создания сервисного принципа хоста `jupiter.example.com` введите следующие команды в сессии `kadmin`:

```
kadmin -p geeko/admin
Authenticating as principal geeko/admin@EXAMPLE.COM with password.
Password for geeko/admin@EXAMPLE.COM:
kadmin: addprinc -randkey host/jupiter.example.com
WARNING: no policy specified for host/jupiter.example.com@EXAMPLE.COM;
defaulting to no policy
Principal "host/jupiter.example.com@EXAMPLE.COM" created.
```

Вместо установки пароля для нового принципа, флаг `-randkey` указывает `kadmin` создать случайный ключ. Он используется потому, что для этого принципа не нужно взаимодействие с пользователем. Это серверная учетная запись.

Затем извлеките ключ и сохраните его в локальном файле `keytab /etc/krb5.keytab`. Владелец этого файла является суперпользователь, так что Вы должны запустить команду от `root` используя оболочку `kadmin`:

```
kadmin: ktadd host/jupiter.example.com
Entry for principal host/jupiter.example.com with kvno 3, encryption type
Triple
DES cbc mode with HMAC/sha1 added to keytab WRFILE:/etc/krb5.keytab.
Entry for principal host/jupiter.example.com with kvno 3, encryption type
DES
cbc mode with CRC-32 added to keytab WRFILE:/etc/krb5.keytab.
kadmin:
```

После завершения убедитесь, что вы уничтожили администраторский билет, полученный `kinit` выше командой `kdestroy`.

6.4.9 Включение поддержки PAM в Kerberos

содержит модуль PAM с именем `pam_krb5`, который поддерживает вход в систему через Kerberos и изменение пароля. Этот модуль может использоваться как консольными приложениями: `login`, `su`, так и приложениями графической авторизации, например KDM (где пользователь вводит пароль и хочет, чтобы приложение, удостоверяющее подлинность, получило для него начальный билет Kerberos). Для настройки поддержки PAM в Kerberos, используйте следующую команду:

```
pam-config --add --krb5
```

Эта команда добавляет модуль `pam_krb5` в существующие файлы конфигурации PAM и проверяет, в правильном порядке ли он запускается. Для точной настройки использования `pam_krb5`, отредактируйте файл `/etc/krb5.conf` и добавьте приложения по умолчанию в `pam`. Подробности доступны на странице `man 5 pam_krb5`.

Модуль `pam_krb5` специально не предназначен для сетевых служб, которые принимают билеты Kerberos как часть аутентификации пользователя. Это совсем другой случай, который будет описан ниже.

6.4.10 Настройка SSH для аутентификации с помощью Kerberos

OpenSSH поддерживает аутентификацию Kerberos как для протоколов версии 1 и 2. В версии 1, есть специальные сообщения протокола передачи билетов Kerberos. Версия 2 уже не использует Kerberos напрямую, полагаясь на GSSAPI, Общее API служб безопасности (General Security Services API). Это программный интерфейс, не характерный только для Kerberos — он был создан для скрывания особенностей систем аутентификации, будь то Kerberos, система аутентификации на основе открытого ключа наподобие SPKM или другая. Однако, имеющаяся библиотека GSSAPI поддерживает только Kerberos.

Для использования sshd с аутентификацией Kerberos, отредактируйте файл `/etc/ssh/sshd_config` и установите следующие опции:

```
# These are for protocol version 1
#
# KerberosAuthentication yes
# KerberosTicketCleanup yes

# These are for version 2 - better to use this
GSSAPIAuthentication yes
GSSAPICleanupCredentials yes
```

Затем перезапустите демон SSH с помощью команды `rcsshd restart`.

Для использования аутентификации Kerberos с протоколом версии 2, включите его на стороне клиента. Это можно сделать либо в системном файле конфигурации `/etc/ssh/ssh_config` или на уровне пользователя, отредактировав `~/.ssh/config`. В любом случае, добавьте опцию `GSSAPIAuthentication yes`.

Теперь Вы сможете подключаться используя аутентификацию Kerberos. Используйте `klist` для проверки наличия действительного билета, затем подключитесь к серверу SSH. Для принудительного использования версии 1 протокола SSH, укажите опцию `-1` в командной строке.

ПОДСКАЗКА: Дополнительная информация

В файле `/usr/share/doc/packages/openssh/README.kerberos` содержит более подробное описание взаимодействия OpenSSH и Kerberos.

6.4.11 Использование LDAP и Kerberos

при использовании Kerberos, одним из способов распространения пользовательской информации (такой как ID пользователя, его группы и домашняя директория) в Вашей локальной сети, является использование LDAP. Это требует надежного механизма аутентификации, предотвращающего подмену пакетов и другие атаки. Одним из решений является использование Kerberos для взаимодействия с LDAP.

OpenLDAP реализует большинство способов аутентификации через SASL, уровень простой сессионной авторизации. SASL это сетевой протокол, созданный для аутентификации. Реализацией SASL является `cyrus-sasl`, который поддерживает различные методы аутентификации. Аутентификация Kerberos выполняется посредством GSSAPI (General Security Services API). По умолчанию плагин SASL для GSSAPI не установлен. Установите `cyrus-sasl-gssapi` при помощи YaST.

Чтобы разрешить Kerberos использование сервера OpenLDAP, создайте принципа `ldap/ldap.example.com` и добавьте его в `keytab`.

По умолчанию сервер LDAP `slapd` запущен от пользователя и группы `ldap`, в то время как файл `keytab` может читать только `root`. Поэтому, либо измените конфигурацию LDAP, чтобы сервер работал от `root` или разрешите чтение файла `keytab` группе `ldap`. Это можно автоматизировать используя скрипт запуска OpenLDAP (`/etc/init.d/ldap`) если файл `keytab` был указан в переменной `OPENLDAP_KRB5_KEYTAB` файла `/etc/sysconfig/openldap` и переменная `OPENLDAP_CHOWN_DIRS` установлена в `yes`, что уже сделано по умолчанию. Если `OPENLDAP_KRB5_KEYTAB` пуста, используется `keytab` файл `/etc/krb5.keytab` и Вы должны сами изменить права доступа согласно дальнейшей инструкции.

Для запуска `slapd` от `root`, отредактируйте `/etc/sysconfig/openldap`. Отключите переменные `OPENLDAP_USER` и `OPENLDAP_GROUP`, добавив перед ними символ комментария.

Чтобы разрешить чтение файла `keytab` группе LDAP, выполните

```
chgrp ldap /etc/krb5.keytab
chmod 640 /etc/krb5.keytab
```

Третье (и возможно наилучшее) решение — использовать с OpenLDAP отдельный файл `keytab`. Для этого запустите `kadmin` и введите следующую команду после добавления принципа `ldap/ldap.example.com`:

```
ktadd -k /etc/openldap/ldap.keytab ldap/ldap.example.com@EXAMPLE.COM
```

Затем выполните в консоли:

```
chown ldap.ldap /etc/openldap/ldap.keytab  
chmod 600 /etc/openldap/ldap.keytab
```

Чтобы указать OpenLDAP использовать отдельный файл keytab, измените переменную в файле `/etc/sysconfig/openldap`:

```
OPENLDAP_KRB5_KEYTAB="/etc/openldap/ldap.keytab"
```

И наконец, перезапустите сервер LDAP с помощью `rcldap restart`.

6.4.11.1 Использование аутентификации Kerberos для LDAP

Сейчас Вы сможете автоматически использовать утилиты, например `ldapsearch`, с аутентификацией Kerberos.

```
ldapsearch -b ou=people,dc=example,dc=com '(uid=geeko)'
```

```
SASL/GSSAPI authentication started  
SASL SSF: 56  
SASL installing layers  
[...]  
  
# geeko, people, example.com  
dn: uid=geeko,ou=people,dc=example,dc=com  
uid: geeko  
cn: Olaf Kirch  
[...]
```

Как видите, `ldapsearch` выводит сообщение о начале аутентификации через GSSAPI. Следующее сообщение крайне загадочно, но оно показывает, что *фактор уровня безопасности* (security strength factor — SSF для краткости) равен 56. (Значение 56 произвольно. Возможно его выбрали поскольку оно равно числу бит в ключе шифрования DES). Вам это говорит о том, что аутентификация GSSAPI была успешной и что шифрование используется для защиты целостности и конфиденциальности соединения с LDAP.

Аутентификация в Kerberos всегда обоюдна. Это означает, что не только Вы аутентифицируетесь на сервере LDAP, но и сервер LDAP аутентифицируется на Вашей стороне. В частности, это означает что Вы соединяетесь с требуемым сервером LDAP, а не с каким-либо фиктивной службой, предложенной Вам злоумышленником.

6.4.11.2 Аутентификация Kerberos и настройки доступа LDAP

Теперь мы разрешим каждому пользователю изменять атрибут консоли в своей записи LDAP. Допустим, запись LDAP пользователя `joe` размещена в `uid=joe,ou=people,dc=example,dc=com` и установим следующие настройки доступа в `/etc/openldap/slapd.conf`:

```
# This is required for things to work _at all_
access to dn.base="" by * read
# Let each user change their login shell
access to dn="*,ou=people,dc=example,dc=com" attrs=loginShell
        by self write
# Every user can read everything
access to *
        by users read
```

Второе выражение предоставляет аутентифицированным пользователям доступ на запись к атрибуту `loginShell` их собственной записи LDAP. Третье выражение предоставляет всем аутентифицированным пользователям доступ на чтение ко всей директории LDAP.

Теперь нам не хватает еще одной части паззла — как сервер LDAP определит, что пользователь Kerberos `joe@EXAMPLE.COM` относится к отличительному имени LDAP `uid=joe,ou=people,dc=example,dc=com`. Этот вид отображения должен быть настроен вручную с использованием директивы `saslExpr`. Для этого примера, нужно добавить в файл `slapd.conf`:

```
authz-regexp
    uid=(.*),cn=GSSAPI,cn=auth
    uid=$1,ou=people,dc=example,dc=com
```

Чтобы понять, как это работает, Вам нужно знать, что при аутентификации SASL пользователя, OpenLDAP создает отличительное имя из имени данным ему SASL (такого как `joe`) и имени метода SASL (`GSSAPI`). Результатом будет `uid=joe,cn=GSSAPI,cn=auth`.

Если настроен `authz-regexp`, он проверяет отличительное имя, созданное на основе информации SASL, используя первый аргумент как регулярное выражение. При совпадении этого регулярного выражения, имя заменяется вторым аргументом выражения `authz-regexp`. Заместитель `$1` заменяется подстрокой, совпавшей с выражением `(.*)`.

Возможны более сложные выражения. Если Ваша структура директорий более сложна или имя пользователя не является частью отличительного имени, Вы мо-

жете использовать выражения поиска для отображения отличительного имени SASL на отличительное имя пользователя.

6.5 Дополнительная информация

Официальный сайт MIT Kerberos <http://web.mit.edu/kerberos>. Здесь можно найти ссылки на любые другие ресурсы о Kerberos, включая установку Kerberos, руководства пользователя и администратора.

Документ по адресу <ftp://athena-dist.mit.edu/pub/kerberos/doc/usenix.PS> предоставляет довольно обширное описание основных принципов Kerberos, и легко читается. Также в нем содержится много информации для дальнейшего изучения Kerberos.

Официальный Kerberos FAQ доступен по адресу <http://www.nrl.navy.mil/CCS/people/kenh/kerberos-faq.html>. Книга *Kerberos — Система сетевой аутентификации* Brian Tung (ISBN 0-201-37924-4) также предлагает широкий спектр информации.

Использование сканера отпечатков пальцев

Если у Вас есть сканер отпечатков пальцев, Вы можете использовать биометрическую аутентификацию в дополнение к стандартной (посредством логина и пароля). После регистрации своих отпечатков пальцев, пользователь может войти в систему, проведя пальцем по сканеру или введя пароль. поддерживает большое количество подобных сканеров. Список поддерживаемых устройств можно найти на <http://www.freedesktop.org/wiki/Software/fprint/libfprint>

Если в результате проверки оборудования будет обнаружен интегрированный или подключенный к системе сканер отпечатков пальцев, пакеты `libfprint`, `pam_fp`, и `yast2-fingerprint-reader` будут установлены автоматически.

В настоящее время может быть зарегистрирован только один отпечаток пальца для каждого пользователя. Он хранится в `/home/login/.fprint/`.

7.1 Программы, поддерживающие биометрическую аутентификацию

РАМ модуль `pam_fp` поддерживает аутентификации распознавания отпечатка пальца в следующих случаях (хотя и не во всех случаях Вам может быть предложено аутентифицировать себя таким образом):

- Авторизация в GDM/KDM или при использовании программы login
- Блокировка экрана в GNOME/KDE
- Запуск YaST и модулей YaST
- Запуск приложений, которые требуют права root: `sudo` или `gnomesu`
- Вход в систему от имени другого пользователя при помощи `su` или `su - username`

ПРИМЕЧАНИЕ: Устройства для считывания отпечатка пальца и шифрование домашней директории

Если Вы хотите аутентифицировать себя с помощью сканера, Вы не должны использовать шифрование домашней директории (см. Глава 10, *Managing Users with YaST* (↑Вступление) для получения дополнительной информации). В противном случае аутентифицировать себя таким образом не получится, т.к. домашняя директория будет расшифрована только после успешного окончания процесса авторизации.

7.2 Управление биометрической аутентификацией через YaST

Процедура 7.1 Включение биометрической аутентификации

Для использования биометрической аутентификации Вы должны настроить РАМ. Как правило, это происходит автоматически во время установки системы, если поддерживаемый сканер был обнаружен. Если этого не произошло, Вы можете включить поддержку отпечатков пальцев в YaST следующим образом:

- 1 Запустите YaST и выберите *Оборудование > Распознавание отпечатков пальцев*.
- 2 В окне настроек, выберете *Использовать распознавание отпечатков пальцев* и нажмите *Готово* для сохранения настроек.

После этого Вы можете аутентифицировать себя с помощью сканера.

Процедура 7.2 Регистрация отпечатков пальцев

- 1 В YaST, нажмите *Безопасность и пользователи > Управление пользователями и группами* для открытия окна *User and Group Administration*. В нем отображается список пользователей и групп в системе.
- 2 Выберите пользователя, для которого Вы хотите зарегистрировать отпечаток пальца и нажмите *Edit*.
- 3 Во вкладке *Plug-Ins*, выберите the fingerprint entry и нажмите *Launch* для открытия окна *Конфигурация отпечатков пальцев*.
- 4 YaST попросит пользователя сканировать отпечаток пальца до тех пор, пока не будет сделано три четких снимка.



- 5 После того, как отпечатки были сняты, нажмите кнопку *Применить*, чтобы закрыть окно *Конфигурация отпечатков пальцев*.
- 6 Если Вы так же хотите использовать отпечатки пальцев для аутентификации при запуске YaST или YaST модулей, Вы должны настроить использование отпечатков для пользователя `root`.

Чтобы сделать это, установите фильтр *System Users* в окне *User and Group Administration*, и выберете пользователя `root`, а так же оставьте 3 четких отпечатка пальца, как описано выше.

- 7 После регистрации отпечатков пальцев пользователя, нажмите *Finish*, чтобы закрыть диалоговое окно и сохранить изменения.

Как только отпечатки пальцев зарегистрированы, пользователь может выбрать метод аутентификации: отпечаток пальца или пароль для доступа к программам,

перечисленным в Раздел 7.1, «Программы, поддерживающие биометрическую аутентификацию» (стр. 115).

В настоящее время в YaST не реализована функция проверки или удаления существующих отпечатков пальцев. Если Вы все же хотите удалить их, просто удалите директорию `/home/login/.fprint`.

Для получения дополнительной технической информации см. <http://www.freedesktop.org/wiki/Software/fprint>.

Часть II. Локальная безопасность

Настройка параметров безопасности с помощью YaST

8

Модуль *Центр Безопасности* YaST предназначен для изменения настроек связанных с безопасностью системы. Этот модуль используется для настройки таких аспектов безопасности, как вход в систему, установка пароля, опции загрузки, создание пользователей и права доступа к файлам по умолчанию. Он запускается из Центра Управления YaST *Безопасность и Пользователи* > *Центр Безопасности*. После запуска *Центра Безопасности* активным является диалог *Обзор безопасности*, остальные диалоги настроек доступны в правой панели.

8.1 Обзор безопасности

Обзор безопасности отображает понятный список самых важных настроек безопасности Вашей системы. Статус безопасности каждого пункта списка очевиден. Зеленая галка соответствует безопасным значениям, в то время как красный крест говорит о том, что значение данного пункта списка не является безопасным. Нажав *Справка* Вы получите обзор настроек и информацию о том, как обезопасить систему. Для изменения настроек нажмите соответствующую ссылку в колонке Состояние. В зависимости от настройки доступны следующие значения:

Включено/Выключено

Нажав на этот элемент Вы можете включить или отключить соответствующую опцию.

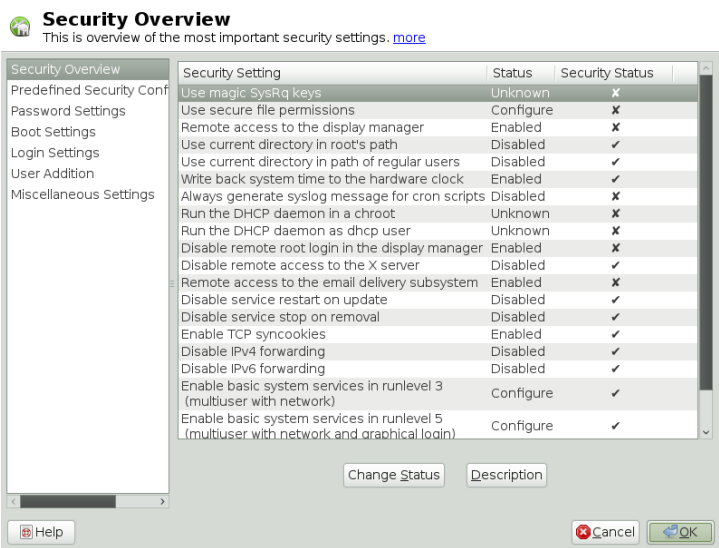
Настроить

Нажав на этот элемент Вы запустите соответствующий модуль YaST для изменения настроек. Вы вернетесь в диалог Обзор безопасности, как только модуль будет закрыт.

Неизвестно

Данный статус означает, что соответствующая служба не установлена. Он не является индикатором потенциальных проблем безопасности.

Рисунок 8.1 Центр безопасности YaST- Обзор безопасности



8.2 Предопределенные настройки безопасности

В есть три предопределенных набора настроек безопасности. Эти наборы применяются ко всем настройкам модуля *Центр Безопасности*. Каждый из них может быть изменен по своему усмотрению используя диалоги в правой панели. Список наборов следующий:

Домашняя рабочая станция

Этот набор настроек разработан для компьютера, который не является частью локальной сети и не подключен к Интернету. Соответствует состоянию с наименьшей безопасностью.

Сетевая рабочая станция

Данная конфигурация используется для компьютера, подключенного к локальной сети либо Интернету.

Сервер сети

Конфигурация разработана для компьютера предоставляющего сетевые сервисы: вебсервера, файлового сервера, DNS сервера и т.д. Соответствует наиболее безопасным значениям настроек.

Пользовательские настройки

Если пункт *Пользовательские настройки* выбран при открытии диалога *Предопределенные настройки безопасности* это говорит о том, что значения одного из предустановленных наборов настроек были изменены. Самостоятельное переключение на этот пункт с любого другого не изменяет настроек безопасности - Вы должны изменять их с помощью диалога *Обзор безопасности*.

8.3 Настройки пароля

Одной из самых важных проблем безопасности являются легко подбираемые пароли. Средства диалога *Настройки пароля* предназначены для проверки безопасности используемых паролей.

Проверять новые пароли

Если этот пункт активирован, пользователь получит предупреждение, при использовании в качестве пароля слова из словаря или имени собственного. Для ограничения минимальной длины пароля введите ее значение в поле *Минимальная приемлемая длина пароля* после активации *Проверять новые пароли*.

Число запоминаемых паролей

Когда проверка возраста пароля активирована, эта настройка используется для сохранения заданного числа предыдущих паролей пользователя, предотвращая их повторное использование.

Метод шифрования пароля

Алгоритм шифрования пароля. Значение по умолчанию (Blowfish) обычно не требует изменений.

Возраст пароля

Проверка возраста пароля активируется путем указания минимального и максимального значений (в днях). Установив минимальное значение больше 0 дней, вы можете запретить пользователю повторную немедленную смену пароля (и следующее за ней окончание времени жизни пароля). Для отключения проверки срока действия пароля используются значения 0 и 99999 соответственно.

Дней до предупреждения об истечении срока действия пароля

Пользователь может заблаговременно получать предупреждение об истечении срока действия пароля. В этом поле указывается количество дней до истечения срока действия пароля, по достижении которого пользователь будет получать предупреждение о необходимости сменить пароль.

8.4 Настройки загрузки

Этот диалог устанавливает, кому из пользователей разрешено выключать компьютер через менеджер авторизации. Здесь также можно указать, как будет интерпретироваться + + .

8.5 Настройки входа в систему

Диалог позволяет установить настройки безопасности, связанные со входом в систему:

Задержка после неправильной попытки входа

Определяет задержку (в секундах) после неудачной попытки входа в систему. Рекомендуется установить данное значение для затруднения проникновения в систему путем перебора паролей. Поскольку эта настройка влияет на пользователей, допустивших ошибку при наборе пароля, не заставляйте их ждать повторной авторизации слишком долго.

Записывать успешные попытки входа

При включении данной опции последняя удачная попытка авторизации будет сохраняться в файл `/var/log/lastlog` и отображаться при последующем входе в систему. Эти данные также используются командой `finger`.

ПРИМЕЧАНИЕ

Внимание! Эта опция не влияет на журнал `/var/log/wtmp`, который содержит дату и время всех авторизаций и перезагрузок системы. Содержимое `/var/log/wtmp` отображается с помощью команды `last`.

Разрешить удаленный графический вход

Если включено, графический менеджер авторизации (например `gdm` или `kdm`) будет доступен из сети. Включение представляет потенциальную угрозу безопасности системы.

8.6 Добавление пользователя

Минимальные и максимальные значения идентификаторов групп и пользователей. Изменение значений по умолчанию требуется крайне редко.

8.7 Различные настройки

Здесь представлены настройки безопасности, не соответствующие остальным категориям:

Разрешения файлов

предлагает три предустановленных уровня файловых привилегий для системных файлов. Эти наборы привилегий определяют, может ли обычный пользователь читать файлы журналов и запускать определенные программы. Уровень *Легкий* предназначен для системы с одним пользователем и позволяет, например, читать большинство системных файлов обычному пользователю. Полный список привилегий доступен в файле `/etc/permissions.easy`. Уровень *Безопасный* разработан для многопользовательских систем с доступом к сети. Подробное объяснение его настроек можно посмотреть в файле `/etc/permissions.secure`. Наиболее

жестким является уровень *Параноидальный*, использовать который следует осторожно. Информацию об этом уровне можно получить из файла `/etc/permissions.paranoid`.

Пользователь, запускающий updatedb

Программа `updatedb` сканирует систему и создает базу данных размещения файлов, используемую командой `locate`. При запуске `updatedb` от пользователя `nobody`, в базу данных добавляются только файлы доступные на чтение всем пользователям. При запуске ее от пользователя `root`, в базу данных попадут практически все файлы (за исключением тех, к которым `root` не имеет доступа на чтение).

Текущий каталог в пути root / Текущий каталог в пути обычных пользователей

При запуске программы без указания полного пути к ее исполняемому файлу, система производит поиск этой программы по пути установленному переменной `$PATH`. По умолчанию в список директорий для поиска команды не входит текущая директория. Это необходимо для того, чтобы, например, при запуске команды `ls` запускалась программа `/bin/ls`, а не троян из */текущий каталог/ls*. Для запуска программ из текущей директории к ее имени нужно добавить префикс `./`. При изменении этой опции текущая директория (`.`) будет добавлена в путь для поиска команды. Изменять значение по умолчанию для этих опций не рекомендуется.

Магические клавиши SysRq

Волшебная кнопка `SysRq` - это клавиатурное сочетание, позволяющее контролировать систему, даже если в ней произошел сбой. Для получения подробной информации обратитесь к файлу `/usr/src/linux/Documentation/sysrq.txt` (требуется установка пакета `kernel-source`).

Списки управления доступом в Linux

POSIX ACLs (access control lists - списки управления доступом) могут быть использованы как расширение традиционной концепции привилегий для объектов файловой системы. С помощью ACL привилегии могут быть установлены более гибко, чем при помощи традиционной концепции битов доступа.

Термин *POSIX ACL* предполагает, что это настоящий стандарт POSIX (*portable operating system interface* - переносимый интерфейс операционных систем). Соответствующие черновые стандарты POSIX 1003.1e и POSIX 1003.2c были исключены по ряду причин. Тем не менее, ACLs (в том виде, в котором они реализованы во многих системах, принадлежащих семейству UNIX) основаны на этих черновиках. Реализация ACL для файловой системы (описанная в этой главе) также следует этим двум стандартам.

9.1 Традиционные файловые привилегии

Вы можете найти подробную информацию о традиционных файловых привилегиях на Info странице пакета GNU Coreutils, секция *File permissions* (`info coreutils "File permissions"`). Существуют также дополнительные атрибуты `setuid`, `setgid`, `sticky bit`.

9.1.1 Бит `setuid`

В некоторых ситуациях привилегии доступа могут быть слишком строгими. Поэтому в Linux существуют дополнительные настройки, позволяющие для определенного действия временно сменить идентификатор текущего пользователя или группы. Например, программа `passwd` обычно требует привилегий суперпользователя для доступа к файлу `/etc/passwd`. Этот файл содержит важную информацию, такую как домашние каталоги пользователей, идентификаторы пользователей и групп. Таким образом обычный пользователь не сможет изменить `passwd`, потому что предоставлять всем пользователям прямой доступ к этому файлу слишком опасно. Возможным решением этой проблемы является механизм `setuid`. `setuid` (set user ID - установка идентификатора пользователя) это специальный файловый атрибут, который сообщает системе, что программу нужно выполнять используя определенный идентификатор пользователя. Рассмотрим команду `passwd`:

```
-rwsr-xr-x  1 root shadow 80036 2004-10-02 11:08 /usr/bin/passwd
```

Вы видите `s`, что указывает на установленный бит `setuid` для привилегий пользователя. Согласно ему, все пользователи, запускающие команду `passwd`, выполнят ее от пользователя `root`.

9.1.2 Бит `setgid`

Бит `setuid` применяется к пользователям. Однако существует аналогичное свойство для групп: бит `setgid`. Если этот бит установлен для программы, то программа будет использовать при работе идентификатор собственной группы, вне зависимости от того, какой пользователь ее запустил. В директории с установленным битом `setgid`, все вновь созданные поддиректории и файлы будут принадлежать к той же группе, к которой принадлежит директория. Рассмотрим следующую директорию:

```
drwxrws---  2 tux archive 48 Nov 19 17:12  backup
```

Вы видите `s`, что указывает на установленный бит `setgid` для привилегий группы. Владелец директории и члены группы `archive` будут иметь доступ к этой директории. Пользователи, не принадлежащие к этой группе, «отразятся» на нее. Эффективным идентификатором группы для для всех записанных в директорию файлов будет `archive`. Например, программе резервного копирования, работающей с идентификатором группы `archive`, не понадобятся привилегии суперпользователя для доступа к этой директории.

9.1.3 Sticky Bit

Существует также *sticky bit*. Его поведение отличается для файлов и директорий. Если он установлен для исполняемого файла, то соответствующая программа не будет выгружаться из оперативной памяти, чтобы избежать повторной загрузки с жесткого диска. Поскольку современные жесткие диски достаточно производительны, используется это редко. Если этот бит установлен для директории, пользователи не смогут удалять из нее чужие файлы. Типичными примерами являются директории `/tmp` и `/var/tmp`:

```
drwxrwxrwt 2 root root 1160 2002-11-19 17:15 /tmp
```

9.2 Преимущества ACL

Традиционно в Linux для каждого файлового объекта задано три набора привилегий. Они включают в себя права на чтение (`r`), запись (`w`) и исполнение (`x`) для каждого из трех типов пользователей: владельца файла, группы, и остальных пользователей. Помимо этого можно установить *set user id*, *set group id* и *sticky* бит. Эта скучная концепция вполне применима в большинстве ситуаций. Однако для более сложных сценариев и приложений системные администраторы должны были использовать обходные пути, чтобы преодолеть ограничения традиционной концепции привилегий.

ACL могут быть использованы как расширение традиционной концепции привилегий. Они позволяют предоставлять доступ индивидуальным пользователям или группам даже если те не являются владельцем или группой-владельцем. Списки управления доступом — это компонент ядра Linux, который в настоящее время поддерживается ReiserFS, Ext2, Ext3, JFS и XFS. С помощью ACL сложные сценарии могут быть реализованы без использования сложных моделей привилегий на уровне приложения.

Если Вы хотите заменить Windows сервер на Linux сервер, то преимущества ACL очевидны. Некоторые из подключенных рабочих машин могут работать под Windows даже после миграции. ОС Linux предоставляет службы печати и доступа к файлам клиентам Windows посредством Samba. Поскольку Samba поддерживает списки управления доступом, привилегии пользователя могут быть установлены используя графический интерфейс как на сервере Linux, так и в Windows (только для Windows NT и более поздних). С помощью части пакета Samba `winbindd`, возможно также назначить привилегии пользователям существующим только в домене Windows и не имеющим аккаунта на сервере Linux.

9.3 Определения

Класс пользователя

Стандартная концепция привилегий POSIX использует три *класса* пользователей для назначения привилегий файловой системы: владелец, группа-владелец и другие пользователи. Три бита доступа могут быть установлены для каждого класса пользователей предоставляя доступ на чтение (r), запись (w) и выполнение (x).

ACL

Привилегии пользователя и группы для всех видов объектов файловой системы (файлов и директорий) определяются посредством ACL.

ACL по умолчанию

Дефолтные ACL могут применяться только к директориям. Они определяют привилегии, которые объект файловой системы наследует от родительской директории при создании.

Запись ACL

Каждый список управления доступом состоит из набора записей ACL. Запись ACL содержит тип, описатель пользователя или группы, на который ссылается эта запись, и набор привилегий. Для некоторых типов записей описатель группы и пользователя не указывается.

9.4 Работа с ACL

Таблица 9.1, «Типы записей ACL» (стр. 131) отражает 6 существующих типов записей ACL, каждый из которых определяет привилегии пользователя или группы пользователей. Запись *владелец* определяет привилегии владельца файла или директории. Запись *группа-владелец* определяет привилегии группы, владеющей файлом. Суперпользователь может сменить владельца или группу-владельца при помощи команд `chown` и `chgrp`, в случае чего владелец и группа-владелец будут ссылаться на нового владельца или группу-владельца. Каждая запись *именованный пользователь* определяет привилегии пользователя, указанного в поле "квалификатор" этой записи. Каждая запись *именованная группа* определяет привилегии группы, указанной в поле "квалификатор" этой записи. Поле "квалификатор" заполнено только для записей типа *именованный пользователь* и *именованная группа*. Запись *другие* определяет привилегии всех остальных пользователей.

Запись *маска* ограничивает привилегии именованного пользователя, именованной группы и группы-владельца определяя, какие разрешения этих записей будут применяться, а какие - маскироваться. Если право доступа существует в одной из этих записей и в маске, то оно применяется. Права, содержащиеся только в маске или только в записи не применяются, и доступ в этом случае не будет разрешен. Права владельца и группы-владельца применяются всегда. Таблица 9.2, «Маскировка привилегий доступа» (стр. 131) демонстрирует этот механизм.

Существует два основных класса ACL: *минимальный ACL* — содержит записи типов «владелец», «группа-владелец» и «другие», соответствующие традиционным битам доступа для файлов и директорий. *Расширенный ACL* более продвинут. Он должен содержать маску и может включать в себя несколько записей для именованных пользователей и именованных групп.

Таблица 9.1 *Типы записей ACL*

Тип	Текстовая форма
владелец	user::rwx
именованный пользователь	user:name:rwx
группа-владелец	group::rwx
именованная группа	group:name:rwx
маска	mask::rwx
другие	other::rwx

Таблица 9.2 *Маскировка привилегий доступа*

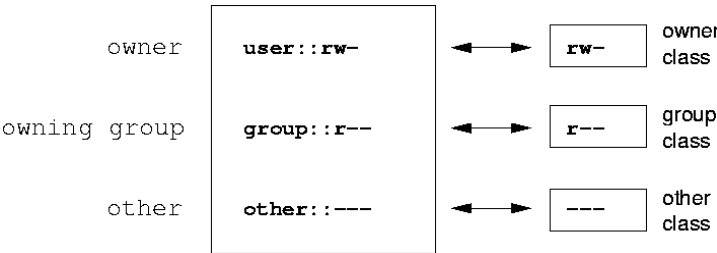
Тип записи	Текстовая форма	Привилегии
именованный пользователь	user:geeko:r-x	r-x
маска	mask::rw-	rw-

Тип записи	Текстовая форма	Привилегии
	эффективные привилегии:	r--

9.4.1 Записи ACL и биты доступа

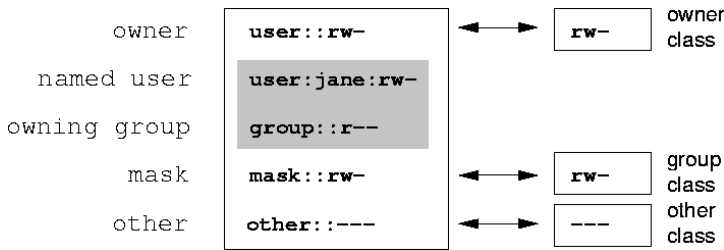
Рисунок 9.1, «Минимальный ACL: Сравнение записей ACL с битами доступа» (стр. 132) и Рисунок 9.2, «Расширенный ACL: Сравнение записей ACL с битами доступа» (стр. 133) соответствуют минимальному и расширенному ACL. На рисунках изображены три блока: левый показывает тип спецификации записей ACL, центральный отображает пример ACL, и правый блок соответствует битам доступа традиционной концепции привилегий (отображаемым, например, командой `ls -l`). В обоих случаях привилегии класса *владелец* отображаются на запись ACL «владелец». Привилегии класса *другие* отображаются на соответствующую запись ACL. Однако отображение прав доступа класса *группа* отличается для каждого из случаев.

Рисунок 9.1 Минимальный ACL: Сравнение записей ACL с битами доступа



В случае минимального ACL — без маски — права доступа группы отражаются на запись ACL группа-владелец, как показывает Рисунок 9.1, «Минимальный ACL: Сравнение записей ACL с битами доступа» (стр. 132). В случае расширенного ACL — с маской — права доступа группы отображаются на маску, Рисунок 9.2, «Расширенный ACL: Сравнение записей ACL с битами доступа» (стр. 133).

Рисунок 9.2 Расширенный ACL: Сравнение записей ACL с битами доступа



Это отображение используется для упрощения взаимодействия с приложениями, избавляя от необходимости поддержки ACL приложением. Привилегии доступа, назначенные посредством битов доступа, задают верхний предел для всех «тонких настроек», сделанных при помощи ACL. Изменение битов доступа отражается на ACL и наоборот.

9.4.2 Директория с ACL

Вы можете получить доступ к ACL с помощью команд `getfacl` и `setfacl`. Использование этих команд показано в следующем примере.

Перед созданием директории используйте команду `umask`, чтобы задать биты доступа маскируемые при создании файлового объекта. Команда `umask 027` устанавливает привилегии по умолчанию следующим образом: владелец получает полный доступ (0), группе запрещен доступ на запись (2), доступ остальным пользователям закрыт (7). В действительности `umask` маскирует соответствующие биты доступа или сбрасывает их. Подробности можно узнать на [man](#) странице `umask`.

`mkdir mydir` создает директорию `mydir` с привилегиями, установленными командой `umask`. С помощью `ls -dl mydir` можно определить правильность установки привилегий. Для данного примера вывод будет следующим:

```
drwxr-x--- ... tux project3 ... mydir
```

Используя `getfacl mydir`, проверьте исходное состояние ACL. Будет отображена следующая информация:

```
# file: mydir
# owner: tux
# group: project3
user::rwx
group::r-x
```

```
other::---
```

Первые три строки вывода отображают имя, владельца и группу-владельца директории. Следующие три строки содержат три записи ACL: «владелец», «группа-владелец» и «другие». Фактически мы имеем дело с минимальным ACL и команда `getfacl` не дает дополнительной информации по сравнению с командой `ls`.

Измените ACL, разрешив доступ на чтение, запись и выполнение пользователю `geeko` и группе `mascots` командой:

```
setfacl -m user:geeko:rw,group:mascots:rw mydir
```

Опция `-m` указывает `setfacl` изменить существующий ACL. Следующий за ней аргумент определяет записи ACL, которые будут изменены (можно указать несколько записей, разделяя их запятыми). Последняя часть определяет имя директории, к которой будут применены эти изменения. Используйте команду `getfacl`, чтобы узнать полученные ACL.

```
# file: mydir
# owner: tux
# group: project3
user::rw
user:geeko:rw
group::r-x
group:mascots:rw
mask::rw
other::---
```

В дополнение к записям, созданным для пользователя `geeko` и группы `mascots`, была добавлена запись с маской. Эта запись была создана автоматически, таким образом, что все заданные права доступа будут применены. `setfacl` автоматически адаптирует существующие записи с масками к изменяемым битам. Это поведение можно изменить используя опцию `-n`. Маска определяет максимальные эффективные привилегии для класса «группа», включая именованного пользователя, именованную группу и группу-владельца. Биты доступа класса «группа», отображаемые командой `ls`, `-dl mydir` теперь соответствуют записи маски.

```
drwxrwx---+ ... tux project3 ... mydir
```

Первый столбец вывода теперь содержит дополнительно знак `+`, указывая на существование *расширенных* ACL для этого элемента.

Согласно выводу команды `ls`, маска включает в себя доступ на запись. Традиционно эти биты означали бы, что группа-владелец (`project3`) также имеет пра-

во на запись в директорию `mydir`. Однако эффективные права доступа соответствуют сочетанию прав заданных для группы-владельца и маски — `r-x` в нашем примере (Таблица 9.2, «Маскировка привилегий доступа» (стр. 131)). Поэтому в отношении эффективных привилегий группы-владельца, даже после добавления дополнительных записей ACL, ничего не изменилось.

Измените значение маски с помощью команд `setfacl` или `chmod`. Например используйте `chmod g-w mydir`. `ls -dl mydir` покажет:

```
drwxr-x---+ ... tux project3 ... mydir
```

`getfacl mydir` даст следующий вывод:

```
# file: mydir
# owner: tux
# group: project3
user::rwx
user:geeko:rwx          # effective: r-x
group::r-x
group:mascots:rwx       # effective: r-x
mask::r-x
other:---
```

После сброса бита доступа на запись для группы командой `chmod`, вывода команды `ls` достаточно, чтобы увидеть, что биты маски изменились: доступ на запись снова есть только у владельца `mydir`. Вывод `getfacl` подтверждает это. Он включает в себя комментарий для всех записей, в которых биты доступа не соответствуют реальным привилегиям из-за применения маски. Оригинальные права могут быть восстановлены в любое время командой `chmod g+w mydir`.

9.4.3 Директория с ACL по умолчанию

Директории могут иметь ACL по умолчанию, которые являются специальной разновидностью ACL, определяющей права доступа, наследуемые объектами в этой директории при их создании. ACL по умолчанию влияет на поддиректории и файлы.

9.4.3.1 Действия ACL по умолчанию

Существует два пути передачи ACL по умолчанию файлам и поддиректориям:

- Поддиректория наследует ACL родительской директории как свои ACL по умолчанию и обычные ACL.

- Файл наследует ACL по умолчанию как свои ACL.

Все системные вызовы, создающие объекты файловой системы, используют параметр режим, который определяет режим доступа к созданному объекту. Если родительская директория не имеет ACL по умолчанию, заданные `umask` биты доступа вычитаются из битов доступа параметра режим, а результат устанавливается созданному объекту. Если же ACL по умолчанию задан, биты доступа нового объекта соответствуют совпадающим частям параметра режим и разрешениям ACL по умолчанию. В этом случае `umask` не учитывается.

9.4.3.2 Применение ACL по умолчанию

Следующие три примера показывают основные операции для директорий и ACL по умолчанию:

1. Добавить ACL по умолчанию существующей директории `mydir`:

```
setfacl -d -m group:mascots:r-x mydir
```

Опция `-d` команды `setfacl` указывает `setfacl` выполнить изменения (опция `-m`) для ACL по умолчанию.

Взглянем поближе на результат выполнения этой команды:

```
getfacl mydir

# file: mydir
# owner: tux
# group: project3
user::rwx
user:geeko:rwx
group::r-x
group:mascots:rwx
mask::rwx
other:---
default:user::rwx
default:group::r-x
default:group:mascots:r-x
default:mask::r-x
default:other:---
```

`getfacl` возвращает как ACL, так и ACL по умолчанию. ACL по умолчанию соответствуют строкам, начинающимся с `default`. Несмотря на то, что Вы передали команде `setfacl` только запись ACL по умолчанию для группы `mascots`, `setfacl` автоматически скопировала все остальные записи из ACL, чтобы создать валидный ACL по умолчанию. ACL по умолчанию не ока-

зывают моментального влияния на доступ к объекту. Они вступают в игру при создании новых объектов. Эти новые объекты наследуют привилегии только от ACL по умолчанию своей родительской директории.

2. В следующем примере используйте команду `mkdir`, чтобы создать в `mydir` поддиректорию, которая унаследует ACL по умолчанию.

```
mkdir mydir/mysubdir

getfacl mydir/mysubdir

# file: mydir/mysubdir
# owner: tux
# group: project3
user::rwx
group::r-x
group:mascots:r-x
mask::r-x
other:---
default:user::rwx
default:group::r-x
default:group:mascots:r-x
default:mask::r-x
default:other:---
```

Как и ожидалось, вновь созданная директория `mysubdir` получила права доступа из ACL по умолчанию родительской директории. ACL директории `mysubdir` является точным отражением ACL по умолчанию директории `mydir`. Поддиректория передаст эти права вложенным в нее объектам и т.д.

3. Используйте `touch` для создания файла в директории `mydir`, например, `touch mydir/myfile.ls -l mydir/myfile` покажет:

```
-rw-r-----+ ... tux project3 ... mydir/myfile
```

Вывод `getfacl mydir/myfile`:

```
# file: mydir/myfile
# owner: tux
# group: project3
user::rw-
group::r-x      # effective:r--
group:mascots:r-x # effective:r--
mask::r--
other:---
```

Если ACL по умолчанию и `umask` не накладывают никаких ограничений, `touch` при создании новых файлов использует режим со значением `0666`, создавая файлы с доступом на чтение и запись для всех (Раздел 9.4.3.1, «Дей-

ствия ACL по умолчанию» (стр. 135)). Также это означает, что все права доступа, которые не содержит значение режим будут удалены из соответствующих записей ACL. Несмотря на то, что записи для группы не были удалены из ACL, значение маски было модифицировано для маскировки прав не заданных значением режим.

Эта модель необходима для правильного взаимодействия приложений (например, компиляторов) с ACL. Вы можете создавать файлы с ограниченным доступом и затем помечать их как исполняемые. Механизм `mask` гарантирует, что только корректные пользователи и группы смогут запускать их как им благорассудится.

9.4.4 Алгоритм проверки ACL

Алгоритм проверки применяется перед тем, как любому процессу или приложению будет предоставлен доступ к защищенному ACL объекту файловой системы. Основное правило проверки заключается в том, что записи ACL проверяются в следующем порядке: владелец, именованный пользователь, группа-владелец или именованная группа, другие пользователи. Доступ предоставляется в соответствии с записью, которая более подходит процессу. Привилегии не накапливаются.

Все усложняется, если процесс принадлежит более чем к одной группе и потенциально соответствует нескольким групповым записям. В этом случае из множества выбирается произвольная запись с требуемыми привилегиями. Неважно, какая из записей приведет к результату «доступ разрешен». Аналогично, если ни одна из групповых записей не содержит требуемых привилегий, случайно выбранная запись приведет к конечному результату «доступ запрещен».

9.5 Поддержка ACL приложениями

ACL могут быть использованы для реализации очень сложных моделей привилегий, которые соответствуют требованиям современных приложений. Традиционная система привилегий и ACL могут эффективно сочетаться. Основные файловые команды (`cp`, `mv`, `ls`, и т.д.) поддерживают ACL, равно как Samba и Konqueror.

К сожалению множество редакторов и файловых менеджеров до сих пор не поддерживает ACL. Например, копирование файлов с помощью Emacs приводит к потере ACL. При изменении файлов в редакторе ACL иногда сохраняются, а иногда нет, в зависимости от того способа, которым редактор создает резервные копии. Если редактор записывает изменения в оригинальный файл, ACL сохраняются. Если измененное содержание сохраняется в новый файл, который впоследствии получает имя оригинального файла, и сам редактор не поддерживает ACL, они могут быть потеряны. За исключением архиватора star в настоящее время не существует программ резервного копирования, сохраняющих ACL.

9.6 Дополнительная информация

За подробной информацией об ACL обращайтесь в справку по командам `getfacl(1)`, `acl(5)` и `setfacl(1)`.

Шифрование файлов и разделов

10

Большинство пользователей имеет на своих компьютерах конфиденциальные данные, доступ к которым должен быть закрыт для третьих лиц. Чем больше Вы полагаетесь на переносные компьютеры и работу в различном окружении и разных сетях, тем более осторожным Вам следует быть со своими данными. При наличии сетевого или физического доступа к Вашей системе посторонних лиц рекомендуется шифровать файлы или целые дисковые разделы. Ноутбуки или носители информации, такие как внешние жесткие диски или USB-накопители, могут быть потеряны или украдены. Поэтому рекомендуется шифровать часть файловой системы, хранящую конфиденциальные данные.

Существует несколько способов защиты данных при помощи шифрования:

Шифрование раздела жесткого диска

Вы можете создать зашифрованный раздел во время инсталляции или на уже установленной системе при помощи YaST. Более подробная информация содержится в Раздел 10.1.1, «Создание зашифрованного раздела во время инсталляции» (стр. 143) и Раздел 10.1.2, «Создание зашифрованного раздела на работающей системе» (стр. 144). Этот же способ может быть использован для сменных накопителей, таких как внешние жесткие диски, как описано в Раздел 10.1.4, «Шифрование содержимого съемных носителей» (стр. 145).

Создание зашифрованного файла в качестве контейнера

Вы можете создать зашифрованный файл на своем жестком диске или сменном накопителе при помощи YaST. Этот файл может быть использован для того, чтобы *хранить* другие файлы и каталоги. За дополнительной информа-

цией обратитесь к Раздел 10.1.3, «Создание зашифрованного файла в качестве контейнера» (стр. 144).

Шифрование домашних каталогов

С помощью Вы также можете создавать зашифрованные домашние директории пользователей. Когда пользователь входит в систему, зашифрованная домашняя директория монтируется, и ее содержимое становится доступно пользователю. За подробной информацией обратитесь к Раздел 10.2, «Использование зашифрованных домашних директорий» (стр. 146)

Шифрование отдельных текстовых файлов ASCII

Если вся Ваша конфиденциальная информация хранится в нескольких текстовых файлах ASCII, Вы можете зашифровать их отдельно и защитить паролем используя Kpg или редактор vi. За дальнейшей информацией обратитесь к Раздел 10.3, «Использование vi для шифрования отдельных текстовых ASCII файлов» (стр. 147) .

ПРЕДУПРЕЖДЕНИЕ: Зашифрованный носитель предлагает ограниченную защиту

Описанные в этой главе методы предлагают только ограниченную защиту. Вы не можете защитить запущенную систему от взлома. После того, как зашифрованный носитель был благополучно смонтирован, любой пользователь с соответствующими привилегиями будет иметь к нему доступ. Однако зашифрованный носитель будет полезен в случае потери или кражи Вашего компьютера или для предотвращения чтения Ваших конфиденциальных данных третьими лицами.

10.1 Создание зашифрованной файловой системы при помощи YaST

YaST можно использовать для шифрования разделов или частей Вашей файловой системы как во время инсталляции, так и на уже установленной системе. Однако шифрование раздела уже установленной системы более трудоемко, поскольку оно требует внесения изменений в таблицу разделов. В этом случае

более целесообразным может быть создание зашифрованного файла фиксированного размера, который будет использован для *хранения* других файлов или частей Вашей файловой системы. Чтобы зашифровать целый раздел, выделите в таблице разделов раздел для шифрования. Разметка, предлагаемая YaST по умолчанию, не содержит зашифрованных разделов. Добавьте их вручную в диалог разметки диска.

10.1.1 Создание зашифрованного раздела во время инсталляции

ПРЕДУПРЕЖДЕНИЕ: Ввод пароля

Убедитесь, что Вы хорошо запомнили пароль для зашифрованного раздела. Без этого пароля Вы не сможете ни получить доступ, ни восстановить зашифрованные данные.

Диалог YaST Разметка предоставляет возможность создания зашифрованного раздела. Чтобы создать новый зашифрованный раздел:

- 1 Запустите модуль YaST Разметка *Компьютер > Система > Разметка*.
- 2 Выберите жесткий диск, нажмите *Добавить* и выберите первичный или расширенный раздел.
- 3 Выберите размер раздела или область для использования на диске.
- 4 Выберите тип файловой системы и точку монтирования раздела.
- 5 Отметьте *Шифровать устройство*.

ПРИМЕЧАНИЕ: Требуется дополнительное программное обеспечение

После выбора *Шифровать устройство*, возможно появление всплывающего окна, запрашивающего установку дополнительного программного обеспечения. Подтвердите установку всех требуемых пакетов если хотите, чтобы зашифрованный раздел был работоспособен.

- 6 Нажмите *Далее* и введите пароль, который будет использоваться для шифрования раздела. Он не будет отображаться на экране, поэтому во избежание ошибок его нужно будет ввести дважды.
- 7 Завершите процесс нажав *Завершить*, после чего будет создан новый зашифрованный раздел.

Когда Вам понадобится смонтировать зашифрованный раздел, откройте файловый менеджер и во вкладке, отображающей общие места Вашей файловой системы, выберите запись с разделом. У Вас будет запрошен пароль, после чего раздел будет смонтирован.

При инсталляции системы на компьютер с уже существующими разделами Вы можете зашифровать существующий раздел. В этом случае следуйте описанию в Раздел 10.1.2, «Создание зашифрованного раздела на работающей системе» (стр. 144) и имейте в виду, что все существующие данные будут уничтожены во время этой операции.

10.1.2 Создание зашифрованного раздела на работающей системе

ПРЕДУПРЕЖДЕНИЕ: Активация шифрования на работающей системе

Также возможно создать зашифрованный раздел на работающей системе. Однако шифрование существующего раздела уничтожит все данные на нем, а также потребует изменения размеров и структуры существующих разделов.

На запущенной системе выберите *Система > Разметка* в Центре управления YaST. Нажмите *Да* чтобы продолжить. В *Экспертной разметке* выберите раздел для шифрования и нажмите *Редактировать*. Остальная процедура сходна с описанной в Раздел 10.1.1, «Создание зашифрованного раздела во время инсталляции» (стр. 143).

10.1.3 Создание зашифрованного файла в качестве контейнера

Вместо использования раздела можно создать зашифрованный файл, который будет содержать в себе другие файлы и каталоги с конфиденциальными данными. Такие контейнерные файлы создаются из диалога Экспертная разметка YaST. Выберите *Шифрованные файлы > Добавить шифрованный файл* и введите полный путь к файлу и его размер. Если YaST должен создать контейнерный файл, активируйте опцию *Создать петлевой файл*. Подтвердите или измените предлагаемые настройки форматирования и тип файловой системы. Укажите точку монтирования и убедитесь, что опция *Зашифровать устройство* выбрана.

Нажмите *Далее*, введите пароль для дешифрования файла и примените изменения нажав *Завершить*.

Преимущество зашифрованных файлов-контейнеров над разделами заключается в том, что они могут быть добавлены без изменения разметки жесткого диска. Они монтируются при помощи петлевого устройства и ведут себя как обычные разделы.

10.1.4 Шифрование содержимого съемных носителей

YaST относится к съемному носителю (такому, как жесткий диск или USB flash) так же, как обычному жесткому диску. Файлы-контейнеры или разделы на таких устройствах могут быть зашифрованы по инструкции выше. Однако не разрешайте им монтирование во время загрузки, поскольку съемные носители обычно подключаются к уже запущенной системе

Если съемное устройство было зашифровано при помощи YaST, то среды KDE и GNOME автоматически распознают зашифрованный раздел и запросят пароль при обнаружении устройства. Если Вы подключите съемное устройство отформатированное в FAT при запущенном KDE или GNOME, пользователь после ввода пароля автоматически станет владельцем устройства и сможет читать и записывать файлы. Для других файловых систем необходимо явно указать владельца отличного от `root`, чтобы разрешить этим пользователям чтение или запись файлов на устройство.

10.2 Использование зашифрованных домашних директорий

Для защиты данных в домашних директориях от кражи и последующего несанкционированного доступа, используйте модуль YaST Управление пользователями, чтобы разрешить шифрование домашних директорий. Вы можете создавать зашифрованные домашние директории для новых и существующих пользователей. Для шифрования или дешифрования домашних директорий уже существующих пользователей Вы должны знать их пароли. Раздел “Managing Encrypted Home Directories” (Глава 10, *Managing Users with YaST*, ↑Вступление) подробно описывает эту процедуру.

Зашифрованные домашние разделы создаются внутри файла-контейнера как описано в Раздел 10.1.3, «Создание зашифрованного файла в качестве контейнера» (стр. 144). В директории `/home` создается два файла для каждой зашифрованной домашней директории:

`LOGIN.img`

Файл, содержащий образ директории

`LOGIN.key`

Защищенный паролем пользователя ключ к образу .

При входе в систему домашняя директория дешифруется автоматически. Для этого используется модуль `pam` называемый `pam_mount`. Если Вам надо добавить дополнительный метод входа в систему, который обеспечивает зашифрованные домашние директории, добавьте этот модуль в соответствующий конфигурационный файл `/etc/pam.d/`. За подробностями обратитесь к Глава 2, *Авторизация с помощью PAM* (стр. 19) и `man` странице `pam_mount`.

ПРЕДУПРЕЖДЕНИЕ: Ограничения безопасности

Шифрование домашней директории пользователя не обеспечивает высокого уровня защиты от других пользователей. Его можно добиться только не разделяя систему с другими пользователями физически.

Для усиления безопасности Вы можете также зашифровать `swap` раздел, а также директории `/tmp` и `/var/tmp`, поскольку они могут содер-

жать временные образы или критические данные. Зашифровать `swap`, `/tmp` и `/var/tmp` можно при помощи модуля Разметка YaST как описано в Раздел 10.1.1, «Создание зашифрованного раздела во время инсталляции» (стр. 143) или Раздел 10.1.3, «Создание зашифрованного файла в качестве контейнера» (стр. 144).

10.3 Использование `vi` для шифрования отдельных текстовых ASCII файлов

Недостаток использования зашифрованных разделов очевиден: когда раздел смонтирован, по крайней мере `root` будет также иметь доступ к данным в этом разделе. Для предотвращения этого можно использовать `vi` в режиме шифрования.

Чтобы отредактировать новый файл используйте команду `vi -x filename.vi` предложит Вам установить пароль, после чего он зашифрует содержимое файла. Откуда бы Вы не запросили доступ к этому файлу, `vi` будет требовать пароль для доступа.

Для еще большей безопасности Вы можете разместить зашифрованный файл на зашифрованном разделе. Это рекомендуется, поскольку используемое в `vi` шифрование не слишком устойчиво.

Обнаружение вторжений при помощи AIDE

11

Защита системы - обязательная задача для любого ответственного системного администратора. Поскольку невозможно гарантировать, что система не подвергнется взлому, очень важно производить дополнительные проверки регулярно (например по крону), чтобы убедиться в том, что система все еще контролируется Вами. Для этого удобно использовать *AIDE Advanced Intrusion Detection Environment - усовершенствованную среду обнаружения вторжений*.

11.1 Для чего нужна AIDE?

Простая проверка, которая часто помогает обнаружить нежелательные изменения, может быть произведена с помощью RPM. В пакетном менеджере есть встроенная функция проверки изменений в файлах системы. Для проверки всех файлов, выполните команду `rpm -Va`. Однако эта команда также отобразит изменения в файлах конфигурации и Вам придется отфильтровать вывод, чтобы определить только важные изменения.

Еще одна проблема с RPM заключается в том, что умный взломщик может подменить `rpm`, чтобы замаскировать все изменения. Это может быть сделано при помощи руткита, который позволит взломщику скрыть вторжение и получить привилегии суперпользователя. Поэтому Вы должны реализовать еще одну проверку, которую следует производить независимо от проверяемой системы.

11.2 Настройка базы данных AIDE

ВАЖНО: Инициализируйте базу данных AIDE после установки

Перед установкой системы проверьте контрольную сумму носителя (Раздел “Checking Media” (Приложение А, *Помощь и решение проблем*, ↑Вступление)), чтобы убедиться в его подлинности. После установки системы инициализируйте базу данных AIDE. Чтобы удостовериться, что все прошло успешно во время и после инсталляции, произведите инсталляцию прямо из консоли, на компьютер не подключенный к сети. Не оставляйте компьютер без присмотра и не подключайте его к сети до завершения процесса создания базы данных AIDE.

AIDE по умолчанию не установлена на . Для установки используйте *Компьютер* > *Установка программ*, или введите `zypper install aide` в командной строке от `root`.

Чтобы указать AIDE какие атрибуты каких файлов должны проверяться, используйте конфигурационный файл `/etc/aide.conf`. Работу с программой следует начать с этого файла. Первая его секция управляет общими параметрами, такими как размещение файла базы данных AIDE. К локальным настройкам относятся секции `Custom Rules` и `Directories and Files`. Типичное правило выглядит следующим образом:

```
Binlib      = p+i+n+u+g+s+b+m+c+md5+sha1
```

После определения переменной `Binlib`, соответствующие опции проверки используются в секции файлов. Следующие настройки являются важными:

Таблица 11.1 Важные настройки AIDE

Опция	Описание
p	Проверяет права доступа выбранных файлов и директорий.
i	Проверяет номер файлового дескриптора (inode). Каждому имени файла соответствует уникальный номер дескриптора, который не должен изменяться.

Опция	Описание
n	Проверяет число ссылок на соответствующий файл.
u	Проверяет, изменился ли владелец файла.
g	Проверяет, изменилась ли группа файла.
s	Проверяет, изменился ли размер файла.
b	Проверяет, изменилось ли число используемых файлом блоков.
m	Проверяет, изменилось ли время модификации файла.
c	Проверяет, изменилось ли время последнего доступа к файлу.
md5	Проверяет, изменилась ли у файла контрольная сумма md5.
sha1	Проверяет, изменилась ли у файла контрольная сумма sha1 (160 бит).

Вот конфигурация, проверяющая все файлы в директории `/sbin` за исключением директории `/sbin/conf.d/` с опциями, заданными в `Binlib`:

```
/sbin Binlib
!/sbin/conf.d
```

Для создания базы данных AIDE выполните следующие действия:

- 1 Откройте `/etc/aide.conf`.
- 2 Укажите файлы, которые необходимо проверять и настройки проверки. За подробным списком настроек обратитесь к `/usr/share/doc/packages/`

`aide/manual.html`. Определение файлов требует некоторых знаний о регулярных выражениях. Сохраните Ваши изменения.

- 3 Чтобы проверить правильность конфигурационного файла, выполните:

```
aide --config-check
```

Любой вывод, произведенный этой командой, является подсказкой об ошибках в конфигурации. Например вы можете увидеть следующее:

```
aide --config-check
35:syntax error:!
35:Error while reading configuration:!
Configuration error
```

Ошибку следует искать в строке 36 файла `/etc/aide.conf`. Обратите внимание, что сообщение об ошибке содержит последнюю успешно считанную строку конфигурационного файла.

- 4 Инициализируйте базу данных AIDE. Выполните команду:

```
aide -i
```

- 5 Скопируйте созданную базу в безопасное место, например CD-R или DVD-R, удаленный сервер или USB диск для последующего использования.

ВАЖНО:

Этот шаг важен, поскольку он помогает избежать подделки базы данных. Чтобы предотвратить изменение базы данных, рекомендуется использовать носитель, который может быть записан только один раз. *Никогда* не оставляйте базу на компьютере, за которым Вы хотите наблюдать.

11.3 Локальные проверки AIDE

Чтобы проверить файловую систему, выполните следующие шаги:

- 1 Переименуйте базу данных:

```
mv /var/lib/aide/aide.db.new /var/lib/aide/aide.db
```

- 2 После любых изменений конфигурации Вы всегда должны переинициализировать базу данных AIDE и перенести вновь созданную базу данных. Создание

резервной копии базы также хорошая идея. Подробнее об этом рассказывает Раздел 11.2, «Настройка базы данных AIDE» (стр. 150).

3 Выполните проверку при помощи следующей команды:

```
aide --check
```

Если вывод пуст - все хорошо. Если AIDE найдет изменения, будет отображен сводный отчет изменений, например:

```
aide --check
AIDE found differences between database and filesystem!!

Summary:
  Total number of files:      1992
  Added files:                0
  Removed files:              0
  Changed files:              1
```

Чтобы подробнее узнать об изменениях, смените уровень детализации проверки с помощью параметра `-V`. Для предыдущего примера, результат может выглядеть следующим образом:

```
aide --check -V
AIDE found differences between database and filesystem!!
Start timestamp: 2009-02-18 15:14:10

Summary:
  Total number of files:      1992
  Added files:                0
  Removed files:              0
  Changed files:              1

-----
Changed files:
-----

changed: /etc/passwd

-----
Detailed information about changes:
-----

File: /etc/passwd
  Mtime   : 2009-02-18 15:11:02           , 2009-02-18 15:11:47
  Ctime   : 2009-02-18 15:11:02           , 2009-02-18 15:11:47
```

В этом примере мы обработали файл `/etc/passwd` командой `touch`.

11.4 Проверка из независимой системы

Для ответственных администраторов (которыми мы все являемся) рекомендуется запускать бинарный файл AIDE из доверенного источника. Это исключает возможность модификации файла AIDE взломщиком для скрытия следов проникновения в систему.

Для этого AIDE должен быть запущен в системе, независимой от установленной системы. При помощи относительно легко добавить в Систему аварийного восстановления произвольные программы для достижения требуемого функционала.

Перед началом использования Системы аварийного восстановления Вы должны предоставить системе два пакета. Они включаются в нее точно также, как диск с обновлениями драйверов. За подробным описанием возможностей linuxgc используемых для этих целей обратитесь к <http://en.opensuse.org/SDB:Linuxrc>. Далее будет описан один из возможных способов.

Процедура 11.1 *Запуск Системы аварийного восстановления с AIDE*

- 1 Укажите FTP сервер в качестве второго компьютера.
- 2 Скопируйте пакеты `aide` и `mhash` в нужную директорию FTP сервера, в нашем случае `/srv/ftp/`. Замените `ARCH` и `VERSION` на Ваши значения:

```
cp DVD1/suse/ARCH/aideVERSION.ARCH.rpm /srv/ftp
cp DVD1/suse/ARCH/mhashVERSION.ARCH.rpm /srv/ftp
```

- 3 Создайте файл `/srv/ftp/info.txt`, содержащий параметры загрузки Системы аварийного восстановления:

```
dud:ftp://ftp.example.com/aideVERSION.ARCH.rpm
dud:ftp://ftp.example.com/mhashVERSION.ARCH.rpm
```

Замените FTP домен, `VERSION` и `ARCH` значениями, используемыми на Вашей системе.

- 4 Перезапустите сервер, который необходимо проверить при помощи AIDE, и загрузите его с DVD, содержащего Систему аварийного восстановления. Добавьте следующую строку в параметры загрузки:


```
info=ftp://ftp.example.com/info.txt
```

Этот параметр предписывает `linuxrc` считать всю информацию из файла `info.txt`.

После загрузки Системы аварийного восстановления программа AIDE готова к использованию.

11.5 Для дальнейшей информации

Информация об AIDE доступна:

- Домашняя страница AIDE <http://aide.sourceforge.net>.
- В виде документации к шаблону файла конфигурации `/etc/aide.conf`.
- В нескольких файлах, перечисленных ниже `/usr/share/doc/packages/aide`, после установки пакета `aide`.
- В рассылке новостей AIDE <https://mailman.cs.tut.fi/mailman/listinfo/aide>.

Часть III. Сетевая безопасность

SSH: Безопасная работа в сети

12

Часто бывает необходимо получить доступ к компьютеру удаленно. Если пользователь отправляет логин и пароль в виде обычного текста, они могут быть перехвачены и использованы злоумышленником для того, чтобы получить доступ к удаленной системе от имени этого пользователя. Это откроет нападающему доступ ко всем файлам пользователя и может быть использовано для попытки получения `root` привилегий или попытки проникновения на другую систему. В прошлом для удаленных соединений использовались `telnet`, `rsh` и `rlogin`, не шифровавшие передаваемый трафик и не защищенные от прослушивания. Есть и другие незащищенные каналы связи, создаваемые, например, при использовании `FTP` или других программ для копирования через сеть, например `rscp`.

Комплект программ SSH обеспечивает необходимую защиту, шифруя передаваемый трафик, включая логин и пароль. При использовании SSH данные все же могут быть перехвачены, но без ключа, использующегося для шифрования, их невозможно будет расшифровать. Таким образом SSH обеспечивает безопасное соединение в небезопасной сети, такой как интернет. Комплект программ SSH доступен в пакете `OpenSSH`.

В пакет `OpenSSH` установлен по умолчанию и включает в себя программы `ssh`, `scp` и `sftp`. В конфигурации по умолчанию удаленный доступ к возможен только с помощью служебных программ из комплекта `OpenSSH`, и только если запущен `sshd` и открыты соответствующие порты в брандмауэре.

12.1 ssh—Secure Shell

Используя `ssh` можно подключаться к удаленным системам и работать с ними в интерактивном режиме. Для того, чтобы подключиться к `sun` как пользователь `tux` используйте одну из следующих команд:

```
ssh tux@sun
ssh -l tux sun
```

Если имя пользователя на локальном и удаленном компьютере совпадает, Вы можете его опустить: `ssh sun`. Удаленный компьютер запросит пароль удаленного пользователя. После успешной аутентификации Вы можете работать на удаленном компьютере в режиме командной строки и использовать интерактивные приложения, например YaST, в текстовом режиме.

Более того, `ssh` предоставляет возможность удаленного запуска неинтерактивных команд на удаленной системе `ssh HOST COMMAND`. `COMMAND` должна быть при необходимости взята в кавычки. Как и в обычной командной строке, можно объединить несколько команд.

```
ssh root@sun "dmesg | tail -n 25"
ssh root@sun "cat /etc/issue && uptime"
```

12.1.1 Запуск приложений X на удаленном компьютере

SSH также упрощает использование удаленных приложений X. Если Вы запустите `ssh` с ключем `-X`, переменная `DISPLAY` будет автоматически установлена на удаленном компьютере и весь вывод X-сервера будет экспортироваться по SSH соединению. Следует отметить, что запущенные удаленно приложения X не могут быть перехвачены посторонними лицами.

12.1.2 Проброс агента

Используя опцию `-A`, механизм аутентификации `ssh-agent` переносится на следующий компьютер. Таким образом, Вы можете работать с различных компьютеров без ввода пароля, но только если Вы распространили свой публичный ключ на целевые машины и сохранили его там надлежащим образом.

По умолчанию этот режим выключен, но может быть активирован в любое время в файле конфигурации `/etc/ssh/sshd_config` установкой `AllowAgentForwarding yes`.

12.2 scp—безопасное копирование

scp копирует файлы на удаленный компьютер или с него. Если имя пользователя на jupiter отличается от имени пользователя на sun, укажите последнее используя формат username@host. Если файл требуется скопировать в директорию, отличную от домашней директории удаленного пользователя, укажите ее как sun:DIRECTORY. Следующие примеры показывают, как скопировать файл с локальной машины на удаленную и наоборот.

```
# local -> remote
scp ~/MyLetter.tex tux@sun:/tmp
# remote -> local
scp tux@sun:/tmp/MyLetter.tex ~
```

ПОДСКАЗКА: Опция -l

Команда ssh может быть использована с опцией -l, чтобы указать удаленного пользователя (как альтернатива формату username@host). С командой scp опция -l используется для ограничения потребляемого scp канала.

После того, как правильный пароль введен, scp начинает передачу данных и показывает растущий ряд звездочек, имитируя процентное соотношение выполнения процесса. Кроме того, программа показывает предполагаемое время до его завершения. Если указан ключ -q, программа не будет ничего выводить на терминал.

scp также поддерживает рекурсивное копирование директорий. Команда

```
scp -r src/ sun:backup/
```

скопирует содержимое всей директории src, включая все поддиректории, в директорию backup системы sun. Если поддиректории не существуют, они будут созданы автоматически.

При использовании ключа -p, scp сохранит дату последнего изменения источника. Ключ -C позволит сжать передаваемые данные. Это минимизирует объем передаваемых данных, но нагрузит процессоры на обеих машинах.

12.3 sftp—безопасная передача файлов

Если Вы хотите скопировать несколько файлов, утилита `sftp` будет удобной альтернативой `scp`. Эта утилита открывает консоль с набором команд похожим на команды стандартной `ftp` консоли. Введите `help` в подсказке `sftp` для получения списка доступных команд. Дополнительную информацию можно получить на странице `man sftp (1)`.

```
sftp sun
Enter passphrase for key '/home/fs/.ssh/id_rsa':
Connected to sun.
sftp> help
Available commands:
bye                               Quit sftp
cd path                           Change remote directory to 'path'
[...]
```

12.4 SSH демон (sshd)

Для работы с программами `ssh` и `scp`, в фоновом режиме должен быть запущен SSH-демон, который будет использовать порт 22 TCP/IP. При запуске в первый раз, демон генерирует три пары ключей. Каждая пара состоит из секретного и открытого ключа. Таким образом, эта процедура называется основанной на открытом ключе. Чтобы гарантировать безопасность соединения через SSH, доступ к файлу с секретным ключом должен быть только у системного администратора. По умолчанию, этот файл имеет именно такие права. Секретный ключ используется только локальным SSH-демоном и не должен быть предоставлен кому-то ещё. Открытый ключ (узнаваемый по расширению файла `.pub`) отправляется клиенту, запросившему соединение. Файл, в котором находится открытый ключ, может прочитать любой пользователь системы.

Связь инициирует SSH-клиент. SSH-демон ожидает запроса от SSH-клиента для создания соединения. Первым шагом они обмениваются информацией идентификации, проверяя протокол и версию SSH, а так же уточняя номер порта, т.к. дочерние процессы, порожденные SSH-демоном, могут одновременно обслуживать несколько SSH-сессий.

OpenSSH для связи между SSH-сервером и SSH-клиентом поддерживает первую и вторую версию протокола SSH. По умолчанию используется вторая версия. Если Вы хотите использовать первую версию, используйте ключ `-1`.

При использовании версии 1, SSH-сервер посылает свой открытый ключ и так называемый "ключ сервера", которые генерируются раз в час. На основании этих ключей SSH-клиент генерирует ключ сессии, который в последствии он посылает SSH-серверу. SSH-клиент также говорит серверу какой метод шифрования использовать. Вторая версия протокола SSH не требуют ключа сервера. Обе стороны используют алгоритм Диффи-Хеллмана (Diffie-Hellman) для обмена ключами.

Секретный и серверный ключи необходимы для расшифровки ключа сессии и не могут быть получены, при использовании лишь открытых ключей. Только запрашиваемый SSH-демон может расшифровать ключ сессии, используя свои секретные ключи. Эта начальная фаза установления соединения может быть исследована более детально (включая отладочную информацию) при помощи ключа `-v` SSH-клиента.

Рекомендуется сохранять секретные и открытые ключи, которые находятся в `/etc/ssh/`, на каком-нибудь безопасном внешнем носителе. В этом случае можно отследить подделку ключей или использовать старые ключи повторно, к примеру, после переустановки системы.

ПОДСКАЗКА: Существующие SSH ключи

Если Вы установите на машину с уже установленной Linux системой, SSH ключ этой системы будет автоматически импортирован во время инсталляции с сохранением последнего времени доступа к нему.

При первом соединении с удаленным компьютером, клиент сохранит все открытые ключи в `~/.ssh/known_hosts`. Это предотвратит любую атаку типа человек посередине—попытки других SSH серверов использовать поддельные имена и IP адреса. Такие атаки обнаруживаются либо по ключу, отсутствующему в `~/.ssh/known_hosts`, либо по неспособности сервера расшифровать ключ сессии из-за отсутствия соответствующей защищенной части ключа.

В случае, если открытый ключ сервера изменился (что необходимо выяснить перед попыткой подключения к такому серверу), несоответствующие ключи могут быть удалены командой `ssh-keygen -r HOSTNAME`

12.5 Механизм аутентификации SSH

В своей простейшей форме, аутентификация производится вводом пароля пользователя, как и при локальном входе в систему. Однако, запоминать пароли нескольких пользователей удаленных компьютеров достаточно непрактично. К тому же, эти пароли могут быть изменены. С другой стороны— при предоставлении доступа в качестве `root` администратору необходимо иметь возможность быстро отменить эти привилегии без изменения пароля `root`

Для входа в систему, не используя пароль удаленного пользователя, SSH использует другую пару ключей, которая должна быть сгенерирована пользователем. Она состоит из открытого (`id_rsa.pub` или `id_dsa.pub`) и защищенного ключа (`id_rsa` или `id_dsa`).

Чтобы войти в систему не указывая пароль удаленного пользователя, открытый ключ «пользователя SSH» должен существовать в `~/.ssh/authorized_keys`. Этот подход также предполагает, что удаленный пользователь получает полный доступ: добавление ключа требует знания пароля удаленного пользователя и удаление ключа лишает прав удаленного подключения к системе.

Для максимальной безопасности этот ключ должен быть защищен паролем, который вводится при каждом использовании команд `ssh`, `scp` и `sftp`. В отличие от простой аутентификации, этот пароль не зависит от удаленного пользователя и поэтому не изменяется.

В качестве альтернативы описанной выше аутентификации основанной на ключах, SSH также предлагает аутентификацию основанную на имени хоста. С ее помощью пользователь доверенного компьютера может подключиться к другому компьютеру используя своё имя пользователя. сконфигурирована для использования аутентификации основанной на ключах и настройка на ней аутентификации основанной на имени хоста находится за пределами данного руководства.

ПРИМЕЧАНИЕ: Права доступа к файлам при использовании host-аутентификации

При использовании host-аутентификации, файл `/usr/lib/ssh/ssh-keysign` (32-битные системы) или `/usr/lib64/ssh/ssh-keysign`

(64-битные системы) должен иметь SETUID бит, который в не установлен по умолчанию. Вы должны установить его сами вручную. Используйте для этого файл `/etc/permissions.local`, чтобы быть уверенным, что SETUID бит сохранится и после обновления OpenSSH.

12.5.1 Генерация SSH ключа

- 1 Для создания ключа с настройками по умолчанию (RSA, 2048 bits), введите команду `ssh-keygen`.
- 2 Согласитесь с дефолтным размещением ключа (`~/.ssh/id_rsa`), нажав (крайне рекомендуется) или введите альтернативное размещение.
- 3 Введите пароль из 10 - 30 символов. Для него справедливы правила создания безопасных паролей. Крайне рекомендуется не оставлять его пустым.

Вы должны быть абсолютно уверены, что Ваш защищенный ключ недоступен никому кроме Вас (всегда устанавливайте его права доступа в `0600`). Защищенный ключ никогда не должен попадать в чужие руки.

Для смены пароля к существующему ключу используйте команду `ssh-keygen -p`.

12.5.2 Копирование ключа SSH

Для копирования открытого SSH ключа в файл `~/.ssh/authorized_keys` пользователя удаленного компьютера, используйте команду `ssh-copy-id`. Для того, чтобы скопировать Ваш личный ключ из `~/.ssh/id_rsa.pub` Вы можете использовать короткую форму. Для копирования ключей DSA или ключей других пользователей, Вы должны указать путь:

```
# ~/.ssh/id_rsa.pub
ssh-copy-id -i tux@sun

# ~/.ssh/id_dsa.pub
ssh-copy-id -i ~/.ssh/id_dsa.pub tux@sun

# ~notme/.ssh/id_rsa.pub
ssh-copy-id -i ~notme/.ssh/id_rsa.pub tux@sun
```

Для успешного копирования ключа, Вам потребуется ввести пароль удаленного пользователя. Для удаления существующего ключа отредактируйте вручную файл `~/.ssh/authorized_keys`.

12.5.3 Использование ssh-agent

При необходимости произвести большое количество операций довольно неудобно водить пароль SSH для каждой из них. Поэтому пакет SSH содержит еще один инструмент, `ssh-agent`, который сохраняет секретные ключи на время X-сессии или консольной сессии. Все остальные окна или программы запускаются как клиенты `ssh-agent`. После запуска агента, устанавливается набор переменных окружения, который будет использован командами `ssh`, `scp` и `sftp`, чтобы произвести автоматический вход посредством агента. За подробностями обратитесь к `man 1 ssh-agent`.

После запуска `ssh-agent` Вам понадобится добавить ключи используя команду `ssh-add`. Она запросит пароль. Затем Вы сможете использовать в текущей сессии команды `ssh` не вводя пароль для каждой из них.

12.5.3.1 Использование ssh-agent в X-сессии

В `ssh-agent` запускается автоматически оконными менеджерами GNOME и KDE. Для того, чтобы добавить свои ключи посредством `ssh-add` при старте X-сессии, необходимо:

- 1 Войти в систему как пользователь и проверить, существует ли файл `~/.xinitrc`.
- 2 Если он не существует, использовать шаблон или скопировать его из `/etc/skel`:

```
if [ -f ~/.xinitrc.template ]; then mv ~/.xinitrc.template ~/.xinitrc; \
else cp /etc/skel/.xinitrc.template ~/.xinitrc; fi
```
- 3 Если Вы скопировали шаблон, найдите следующие строки и раскомментируйте их. Если `~/.xinitrc` уже существует, добавьте следующие строки (без знаков комментария).

```
# if test -S "$SSH_AUTH_SOCK" -a -x "$SSH_ASKPASS"; then
#     ssh-add < /dev/null
# fi
```
- 4 при старте новой X-сессии, у Вас будет запрошен SSH пароль.

12.5.3.2 Использование ssh-agent в консольном сеансе

В консольном сеансе Вам необходимо вручную запустить `ssh-agent` и затем вызвать `ssh-add`. Есть два способа запуска агента. Первый из приведенных ниже примеров запускает новую сессию `bash` поверх существующей. Второй пример запускает агент в существующей сессии и модифицирует окружение.

```
ssh-agent -s /bin/bash
eval $(ssh-agent)
```

После того, как агент будет запущен, выполните `ssh-add`, чтобы добавить в него свои ключи.

12.5.4 Аутентификация основанная на хостах

12.6 Проброс порта

`ssh` также может быть использован для перенаправления TCP/IP подключений. Эта опция, называемая SSH туннелированием, перенаправляет TCP соединения на определенный порт другого компьютера по зашифрованному каналу.

Используя следующую команду, любое соединение на `jupiter` порт 25 (SMTP) перенаправляется на SMTP порт на `sun`. Это особенно полезно для тех, кто использует SMTP серверы без SMTP-AUTH или POP-before-SMTP. Электронная почта, откуда бы она ни пришла, будет передана для дальнейшей доставки «домашнему» почтовому серверу.

```
ssh -L 25:sun:25 jupiter
```

Аналогично, все POP3 запросы (порт 110) на `jupiter` погут быть перенаправлены на POP3 порт `sun` при помощи следующей команды:

```
ssh -L 110:sun:110 jupiter
```

Обе команды должны быть выполнены с правами `root`, так как соединение использует привилегированные порты. При использовании электронной почты

обычными пользователями будет использоваться SSH соединение. SMTP и POP3 должны быть установлены в `localhost`, чтобы это работало. Дополнительную информацию можно найти в man-руководствах для каждой из программ, описанной выше, а также в документации проекта OpenSSH: `/usr/share/doc/packages/openssh`.

12.7 Конфигурация SSH демона при помощи YaST

Модуль YaST для настройки SSHD не входит в установку по умолчанию. Для работы с ним установите пакет `yast2-sshd`.

Для настройки sshd-сервера с помощью YaST запустите YaST и выберите *Сетевые службы > Настройка SSHD*. Затем сделайте следующее:

- 1 На вкладке *Общий*, в таблице *SSHD TCP Ports* выберите порты, которые будет использовать sshd. По умолчанию используется порт 22. Вы можете выбрать несколько портов. Чтобы добавить новый порт нажмите кнопку *Добавить*, введите номер порта и нажмите кнопку *OK*. Для удаления порта, найдите его в таблице, нажмите кнопку *Удалить* и подтвердите удаление.
- 2 Выберите опции, которые будут использоваться демоном sshd. Чтобы запретить переадресацию TCP, снимите флажок *Разрешить переадресацию TCP*. Отключение переадресации TCP не улучшает безопасность, если пользователи имеют доступ к терминалу, так как они всегда могут установить свои собственные переадресации. См. Раздел 12.6, «Проброс порта» (стр. 167) для получения дополнительной информации о переадресации TCP.

Чтобы отключить переадресацию X, снимите флажок *Разрешить переадресацию X11*. Если эта опция отключена, любые запросы переадресации X11 будут приводить к ошибке. Однако пользователи всегда могут установить свою собственную переадресацию. См. Раздел 12.1, «ssh—Secure Shell» (стр. 159) для получения дополнительной информации о переадресации X.

Опция *Разрешить сжатие* определяет, должно ли соединение между сервером и клиентом подвергаться компрессии.

- 3 Вкладка *Настройки входа в систему* содержит общие настройки входа в систему и аутентификации. В *Показывать Сообщение дня после входа в систе-*

му определяется, должен ли sshd выводить сообщение из `/etc/motd` при интерактивном режиме входа в систему. Если Вы хотите отключить возможность входить в систему пользователю `root`, отключите *Разрешить вход администратора в систему*.

Максимальное число попыток аутентификации устанавливает количество попыток аутентификации за одно соединение. *Аутентификация RSA* определяет, разрешена ли RSA аутентификация. Этот параметр применяется только к первой версии протокола SSH. *Аутентификация по открытому ключу* определяет, разрешена ли аутентификация пользователя с помощью открытого ключа. Этот параметр используется только во второй версии протокола SSH.

- 4 На вкладке *Протокол и шифрования* определяются версии протокола SSH, которые должны поддерживаться. Вы можете выбрать первую или вторую версию, а так же параллельную поддержку обоих SSH-протоколов.

В *Поддерживаемые методы шифрования* перечислены все поддерживаемые алгоритмы шифрования. Вы можете удалить шифр, выбрав его в списке и нажав кнопку *Удалить*. Чтобы добавить шифр, выберите его из выпадающего меню и нажмите кнопку *Добавить*.

- 5 Нажмите *OK* для сохранения настроек.

12.8 Дополнительная информация

<http://www.openssh.com/ru>

Страница проекта OpenSSH

<http://en.wikibooks.org/wiki/OpenSSH>

Викиучебник OpenSSH

`man sshd`

man-страница OpenSSH-демона

`man ssh_config`

Man-страница файлов конфигурации клиента SSH OpenSSH

`man scp` , `man sftp` , `man slogin` , `man ssh` , `man ssh-add` , `man ssh-agent` , `man ssh-copy-id` , `man ssh-keyconvert` , `man ssh-keygen` , `man ssh-keyscan`

man-страницы о программах копирования файлов (`scp`, `sftp`), `login` (`slogin`, `ssh`), и ключах.

Masquerading and Firewalls

Whenever Linux is used in a network environment, you can use the kernel functions that allow the manipulation of network packets to maintain a separation between internal and external network areas. The Linux `netfilter` framework provides the means to establish an effective firewall that keeps different networks apart. With the help of `iptables`—a generic table structure for the definition of rule sets—precisely control the packets allowed to pass a network interface. Such a packet filter can be set up quite easily with the help of `SuSEfirewall2` and the corresponding `YaST` module.

13.1 Packet Filtering with `iptables`

The components `netfilter` and `iptables` are responsible for the filtering and manipulation of network packets as well as for network address translation (NAT). The filtering criteria and any actions associated with them are stored in chains, which must be matched one after another by individual network packets as they arrive. The chains to match are stored in tables. The `iptables` command allows you to alter these tables and rule sets.

The Linux kernel maintains three tables, each for a particular category of functions of the packet filter:

filter

This table holds the bulk of the filter rules, because it implements the *packet filtering* mechanism in the stricter sense, which determines whether packets are let through (ACCEPT) or discarded (DROP), for example.

`nat`

This table defines any changes to the source and target addresses of packets. Using these functions also allows you to implement *masquerading*, which is a special case of NAT used to link a private network with the Internet.

`mangle`

The rules held in this table make it possible to manipulate values stored in IP headers (such as the type of service).

These tables contain several predefined chains to match packets:

PREROUTING

This chain is applied to incoming packets.

INPUT

This chain is applied to packets destined for the system's internal processes.

FORWARD

This chain is applied to packets that are only routed through the system.

OUTPUT

This chain is applied to packets originating from the system itself.

POSTROUTING

This chain is applied to all outgoing packets.

Рисунок 13.1, «iptables: A Packet's Possible Paths» (стр. 172) illustrates the paths along which a network packet may travel on a given system. For the sake of simplicity, the figure lists tables as parts of chains, but in reality these chains are held within the tables themselves.

In the simplest of all possible cases, an incoming packet destined for the system itself arrives at the `eth0` interface. The packet is first referred to the `PREROUTING` chain of the `mangle` table then to the `PREROUTING` chain of the `nat` table. The following step, concerning the routing of the packet, determines that the actual target of the packet is a process of the system itself. After passing the `INPUT` chains of the `mangle` and the `filter` table, the packet finally reaches its target, provided that the rules of the `filter` table are actually matched.

Рисунок 13.1 *iptables: A Packet's Possible Paths*

13.2 Masquerading Basics

Masquerading is the Linux-specific form of NAT (network address translation). It can be used to connect a small LAN (where hosts use IP addresses from the private range—see Раздел “Netmasks and Routing” (Глава 9, *Basic Networking*, ↑Содержание)) with the Internet (where official IP addresses are used). For the LAN hosts to be able to connect to the Internet, their private addresses are translated to an official one. This is done on the router, which acts as the gateway between the LAN and the Internet. The underlying principle is a simple one: The router has more than one network interface, typically a network card and a separate interface connecting with the Internet. While the latter links the router with the outside world, one or several others link it with the LAN hosts. With these hosts in the local network connected to the network card (such as `eth0`) of the router, they can send any packets not destined for the local network to their default gateway or router.

ВАЖНО: Using the Correct Network Mask

When configuring your network, make sure both the broadcast address and the netmask are the same for all local hosts. Failing to do so prevents packets from being routed properly.

As mentioned, whenever one of the LAN hosts sends a packet destined for an Internet address, it goes to the default router. However, the router must be configured before it can forward such packets. For security reasons, this is not enabled in a default installation. To enable it, set the variable `IP_FORWARD` in the file `/etc/sysconfig/sysctl` to `IP_FORWARD=yes`.

The target host of the connection can see your router, but knows nothing about the host in your internal network where the packets originated. This is why the technique is called masquerading. Because of the address translation, the router is the first destination of any reply packets. The router must identify these incoming packets and translate their target addresses, so packets can be forwarded to the correct host in the local network.

With the routing of inbound traffic depending on the masquerading table, there is no way to open a connection to an internal host from the outside. For such a connection, there would be no entry in the table. In addition, any connection already established has a status entry assigned to it in the table, so the entry cannot be used by another connection.

As a consequence of all this, you might experience some problems with a number of application protocols, such as ICQ, cucme, IRC (DCC, CTCP), and FTP (in PORT mode). Web browsers, the standard FTP program, and many other programs use the PASV mode. This passive mode is much less problematic as far as packet filtering and masquerading are concerned.

13.3 Firewalling Basics

Firewall is probably the term most widely used to describe a mechanism that provides and manages a link between networks while also controlling the data flow between them. Strictly speaking, the mechanism described in this section is called a *packet filter*. A packet filter regulates the data flow according to certain criteria, such as protocols, ports, and IP addresses. This allows you to block packets that, according to their addresses, are not supposed to reach your network. To allow public access to your Web server, for example, explicitly open the corresponding port. However, a packet filter does not scan the contents of packets with legitimate addresses, such as those directed to your Web server. For example, if incoming packets were intended to compromise a CGI program on your Web server, the packet filter would still let them through.

A more effective but more complex mechanism is the combination of several types of systems, such as a packet filter interacting with an application gateway or proxy. In this case, the packet filter rejects any packets destined for disabled ports. Only packets directed to the application gateway are accepted. This gateway or proxy pretends to be the actual client of the server. In a sense, such a proxy could be considered a masquerading host on the protocol level used by the application. One example for such a proxy is Squid, an HTTP and FTP proxy server. To use Squid, the browser must be configured to communicate via the proxy. Any HTTP pages or FTP files requested are served from the proxy cache and objects not found in the cache are fetched from the Internet by the proxy.

The following section focuses on the packet filter that comes with . For further information about packet filtering and firewalling, read the Firewall HOWTO included in the `howto` package. If this package is installed, read the HOWTO with

```
less /usr/share/doc/howto/en/txt/Firewall-HOWTO.gz
```

13.4 SuSEfirewall2

SuSEfirewall2 is a script that reads the variables set in `/etc/sysconfig/SuSEfirewall2` to generate a set of iptables rules. It defines three security zones, although only the first and the second one are considered in the following sample configuration:

External Zone

Given that there is no way to control what is happening on the external network, the host needs to be protected from it. In most cases, the external network is the Internet, but it could be another insecure network, such as a WLAN.

Internal Zone

This refers to the private network, in most cases the LAN. If the hosts on this network use IP addresses from the private range (see Раздел “Netmasks and Routing” (Глава 9, *Basic Networking*, ↑Содержание)), enable network address translation (NAT), so hosts on the internal network can access the external one. All ports are open in the internal zone. The main benefit of putting interfaces into the internal zone (rather than stopping the firewall) is that the firewall still runs, so when you add new interfaces, they will be put into the external zone by default. That way an interface is not accidentally «open» by default.

Demilitarized Zone (DMZ)

While hosts located in this zone can be reached both from the external and the internal network, they cannot access the internal network themselves. This setup can be used to put an additional line of defense in front of the internal network, because the DMZ systems are isolated from the internal network.

Any kind of network traffic not explicitly allowed by the filtering rule set is suppressed by iptables. Therefore, each of the interfaces with incoming traffic must be placed into one of the three zones. For each of the zones, define the services or protocols allowed. The rule set is only applied to packets originating from remote hosts. Locally generated packets are not captured by the firewall.

The configuration can be performed with YaST (see Раздел 13.4.1, «Configuring the Firewall with YaST» (ср. 176)). It can also be made manually in the file `/etc/sysconfig/SuSEfirewall2`, which is well commented. Additionally, a number of example scenarios are available in `/usr/share/doc/packages/SuSEfirewall2/EXAMPLES`.

13.4.1 Configuring the Firewall with YaST

БАЖХО: Automatic Firewall Configuration

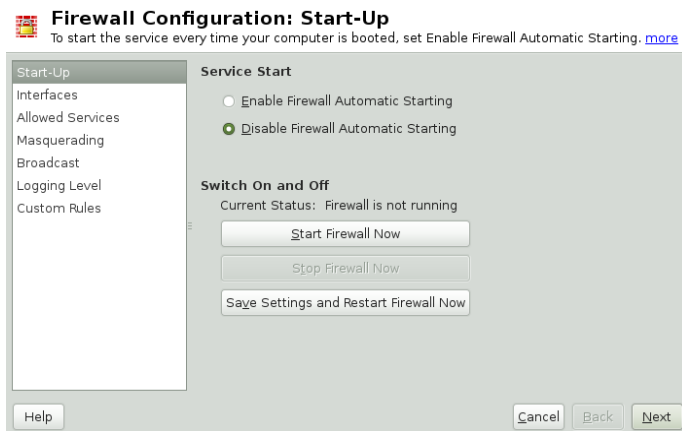
After the installation, YaST automatically starts a firewall on all configured interfaces. If a server is configured and activated on the system, YaST can modify the automatically-generated firewall configuration with the options *Open Ports on Selected Interface in Firewall* or *Open Ports on Firewall* in the server configuration modules. Some server module dialogs include a *Firewall Details* button for activating additional services and ports. The YaST firewall configuration module can be used to activate, deactivate, or reconfigure the firewall.

The YaST dialogs for the graphical configuration can be accessed from the YaST Control Center. Select *Security and Users > Firewall*. The configuration is divided into seven sections that can be accessed directly from the tree structure on the left side.

Start-Up

Set the start-up behavior in this dialog. In a default installation, SuSEfirewall2 is started automatically. You can also start and stop the firewall here. To implement your new settings in a running firewall, use *Save Settings and Restart Firewall Now*.

Рисунок 13.2 The YaST Firewall Configuration



Interfaces

All known network interfaces are listed here. To remove an interface from a zone, select the interface, press *Change*, and choose *No Zone Assigned*. To add an interface to a zone, select the interface, press *Change* and choose any of the available zones. You may also create a special interface with your own settings by using *Custom*.

Allowed Services

You need this option to offer services from your system to a zone from which it is protected. By default, the system is only protected from external zones. Explicitly allow the services that should be available to external hosts. After selecting the desired zone in *Allowed Services for Selected Zone*, activate the services from the list.

Masquerading

Masquerading hides your internal network from external networks (such as the Internet) while enabling hosts in the internal network to access the external network transparently. Requests from the external network to the internal one are blocked and requests from the internal network seem to be issued by the masquerading server when seen externally. If special services of an internal machine need to be available to the external network, add special redirect rules for the service.

Broadcast

In this dialog, configure the UDP ports that allow broadcasts. Add the required port numbers or services to the appropriate zone, separated by spaces. See also the file `/etc/services`.

The logging of broadcasts that are not accepted can be enabled here. This may be problematic, because Windows hosts use broadcasts to know about each other and so generate many packets that are not accepted.

IPsec Support

Configure whether the IPsec service should be available to the external network in this dialog. Configure which packets are trusted under *Details*.

There is another functionality under *Details*: IPsec packets are packed in an encrypted format, so they have to be decrypted and you can configure the way the firewall will handle the decrypted packets. If you select *Internal Zone*, the decrypted IPsec packets will be trusted as if they came from the Internal Zone - although they could possibly come from the external one. Choose *Same Zone as Original Source Network* to avoid this situation.

Logging Level

There are two rules for logging: accepted and not accepted packets. Packets that are not accepted are DROPPED or REJECTED. Select from *Log All*, *Log Only Critical*, or *Do Not Log Any*.

Custom Rules

Here, set special firewall rules that allow connections, matching specified criteria such as source network, protocol, destination port, and source port. Configure such rules for external, internal, and demilitarized zones.

When finished with the firewall configuration, exit this dialog with *Next*. A zone-oriented summary of your firewall configuration then opens. In it, check all settings. All services, ports, and protocols that have been allowed and all custom rules are listed in this summary. To modify the configuration, use *Back*. Press *Finish* to save your configuration.

13.4.2 Configuring Manually

The following paragraphs provide step-by-step instructions for a successful configuration. Each configuration item is marked as to whether it is relevant to firewalling or masquerading. Use port range (for example, 500 : 510) whenever appropriate. Aspects related to the DMZ (demilitarized zone) as mentioned in the configuration file are not covered here. They are applicable only to a more complex network infrastructure found in larger organizations (corporate networks), which require extensive configuration and in-depth knowledge about the subject.

First, use the YaST module Системные службы (Уровень запуска) to enable SuSEfirewall2 in your runlevel (3 or 5 most likely). It sets the symlinks for the SuSEfirewall2_* scripts in the /etc/init.d/rc?.d/ directories.

FW_DEV_EXT (firewall, masquerading)

The device linked to the Internet. For a modem connection, enter ppp0. For an ISDN link, use ippp0. DSL connections use.dsl0. Specify auto to use the interface that corresponds to the default route.

FW_DEV_INT (firewall, masquerading)

The device linked to the internal, private network (such as eth0). Leave this blank if there is no internal network and the firewall protects only the host on which it runs.

FW_ROUTE (firewall, masquerading)

If you need the masquerading function, set this to `yes`. Your internal hosts will not be visible to the outside, because their private network addresses (e.g., `192.168.x.x`) are ignored by Internet routers.

For a firewall without masquerading, set this to `yes` if you want to allow access to the internal network. Your internal hosts need to use officially registered IP addresses in this case. Normally, however, you should *not* allow access to your internal network from the outside.

FW_MASQUERADE (masquerading)

Set this to `yes` if you need the masquerading function. This provides a virtually direct connection to the Internet for the internal hosts. It is more secure to have a proxy server between the hosts of the internal network and the Internet. Masquerading is not needed for services that a proxy server provides.

FW_MASQ_NETS (masquerading)

Specify the hosts or networks to masquerade, leaving a space between the individual entries. For example:

```
FW_MASQ_NETS="192.168.0.0/24 192.168.10.1"
```

FW_PROTECT_FROM_INT (firewall)

Set this to `yes` to protect your firewall host from attacks originating in your internal network. Services are only available to the internal network if explicitly enabled. Also see `FW_SERVICES_INT_TCP` and `FW_SERVICES_INT_UDP`.

FW_SERVICES_EXT_TCP (firewall)

Enter the TCP ports that should be made available. Leave this blank for a normal workstation at home that should not offer any services.

FW_SERVICES_EXT_UDP (firewall)

Leave this blank unless you run a UDP service and want to make it available to the outside. The services that use UDP include DNS servers, IPsec, TFTP, DHCP and others. In that case, enter the UDP ports to use.

FW_SERVICES_ACCEPT_EXT (firewall)

List services to allow from the Internet. This is a more generic form of the `FW_SERVICES_EXT_TCP` and `FW_SERVICES_EXT_UDP` settings, and more specific than `FW_TRUSTED_NETS`. The notation is a space-separated list of `net, protocol[, dport] [, sport]`, for example `0/0, tcp, 22` or `0/0, tcp, 22, , hitcount=3, blockseconds=60, recentname=ssh`,

which means: allow a maximum of three SSH connects per minute from one IP address.

FW_SERVICES_INT_TCP (firewall)

With this variable, define the services available for the internal network.

The notation is the same as for FW_SERVICES_EXT_TCP, but the settings are applied to the *internal* network. The variable only needs to be set if FW_PROTECT_FROM_INT is set to *yes*.

FW_SERVICES_INT_UDP (firewall)

See FW_SERVICES_INT_TCP.

FW_SERVICES_ACCEPT_INT (firewall)

List services to allow from internal hosts. See FW_SERVICES_ACCEPT_EXT.

FW_SERVICES_ACCEPT_RELATED_* (firewall)

This is how the SuSEfirewall2 implementation considers packets RELATED by netfilter.

For example, to allow finer grained filtering of Samba broadcast packets, RELATED packets are not accepted unconditionally. Variables starting with FW_SERVICES_ACCEPT_RELATED_ allow restricting RELATED packets handling to certain networks, protocols and ports.

This means that adding connection tracking modules (conntrack modules) to FW_LOAD_MODULES does not automatically result in accepting the packets tagged by those modules. Additionally, you must set variables starting with FW_SERVICES_ACCEPT_RELATED_ to a suitable value.

FW_CUSTOMRULES (firewall)

Uncomment this variable to install custom rules. Find examples in `/etc/sysconfig/scripts/SuSEfirewall2-custom`.

After configuring the firewall, test your setup. The firewall rule sets are created by entering `SuSEfirewall2 start` as root. Then use `telnet`, for example, from an external host to see whether the connection is actually denied. After that, review `/var/log/messages`, where you should see something like this:

```
Mar 15 13:21:38 linux kernel: SFW2-INext-DROP-DEFLT IN=eth0
OUT= MAC=00:80:c8:94:c3:e7:00:a0:c9:4d:27:56:08:00 SRC=192.168.10.0
DST=192.168.10.1 LEN=60 TOS=0x10 PREC=0x00 TTL=64 ID=15330 DF PROTO=TCP
SPT=48091 DPT=23 WINDOW=5840 RES=0x00 SYN URG=0
OPT (020405B40402080A061AFEB0000000001030300)
```

Other packages to test your firewall setup are Nmap (portscanner) or OpenVAS (Open Vulnerability Assessment System). The documentation of Nmap is found at `/usr/share/doc/packages/nmap` after installing the package and the documentation of openVAS resides at <http://www.openvas.org>.

13.5 For More Information

The most up-to-date information and other documentation about the SuSEfirewall2 package is found in `/usr/share/doc/packages/SuSEfirewall2`. The home page of the netfilter and iptables project, <http://www.netfilter.org>, provides a large collection of documents in many languages.

Configuring VPN Server

Nowadays, the Internet connection is cheap and available almost everywhere. It is important that the connection is as secure as possible. Virtual Private Network (VPN), is a secure network within a second, insecure network such as the Internet or WLAN. It can be implemented in different ways and serves several purposes. In this chapter, we focus on VPNs to link branch offices via secure wide area networks (WANs).

14.1 Conceptual Overview

This section defines some term regarding to VPN and introduces a brief overview of some scenarios.

14.1.1 Terminology

Endpoint

The two «ends» of a tunnel, the source or destination client

Tap Device

A tap device simulates an Ethernet device (layer 2 packets in the OSI model such as IP packets). A tap device is used for creating a network bridge. It works with Ethernet frames.

Tun Device

A tun device simulates a point-to-point network (layer 3 packets in the OSI model such as Ethernet frames). A tun device is used with routing and works with IP frames.

Tunnel

Linking two locations through a primarily public network. From a more technical viewpoint, it is a connection between the client's device and the server's device. Usually a tunnel is encrypted, but it does need to be by definition.

14.1.2 VPN Scenarios

Whenever you setup a VPN connection your IP packets are transferred over your secured *tunnel*. A tunnel can use a so-called *tun* or *tap* device. They are virtual network kernel drivers which implement the transmission of ethernet frames or ip frames/packets.

Any userspace program OpenVPN can attach itself to a tun or tap device to receive packets sent by your OS. The program is also able to write packets to the device.

There are many solutions to set up and build a VPN connection. This section focuses on the OpenVPN package. Compared to other VPN software, OpenVPN can be operated in two modes:

Routed VPN

Routing is an easy solution to set up. It is more efficient and scales better than bridged VPN. Furthermore, it allows the user to tune MTU (Maximum Transfer Unit) to raise efficiency. However, in a heterogeneous environment NetBIOS broadcasts do not work if you do not have a Samba server on the gateway. If you need IPv6, each tun drivers on both ends must support this protocol explicitly. This scenario is depicted in Рисунок 14.1, «Routed VPN» (стр. 184)

Рисунок 14.1 *Routed VPN*

Bridged VPN

Bridging is a more complex solution. It is recommended when you need to browse Windows file shares across the VPN without setting up a Samba or WINS server. Bridged VPN is also needed if you want to use non-IP protocols (such as IPX) or applications relying on network broadcasts. However, it is less efficient than routed VPN. Another disadvantage is that it does not scale well. This scenarios is depicted in the following figures.

Рисунок 14.2 *Bridged VPN - Scenario 1*

Рисунок 14.3 *Bridged VPN - Scenario 2*

Рисунок 14.4 *Bridged VPN - Scenario 3*

The major difference between bridging and routing is that a routed VPN cannot IP-broadcast while a bridged VPN can.

14.1.3 Tun and Tap Devices

Whenever you setup a VPN connection your IP packets are transferred over your secured tunnel. The connection between the client's device and the server's device is called a *tunnel*. A tunnel can use a so-called *tun* or *tap* device. They are virtual network kernel drivers which implement the transmission of ethernet frames or ip frames/packets:

tun device

A tun device simulates a point-to-point network (layer 3 packets in the OSI model such as Ethernet frames). A tun device is used with routing and works with IP frames.

tap device

A tap device simulates an ethernet device (layer 2 packets in the OSI model such as IP packets). A tap device is used for creating a network bridge. It works with Ethernet frames.

The userspace program OpenVPN can attach itself to a tun or tap device to receive packets sent by your OS. The program is also able to write packets to the device. For more information, see `/usr/src/linux/Documentation/networking/tuntap.txt`. You must install the `kernel-source` package to read this file.

14.2 Creating the Simplest VPN Example

The following example creates a point-to-point VPN tunnel. It demonstrates how to create a VPN tunnel between one client and a server. It is assumed that your VPN

server will use private IP addresses like 192.168.1.120 and your client the IP address 192.168.2.110. You can modify these private IP addresses to your needs but make sure you select addresses which do not conflict with other IP addresses.

ПРЕДУПРЕЖДЕНИЕ: Use It Only For Testing

This scenario is only useful for testing and is considered as an example to get familiar with VPN. *Do not use* this as a real world scenario to connect as it can compromise your security and the safety of your IT infrastructure!

14.2.1 Configuring the VPN Server

To configure a VPN server, proceed as follows:

Процедура 14.1 VPN Server Configuration

- 1 Install the package `openvpn` on the machine that will later become your VPN server.

- 2 Open a shell, become `root` and create the VPN secret key:

```
openvpn --genkey --secret /etc/openvpn/secret.key
```

- 3 Copy the secret key to your client:

```
scp /etc/openvpn/secret.key root@192.168.2.110:/etc/openvpn/
```

- 4 Create the file `/etc/openvpn/server.conf` with the following content:

```
dev tun
ifconfig 192.168.1.120 192.168.2.110
secret secret.key
```

- 5 If you use a firewall, start YaST and open UDP port 1194 (*Security and Users > Firewall > Allowed Services*).

- 6 Start the OpenVPN service as `root`:

```
rcopenvpn start
```

14.2.2 Configuring the VPN Client

To configure the VPN client, do the following:

Процедура 14.2 *VPN Client Configuration*

- 1 Install the package `openvpn` on your client VPN machine.
- 2 Create `/etc/openvpn/client.conf` with the following content:

```
remote IP_OF_SERVER
dev tun
ifconfig 192.168.2.110 192.168.1.120
secret secret.key
```

Replace the placeholder `IP_OF_SERVER` in the first line with either the domain name, or the public IP address of your server.

- 3 If you use a firewall, start YaST and open UDP port 1194 as described in [Chapter 5](#) (стр. 186) of *Процедура 14.1, «VPN Server Configuration»* (стр. 186).
- 4 Start the OpenVPN service as `root`:

```
rcopenvpn start
```

14.2.3 Testing the VPN Example

After the OpenVPN is successfully started, test if the `tun` device is available with the following command:

```
ifconfig tun0
```

To verify the VPN connection, use `ping` on both client and server to see if you can reach each other. Ping server from client:

```
ping -I tun0 192.168.1.120
```

Ping client from server:

```
ping -I tun0 192.168.2.110
```

14.3 Setting Up Your VPN Server Using Certificate Authority

The example shown in [Раздел 14.2](#) (стр. 185) is useful for testing, but not for daily work. This section explains how to build a VPN server that allows more than

one connection at the same time. This is done with a public key infrastructure (PKI). A PKI consists of a pair of public and private keys for the server and each client and a master certificate authority (CA), which is used to sign every server and client certificate.

The general overview of this process involves the following steps explained in these sections:

- 1 Раздел 14.3.1, «Creating Certificates» (стр. 188)
- 2 Раздел 14.3.2, «Configuring the Server» (стр. 190)
- 3 Раздел 14.3.3, «Configuring the Clients» (стр. 192)

14.3.1 Creating Certificates

Before a VPN connection gets established, the client must authenticate the server certificate. Conversely, the server must also authenticate the client certificate. This is called *mutual authentication*.

You can use two methods to create the respective certificates and keys:

- Use the YaST CA module (see Глава 15, *Managing X.509 Certification* (стр. 197)), or
- Use the scripts included with the `openvpn` package.

14.3.1.1 Generating Certificates with easy-rsa

The `easy-rsa` utilities use the `openssl.cnf` file stored under `/usr/share/openvpn/easy-rsa/VER/`. In most cases you can leave this file as it is.

Процедура 14.3 *Generate the Master CA And Key*

- 1 Open a shell and become `root`.
- 2 Change the directory to `/usr/share/openvpn/easy-rsa/VER/`. Replace the placeholder `VER` with the current version— either `1.0` or `2.0`.

- 3 Copy the file `vars` to `/etc/openvpn` and set `export EASY_RSA` to `/usr/share/openvpn/easy-rsa`:

```
export EASY_RSA="/usr/share/openvpn/easy-rsa/VER"
```

- 4 In the `vars` file change the `KEY_COUNTRY`, `KEY_PROVINCE`, `KEY_CITY`, `KEY_ORG`, and `KEY_EMAIL` variables according to your needs.

- 5 Initialize the PKI:

```
source /etc/openvpn/vars && ./clean-all && ./build-ca
```

- 6 Enter the data required by the `build-ca` script. Usually you can take the defaults that you have set in Шаг 4 (стр. 189). Additionally set `Organizational Unit Name` and `Common Name` that were not set previously.

Once done, the master certificate and key are saved as `/usr/share/openvpn/easy-rsa/VER/keys/ca.*`.

Процедура 14.4 *Generate The Private Server Key*

- 1 Change to the `/usr/share/openvpn/easy-rsa/VER/` directory.
- 2 Run the following script:

```
./build-key-server server
```

The argument (here: `server`) is used for the private key filename.

- 3 Accept the default parameters, but fill `server` for the `Common Name` option.
- 4 Answer the next two questions («Sign the certificate? [y/n]» and «1 out of 1 certificate requests certified, commit? [y/n]») with `y` (yes).

Once done, the private server key is saved as `/usr/share/openvpn/easy-rsa/VER/keys/server.*`.

Процедура 14.5 *Generate Certificates and Keys for a Client*

- 1 Change to the `/usr/share/openvpn/easy-rsa/VER/` directory. Replace the placeholder `VER` with either `1.0` or `2.0`.
- 2 Create the key as in Шаг 2 (стр. 189) of Процедура 14.4, «Generate The Private Server Key» (стр. 189):

```
./build-key client
```

- 3 Repeat the previous step for each client that is allowed to connect to the VPN server. Make sure you use a different name (other than «client») and an appropriate Common Name, because this parameter has to be unique for each client.

Once done, the client certificate keys are saved as `/usr/share/openvpn/easy-rsa/keys/client.*` (depending on the name that you have given for the `build-key` command).

Процедура 14.6 *Final Configuration Steps*

- 1 Make sure your current working directory is `/usr/share/openvpn/easy-rsa/VER/`.

- 2 Create the Diffie-Hellman parameter:

```
./build-dh
```

- 3 Create the `/etc/openvpn/ssl` directory.

- 4 Copy the following files to `/etc/openvpn/ssl`:

```
cp keys/ca.{crt,key} keys/dh1024.pem keys/server.{crt,key} /etc/openvpn/ssl
```

- 5 Copy the client keys to the relevant client machine. You should have the files `client.crt` and `client.key` in the `/etc/openvpn/ssl` directory.

14.3.2 Configuring the Server

The configuration file is mostly a summary of `/usr/share/doc/packages/openvpn/sample-config-files/server.conf` without the comments and with some small changes concerning some paths.

Пример 14.1 *VPN Server Configuration File*

```
# /etc/openvpn/server.conf
port 1194 ❶
proto udp ❷
dev tun0 ❸
```

```

# Security ❹
ca    ssl/ca.crt
cert  ssl/server.crt
key   ssl/server.key
dh    ssl/dh1024.pem

server 192.168.1.120 255.255.255.0 ❺
ifconfig-pool-persist /var/run/openvpn/ipp.txt ❻

# Privileges ❼
user nobody
group nobody

# Other configuration ❸
keepalive 10 120
comp-lzo
persist-key
persist-tun
status      /var/log/openvpn-status.log
log-append  /var/log/openvpn.log
verb 4

```

- ❶ The TCP/UDP port to which OpenVPN listens. You have to open up the port in the Firewall, see Глава 13, *Masquerading and Firewalls* (стр. 171). The standard port for VPN is 1194, so in most cases you can leave that as it is.
- ❷ The protocol, either UDP or TCP.
- ❸ The tun or tap device, see Раздел 14.1.3, «Tun and Tap Devices» (стр. 185) for the differences.
- ❹ The following lines contain the relative or absolute path to the root server CA certificate (`ca`), the root CA key (`cert`), the private server key (`key`) and the Diffie-Hellman parameters (`dh`). These were generated in Раздел 14.3.1, «Creating Certificates» (стр. 188).
- ❺ Supplies a VPN subnet. The server can be reached by `192.168.1.120`.
- ❻ Records a mapping of clients and its virtual IP address in the given file. Useful when the server goes down and (after the restart) the clients get their previously assigned IP address.
- ❼ For security reasons it is a good idea to run the OpenVPN daemon with reduced privileges. For this reason the group and user `nobody` is used.
- ❸ Several other configurations, see comment in the original configuration from `/usr/share/doc/packages/openvpn/sample-config-files`.

After this configuration, you can see log messages from your OpenVPN server under `/var/log/openvpn.log`. When you have started it for the first time, it should finish it with:

```
... Initialization Sequence Completed
```

If you do not see this message, check the log carefully. Usually OpenVPN gives you some hints what is wrong in your configuration file.

14.3.3 Configuring the Clients

The configuration file is mostly a summary from `/usr/share/doc/packages/openvpn/sample-config-files/client.conf` without the comments and with some small changes concerning some paths.

Пример 14.2 *VPN Client Configuration File*

```
# /etc/openvpn/client.conf
client ❶
dev tun ❷
proto udp ❸
remote IP_OR_HOSTNAME 1194 ❹
resolv-retry infinite
nobind

# Privileges ❺
user nobody
group nobody

# Try to preserve some state across restarts.
persist-key
persist-tun

# Security ❻
ca    ssl/ca.crt
cert  ssl/client.crt
key   ssl/client.key

comp-lzo ❼
```

- ❶ We must specify that this machine is a client.
- ❷ The network device. Both clients and server must use the same device.
- ❸ The protocol. Use the same settings as on the server.
- ❹ Replace the placeholder `IP_OR_HOSTNAME` with the respective hostname or IP address of your VPN server. After the hostname the port of the server is given. You can have multiple lines of `remote` entries pointing to different VPN servers. This is useful for load balancing between different VPN servers.
- ❺ For security reasons it is a good idea to run the OpenVPN daemon with reduced privileges. For this reason the group and user `nobody` is used.
- ❻ Contains the client files. For security reasons, it is better to have a separate file pair for each client.

- ⑦ Turns compression on. Use it only when the server has this parameter switched on as well.

14.4 Changing Nameservers in VPN

If you need to change nameservers before or during your VPN session, use `netconfig`.

Use the following procedure to change a nameserver:

Процедура 14.7 Changing Nameservers

- 1 Open a shell and log in as `root`.
- 2 Create the file `/etc/openvpn/client.up` with the following contents:

```
/sbin/netconfig modify -i "${1}" -s openvpn <<EOT
DNSSEARCH='${domain}'
DNSSERVERS='${dns[*]}'
EOT
```

- 3 Start your VPN connection with `rcopenvpn start`.

- 4 Create the file `/etc/openvpn/client.down` with the following contents:

```
/sbin/netconfig remove -i "${1}" -s openvpn
```

- 5 Run `netconfig` and replace the line `DNSSERVERS` with your respective entry:

```
netconfig modify -i tun0 -s openvpn <<EOT
DNSSEARCH='mt-home.net'
DNSSERVERS='192.168.1.116'
EOT
```

To check, if the entry has been successfully inserted into `/etc/resolv.conf`, execute:

```
grep -v ^# /etc/resolv.conf
search mt-home.net mat-home.net
nameserver ...
nameserver ...
nameserver 192.168.1.116
```

- 6 To remove the DNS entry, execute:

```
netconfig remove -i tun0 -s openvpn
```

Find another example in `/usr/share/doc/packages/openvpn/contrib/pull-resolv-conf/`.

If you need to specify a ranking list of fallback services, use the `NETCONFIG_DNS_RANKING` variable in `/etc/sysconfig/network/config`. The default value is `auto` which resolves to:

```
+strongswan +openswan +racoon +openvpn -avahi
```

Preferred service names have the `+` prefix, fallback services the `-` prefix.

14.5 KDE- and GNOME Applets For Clients

The following sections describe the setup of OpenVPN connections with the GNOME and KDE desktop tools.

14.5.1 KDE

To setup an OpenVPN connection in KDE4 that can be easily turned on or off, proceed as follows:

- 1 Make sure you have installed the `NetworkManager-openvpn-kde4` package with all dependencies resolved.
- 2 Right-click on a widget of your panel and select *Panel Options > Add Widgets...*
- 3 Select *Networks*.
- 4 Right-click on the icon and choose *Manage Connections*.
- 5 Add a new VPN connection with *Add > OpenVPN*. A new window opens.
- 6 Choose the *Connection Type* between *X.509 Certificates* or *X.509 With Password* depending on what you have setup with your OpenVPN server.
- 7 Insert the necessary files into the respective text fields. From our example configuration these are:

<i>CA file</i>	/etc/openvpn/ssl/ca.crt
<i>Certificate</i>	/etc/openvpn/ssl/ client1.crt
<i>Key</i>	/etc/openvpn/ssl/ client1.key
<i>Username</i>	The user
<i>Password</i>	The password for the user

8 If you have not used the KDE Wallet System, you are asked if you want to configure it. Follow the steps in the wizard. After you have finished this step, you are reverted back to the *Network Settings* dialog.

9 Finish with *Ok*.

10 Enable the connection with your Network manager applet.

14.5.2 GNOME

To setup a OpenVPN connection in GNOME that can be easily turned on or off, proceed as follows:

- 1** Make sure you have installed the package `NetworkManager-openvpn-gnome` and have resolved all dependencies.
- 2** Start the Network Connection Editor with `+ F2` and insert `nm-connection-editor` into the text field. A new window appears.
- 3** Select the *VPN* tab and click *Add*.
- 4** Choose the VPN connection type, in this case *OpenVPN*.
- 5** Choose the *Authentication* type. Select between *Certificates (TLS)* or *Password with Certificates (TLS)* depending on the setup of your OpenVPN server.

- 6 Insert the necessary files into the respective text fields. According to the example configuration, these are:

<i>Username</i>	The user (only available when you have selected <i>Password with Certificates (TLS)</i>)
<i>Password</i>	The password for the user (only available when you have selected <i>Password with Certificates (TLS)</i>)
<i>User Certificate</i>	/etc/openvpn/ssl/ client1.crt
<i>CA Certificate</i>	/etc/openvpn/ssl/ca.crt
<i>Private Key</i>	/etc/openvpn/ssl/ client1.key

- 7 Finish with *Apply* and *Close*.
- 8 Enable the connection with your Network Manager applet.

14.6 For More Information

For more information about VPN, visit:

- <http://www.openvpn.net>: Homepage of VPN
- /usr/share/doc/packages/openvpn/sample-config-files/: Examples of configuration files for different scenarios
- /usr/src/linux/Documentation/networking/tuntap.txt, install the kernel-source package

Managing X.509 Certification

An increasing number of authentication mechanisms are based on cryptographic procedures. Digital certificates that assign cryptographic keys to their owners play an important role in this context. These certificates are used for communication and can also be found, for example, on company ID cards. The generation and administration of certificates is mostly handled by official institutions that offer this as a commercial service. In some cases, however, it may make sense to carry out these tasks yourself. For example, if a company does not wish to pass personal data to third parties.

YaST provides two modules for certification, which offer basic management functions for digital X.509 certificates. The following sections explain the basics of digital certification and how to use YaST to create and administer certificates of this type. For more detailed information, refer to <http://www.ietf.org/html.charters/pkix-charter.html>.

15.1 The Principles of Digital Certification

Digital certification uses cryptographic processes to encrypt and protect data from access by unauthorized people. The user data is encrypted using a second data record, or *key*. The key is applied to the user data in a mathematical process, producing an altered data record in which the original content can no longer be identified. Asymmetrical encryption is now in general use (*public key method*). Keys always occur in pairs:

Private Key

The private key must be kept safely by the key owner. Accidental publication of the private key compromises the key pair and renders it useless.

Public Key

The key owner circulates the public key for use by third parties.

15.1.1 Key Authenticity

Because the public key process is in widespread use, there are many public keys in circulation. Successful use of this system requires that every user be sure that a public key actually belongs to the assumed owner. The assignment of users to public keys is confirmed by trustworthy organizations with public key certificates. Such certificates contain the name of the key owner, the corresponding public key, and the electronic signature of the person issuing the certificate.

Trustworthy organizations that issue and sign public key certificates are usually part of a certification infrastructure that is also responsible for the other aspects of certificate management, such as publication, withdrawal, and renewal of certificates. An infrastructure of this kind is generally referred to as a *public key infrastructure* or *PKI*. One familiar PKI is the *OpenPGP* standard in which users publish their certificates themselves without central authorization points. These certificates become trustworthy when signed by other parties in the «web of trust.»

The *X.509 Public Key Infrastructure* (PKIX) is an alternative model defined by the *IETF* (Internet Engineering Task Force) that serves as a model for almost all publicly-used PKIs today. In this model, authentication is made by *certificate authorities* (CA) in a hierarchical tree structure. The root of the tree is the root CA, which certifies all sub-CAs. The lowest level of sub-CAs issue user certificates. The user certificates are trustworthy by certification that can be traced to the root CA.

The security of such a PKI depends on the trustworthiness of the CA certificates. To make certification practices clear to PKI customers, the PKI operator defines a *certification practice statement* (CPS) that defines the procedures for certificate management. This should ensure that the PKI only issues trustworthy certificates.

15.1.2 X.509 Certificates

An X.509 certificate is a data structure with several fixed fields and, optionally, additional extensions. The fixed fields mainly contain the name of the key owner,

the public key, and the data relating to the issuing CA (name and signature). For security reasons, a certificate should only have a limited period of validity, so a field is also provided for this date. The CA guarantees the validity of the certificate in the specified period. The CPS usually requires the PKI (the issuing CA) to create and distribute a new certificate before expiration.

The extensions can contain any additional information. An application is only required to be able to evaluate an extension if it is identified as *critical*. If an application does not recognize a critical extension, it must reject the certificate. Some extensions are only useful for a specific application, such as signature or encryption.

Таблица 15.1 shows the fields of a basic X.509 certificate in version 3.

Таблица 15.1 X.509v3 Certificate

Field	Content
Version	The version of the certificate, for example, v3
Serial Number	Unique certificate ID (an integer)
Signature	The ID of the algorithm used to sign the certificate
Issuer	Unique name (DN) of the issuing authority (CA)
Validity	Period of validity
Subject	Unique name (DN) of the owner
Subject Public Key Info	Public key of the owner and the ID of the algorithm
Issuer Unique ID	Unique ID of the issuing CA (optional)
Subject Unique ID	Unique ID of the owner (optional)

Field	Content
Extensions	Optional additional information, such as «KeyUsage» or «BasicConstraints»

15.1.3 Blocking X.509 Certificates

If a certificate becomes untrustworthy before it has expired, it must be blocked immediately. This can become necessary if, for example, the private key has accidentally been made public. Blocking certificates is especially important if the private key belongs to a CA rather than a user certificate. In this case, all user certificates issued by the relevant CA must be blocked immediately. If a certificate is blocked, the PKI (the responsible CA) must make this information available to all those involved using a *certificate revocation list* (CRL).

These lists are supplied by the CA to public CRL distribution points (CDPs) at regular intervals. The CDP can optionally be named as an extension in the certificate, so a checker can fetch a current CRL for validation purposes. One way to do this is the *online certificate status protocol* (OCSP). The authenticity of the CRLs is ensured with the signature of the issuing CA. Таблица 15.2 shows the basic parts of a X.509 CRL.

Таблица 15.2 X.509 Certificate Revocation List (CRL)

Field	Content
Version	The version of the CRL, such as v2
Signature	The ID of the algorithm used to sign the CRL
Issuer	Unique name (DN) of the publisher of the CRL (usually the issuing CA)
This Update	Time of publication (date, time) of this CRL
Next Update	Time of publication (date, time) of the next CRL

Field	Content
List of revoked certificates	Every entry contains the serial number of the certificate, the time of revocation, and optional extensions (CRL entry extensions)
Extensions	Optional CRL extensions

15.1.4 Repository for Certificates and CRLs

The certificates and CRLs for a CA must be made publicly accessible using a *repository*. Because the signature protects the certificates and CRLs from being forged, the repository itself does not need to be secured in a special way. Instead, it tries to grant the simplest and fastest access possible. For this reason, certificates are often provided on an LDAP or HTTP server. Find explanations about LDAP in Глава 4, *LDAP — Сервис директорий* (стр. 41). Глава 16, *The Apache HTTP Server* (↑Содержание) contains information about the HTTP server.

15.1.5 Proprietary PKI

YaST contains modules for the basic management of X.509 certificates. This mainly involves the creation of CAs, sub-CAs, and their certificates. The services of a PKI go far beyond simply creating and distributing certificates and CRLs. The operation of a PKI requires a well-conceived administrative infrastructure allowing continuous update of certificates and CRLs. This infrastructure is provided by commercial PKI products and can also be partly automated. YaST provides tools for creating and distributing CAs and certificates, but cannot currently offer this background infrastructure. To set up a small PKI, you can use the available YaST modules. However, you should use commercial products to set up an «official» or commercial PKI.

15.2 YaST Modules for CA Management

YaST provides two modules for basic CA management. The primary management tasks with these modules are explained here.

15.2.1 Creating a Root CA

The first step when setting up a PKI is to create a root CA. Do the following:

- 1 Start YaST and go to *Security and Users > CA Management*.
- 2 Click *Create Root CA*.
- 3 Enter the basic data for the CA in the first dialog, shown in Рисунок 15.1. The text fields have the following meanings:

Рисунок 15.1 *YaST CA Module—Basic Data for a Root CA*

Create New Root CA (step 1/3)

CA Name:

Common Name:

E-Mail Addresses ▾ default

✓

Organization:

Organizational Unit:

Locality:

State:

Country:

CA Name

Enter the technical name of the CA. Directory names, among other things, are derived from this name, which is why only the characters listed in the help can

be used. The technical name is also displayed in the overview when the module is started.

Common Name

Enter the name for use in referring to the CA.

E-Mail Addresses

Several e-mail addresses can be entered that can be seen by the CA user. This can be helpful for inquiries.

Country

Select the country where the CA is operated.

Organisation, Organisational Unit, Locality, State

Optional values

Proceed with *Next*.

- 4 Enter a password in the second dialog. This password is always required when using the CA—when creating a sub-CA or generating certificates. The text fields have the following meaning:

Key Length

Key Length contains a meaningful default and does not generally need to be changed unless an application cannot deal with this key length. The higher the number the more secure your password is.

Valid Period (days)

The *Valid Period* in the case of a CA defaults to 3650 days (roughly ten years). This long period makes sense because the replacement of a deleted CA involves an enormous administrative effort.

Clicking *Advanced Options* opens a dialog for setting different attributes from the X.509 extensions (Рисунок 15.4, «YaST CA Module—Extended Settings» (стр. 209)). These values have rational default settings and should only be changed if you are really sure of what you are doing. Proceed with *Next*.

- 5 Review the summary. YaST displays the current settings for confirmation. Click *Create*. The root CA is created then appears in the overview.

ПОДСКАЗКА

In general, it is best not to allow user certificates to be issued by the root CA. It is better to create at least one sub-CA and create the user certificates from there. This has the advantage that the root CA can be kept isolated and secure, for example, on an isolated computer on secure premises. This makes it very difficult to attack the root CA.

15.2.2 Changing Password

If you need to change your password for your CA, proceed as follows:

- 1 Start YaST and open the CA module.
- 2 Select the required root CA and click *Enter CA*.
- 3 Enter the password if you entered a CA the first time. YaST displays the CA key information in the *Description* tab (see Рисунок 15.2).
- 4 Click *Advanced* and select *Change CA Password*. A dialog box opens.
- 5 Enter the old and the new password.
- 6 Finish with *OK*

15.2.3 Creating or Revoking a Sub-CA

A sub-CA is created in exactly the same way as a root CA.

ПРИМЕЧАНИЕ

The validity period for a sub-CA must be fully within the validity period of the «parent» CA. A sub-CA is always created after the «parent» CA, therefore, the default value leads to an error message. To avoid this, enter a permissible value for the period of validity.

Do the following:

- 1 Start YaST and open the CA module.

- 2 Select the required root CA and click *Enter CA*.
- 3 Enter the password if you are entering a CA for the first time. YaST displays the CA key information in the tab *Description* (see Рисунок 15.2).

Рисунок 15.2 *YaST CA Module—Using a CA*



- 4 Click *Advanced* and select *Create SubCA*. This opens the same dialog as for creating a root CA.
- 5 Proceed as described in Раздел 15.2.1, «Creating a Root CA» (стр. 202).

It is possible to use one password for all your CAs. Enable *Use CA Password as Certificate Password* to give your sub-CAs the same password as your root CA. This helps to reduce the amount of passwords for your CAs.

ПРИМЕЧАНИЕ: Check your Valid Period

Take into account that the valid period must be lower than the valid period in the root CA.

- 6 Select the *Certificates* tab. Reset compromised or otherwise unwanted sub-CAs here, using *Revoke*. Revocation alone is not enough to deactivate a sub-CA. You must also publish revoked sub-CAs in a CRL. The creation of CRLs is described in Раздел 15.2.6, «Creating Certificate Revocation Lists (CRLs)» (стр. 209).

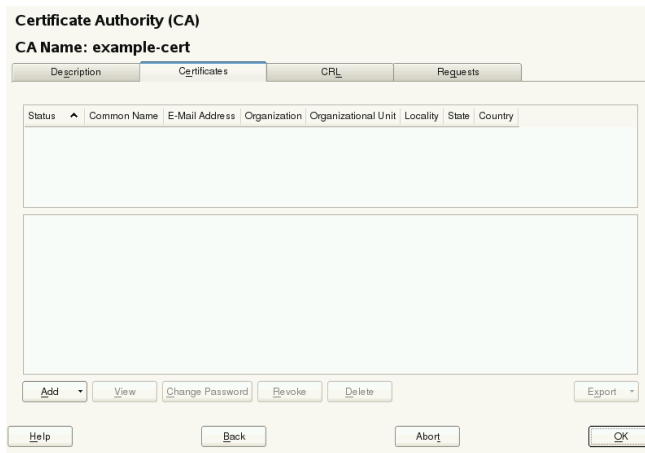
15.2.4 Creating or Revoking User Certificates

Creating client and server certificates is very similar to creating CAs in Раздел 15.2.1, «Creating a Root CA» (стр. 202). The same principles apply here. In certificates intended for e-mail signature, the e-mail address of the sender (the private key owner) should be contained in the certificate to enable the e-mail program to assign the correct certificate. For certificate assignment during encryption, it is necessary for the e-mail address of the recipient (the public key owner) to be included in the certificate. In the case of server and client certificates, the hostname of the server must be entered in the *Common Name* field. The default validity period for certificates is 365 days.

To create client and server certificates, do the following:

- 1 Start YaST and open the CA module.
- 2 Select the required root CA and click *Enter CA*.
- 3 Enter the password if you are entering a CA for the first time. YaST displays the CA key information in the *Description* tab.
- 4 Click *Certificates* (see Рисунок 15.3).

Рисунок 15.3 Certificates of a CA



- 5 Click *Add* > *Add Server Certificate* and create a server certificate.
- 6 Click *Add* > *Add Client Certificate* and create a client certificate. Do not forget to enter an e-mail address.
- 7 Finish with *OK*

To revoke compromised or otherwise unwanted certificates, do the following:

- 1 Start YaST and open the CA module.
- 2 Select the required root CA and click *Enter CA*.
- 3 Enter the password if you are entering a CA for the first time. YaST displays the CA key information in the *Description* tab.
- 4 Click *Certificates* (see Раздел 15.2.3, «Creating or Revoking a Sub-CA» (стр. 204).)
- 5 Select the certificate to revoke and click *Revoke*.
- 6 Choose a reason to revoke this certificate
- 7 Finish with *OK*.

ПРИМЕЧАНИЕ

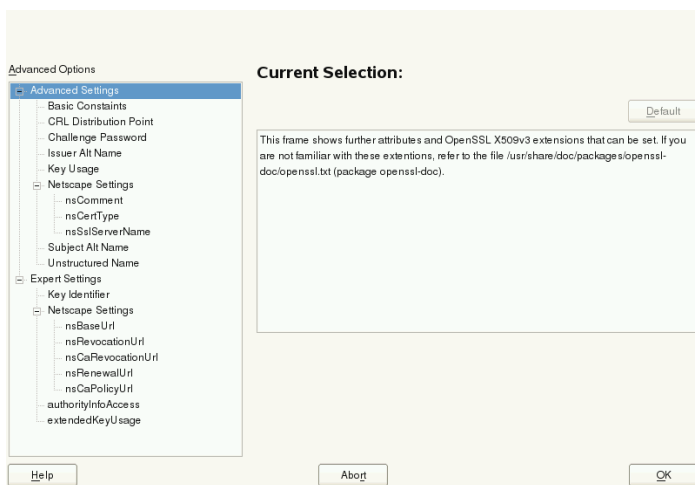
Revocation alone is not enough to deactivate a certificate. Also publish revoked certificates in a CRL. Раздел 15.2.6, «Creating Certificate Revocation Lists (CRLs)» (стр. 209) explains how to create CRLs. Revoked certificates can be completely removed after publication in a CRL with *Delete*.

15.2.5 Changing Default Values

The previous sections explained how to create sub-CAs, client certificates, and server certificates. Special settings are used in the extensions of the X.509 certificate. These settings have been given rational defaults for every certificate type and do not normally need to be changed. However, it may be that you have special requirements for these extensions. In this case, it may make sense to adjust the defaults. Otherwise, start from scratch every time you create a certificate.

- 1 Start YaST and open the CA module.
- 2 Enter the required root CA, as described in Раздел 15.2.3, «Creating or Revoking a Sub-CA» (стр. 204).
- 3 Click *Advanced > Edit Defaults*.
- 4 Choose the type the settings to change. The dialog for changing the defaults, shown in Рисунок 15.4, «YaST CA Module—Extended Settings» (стр. 209), then opens.

Рисунок 15.4 YaST CA Module—Extended Settings



- 5 Change the associated value on the right side and set or delete the critical setting with *critical*.
- 6 Click *Next* to see a short summary.
- 7 Finish your changes with *Save*.

ПРИМЕЧАНИЕ

All changes to the defaults only affect objects created after this point. Already-existing CAs and certificates remain unchanged.

15.2.6 Creating Certificate Revocation Lists (CRLs)

If compromised or otherwise unwanted certificates need to be excluded from further use, they must first be revoked. The procedure for this is explained in Раздел 15.2.3, «Creating or Revoking a Sub-CA» (срп. 204) (for sub-CAs) and Раздел 15.2.4, «Creating or Revoking User Certificates» (срп. 206) (for user certificates). After this, a CRL must be created and published with this information.

The system maintains only one CRL for each CA. To create or update this CRL, do the following:

- 1 Start YaST and open the CA module.
- 2 Enter the required CA, as described in Раздел 15.2.3, «Creating or Revoking a Sub-CA» (стр. 204).
- 3 Click *CRL*. The dialog that opens displays a summary of the last CRL of this CA.
- 4 Create a new CRL with *Generate CRL* if you have revoked new sub-CAs or certificates since its creation.
- 5 Specify the period of validity for the new CRL (default: 30 days).
- 6 Click *OK* to create and display the CRL. Afterwards, you must publish this CRL.

ПРИМЕЧАНИЕ

Applications that evaluate CRLs reject every certificate if the CRL is not available or has expired. As a PKI provider, it is your duty always to create and publish a new CRL before the current CRL expires (period of validity). YaST does not provide a function for automating this procedure.

15.2.7 Exporting CA Objects to LDAP

The executing computer should be configured with the YaST LDAP client for LDAP export. This provides LDAP server information at runtime that can be used when completing dialog fields. Otherwise (although export may be possible), all LDAP data must be entered manually. You must always enter several passwords (see Таблица 15.3, «Passwords during LDAP Export» (стр. 210)).

Таблица 15.3 *Passwords during LDAP Export*

Password	Meaning
LDAP Password	Authorizes the user to make entries in the LDAP tree.

Password	Meaning
Certificate Password	Authorizes the user to export the certificate.
New Certificate Password	The PKCS12 format is used during LDAP export. This format forces the assignment of a new password for the exported certificate.

Certificates, CAs, and CRLs can be exported to LDAP.

Exporting a CA to LDAP

To export a CA, enter the CA as described in Раздел 15.2.3, «Creating or Revoking a Sub-CA» (срп. 204). Select *Extended* > *Export to LDAP* in the subsequent dialog, which opens the dialog for entering LDAP data. If your system has been configured with the YaST LDAP client, the fields are already partly completed. Otherwise, enter all the data manually. Entries are made in LDAP in a separate tree with the attribute «caCertificate».

Exporting a Certificate to LDAP

Enter the CA containing the certificate to export then select *Certificates*. Select the required certificate from the certificate list in the upper part of the dialog and select *Export* > *Export to LDAP*. The LDAP data is entered here in the same way as for CAs. The certificate is saved with the corresponding user object in the LDAP tree with the attributes «userCertificate» (PEM format) and «userPKCS12» (PKCS12 format).

Exporting a CRL to LDAP

Enter the CA containing the CRL to export and select *CRL*. If desired, create a new CRL and click *Export*. The dialog that opens displays the export parameters. You can export the CRL for this CA either once or in periodical time intervals. Activate the export by selecting *Export to LDAP* and enter the respective LDAP data. To do this at regular intervals, select the *Repeated Recreation and Export* radio button and change the interval, if appropriate.

15.2.8 Exporting CA Objects as a File

If you have set up a repository on the computer for administering CAs, you can use this option to create the CA objects directly as a file at the correct location. Different output formats are available, such as PEM, DER, and PKCS12. In the case of PEM, it is also possible to choose whether a certificate should be exported with or without key and whether the key should be encrypted. In the case of PKCS12, it is also possible to export the certification path.

Export a file in the same way for certificates, CAs as with LDAP, described in Раздел 15.2.7, «Exporting CA Objects to LDAP» (стр. 210), except you should select *Export as File* instead of *Export to LDAP*. This then takes you to a dialog for selecting the required output format and entering the password and filename. The certificate is stored at the required location after clicking *OK*.

For CRLs click *Export*, select *Export to file*, choose the export format (PEM or DER) and enter the path. Proceed with *OK* to save it to the respective location.

ПОДСКАЗКА

You can select any storage location in the file system. This option can also be used to save CA objects on a transport medium, such as a USB stick. The `/media` directory generally holds any type of drive except the hard drive of your system.

15.2.9 Importing Common Server Certificates

If you have exported a server certificate with YaST to your media on an isolated CA management computer, you can import this certificate on a server as a *common server certificate*. Do this during installation or at a later point with YaST.

ПРИМЕЧАНИЕ

You need one of the PKCS12 formats to import your certificate successfully.

The general server certificate is stored in `/etc/ssl/servercerts` and can be used there by any CA-supported service. When this certificate expires, it can easily be replaced using the same mechanisms. To get things functioning with the replaced certificate, restart the participating services.

ПОДСКАЗКА

If you select *Import* here, you can select the source in the file system. This option can also be used to import certificates from a transport medium, such as a USB stick.

To import a common server certificate, do the following:

- 1 Start YaST and open *Common Server Certificate* under *Security and Users*
- 2 View the data for the current certificate in the description field after YaST has been started.
- 3 Select *Import* and the certificate file.
- 4 Enter the password and click *Next*. The certificate is imported then displayed in the description field.
- 5 Close YaST with *Finish*.

15.3 For More Information

Detailed information about X.509 certificates, refer to <http://www.ietf.org/html.charters/pkix-charter.html>.

Часть IV. Ограничение привилегий с AppArmor

Introducing AppArmor

Many security vulnerabilities result from bugs in *trusted* programs. A trusted program runs with privileges that attackers would like to have. The program fails to keep that trust if there is a bug in the program that allows the attacker to acquire said privilege.

AppArmor® is an application security solution designed specifically to apply privilege confinement to suspect programs. AppArmor allows the administrator to specify the domain of activities the program can perform by developing a security *profile* for that application (a listing of files that the program may access and the operations the program may perform). AppArmor secures applications by enforcing good application behavior without relying on attack signatures, so it can prevent attacks even if previously unknown vulnerabilities are being exploited.

AppArmor consists of:

- A library of AppArmor profiles for common Linux* applications, describing what files the program needs to access.
- A library of AppArmor profile foundation classes (profile building blocks) needed for common application activities, such as DNS lookup and user authentication.
- A tool suite for developing and enhancing AppArmor profiles, so that you can change the existing profiles to suit your needs and create new profiles for your own local and custom applications.
- Several specially modified applications that are AppArmor enabled to provide enhanced security in the form of unique subprocess confinement (including Apache and Tomcat).

- The AppArmor-loadable kernel module and associated control scripts to enforce AppArmor policies on your system.

16.1 Background Information on AppArmor Profiling

For more information about the science and security of AppArmor, refer to the following papers:

SubDomain: Parsimonious Server Security by Crispin Cowan, Steve Beattie, Greg Kroah-Hartman, Calton Pu, Perry Wagle, and Virgil Gligor

Describes the initial design and implementation of AppArmor. Published in the proceedings of the USENIX LISA Conference, December 2000, New Orleans, LA. This paper is now out of date, describing syntax and features that are different from the current AppArmor product. This paper should be used only for background, and not for technical documentation.

Defcon Capture the Flag: Defending Vulnerable Code from Intense Attack by Crispin Cowan, Seth Arnold, Steve Beattie, Chris Wright, and John Viega

A good guide to strategic and tactical use of AppArmor to solve severe security problems in a very short period of time. Published in the Proceedings of the DARPA Information Survivability Conference and Expo (DISCEX III), April 2003, Washington, DC.

AppArmor for Geeks by Seth Arnold

This document tries to convey a better understanding of the technical details of AppArmor. It is available at http://en.opensuse.org/SDB:AppArmor_geeks.

Getting Started

Prepare a successful deployment of AppArmor on your system by carefully considering the following items:

- 1** Determine the applications to profile. Read more on this in Раздел 17.3, «Choosing the Applications to Profile» (стр. 221).
- 2** Build the needed profiles as roughly outlined in Раздел 17.4, «Building and Modifying Profiles» (стр. 222). Check the results and adjust the profiles when necessary.
- 3** Update your profiles whenever your environment changes or you need to react to security events logged by AppArmor's reporting tool. Refer to Раздел 17.5, «Updating Your Profiles» (стр. 224).

17.1 Installing AppArmor

AppArmor is installed and running on any installation of by default, regardless of what patterns are installed. The packages listed below are needed for a fully-functional instance of AppArmor

- `apparmor-docs`
- `apparmor-parser`
- `apparmor-profiles`
- `apparmor-utils`

- `audit`
- `libapparmor1`
- `perl-libapparmor`
- `yast2-apparmor`

17.2 Enabling and Disabling AppArmor

AppArmor is configured to run by default on any fresh installation of . There are two ways of toggling the status of AppArmor:

Using YaST System Services (Runlevel)

Disable or enable AppArmor by removing or adding its boot script to the sequence of scripts executed on system boot. Status changes are applied on reboot.

Using AppArmor Control Panel

Toggle the status of AppArmor in a running system by switching it off or on using the YaST AppArmor Control Panel. Changes made here are applied instantaneously. The Control Panel triggers a stop or start event for AppArmor and removes or adds its boot script in the system's boot sequence.

To disable AppArmor permanently (by removing it from the sequence of scripts executed on system boot) proceed as follows:

- 1 Start YaST.
- 2 Select *System > System Services (Runlevel)*.
- 3 Select *Expert Mode*.
- 4 Select `boot . apparmor` and click *Set/Reset > Disable the service*.
- 5 Exit the YaST Runlevel tool with *Finish*.

AppArmor will not be initialized on reboot, and stays inactive until you reenable it. Reenabling a service using the YaST Runlevel tool is similar to disabling it.

Toggle the status of AppArmor in a running system by using the AppArmor Control Panel. These changes take effect as soon as you apply them and survive a reboot of the system. To toggle AppArmor's status, proceed as follows:

- 1 Start YaST.
- 2 Select *AppArmor > AppArmor Control Panel*.
- 3 Select *Enable AppArmor*. To disable AppArmor, uncheck this option.
- 4 Exit the AppArmor Control Panel with *Done*.

17.3 Choosing the Applications to Profile

You only need to protect the programs that are exposed to attacks in your particular setup, so only use profiles for those applications you actually run. Use the following list to determine the most likely candidates:

Network Agents
Web Applications
Cron Jobs

To find out which processes are currently running with open network ports and might need a profile to confine them, run `aa-unconfined` as root.

Пример 17.1 *Output of aa-unconfined*

```
19848 /usr/sbin/cupsd not confined
19887 /usr/sbin/sshd not confined
19947 /usr/lib/postfix/master not confined
29205 /usr/sbin/sshd confined by '/usr/sbin/sshd (enforce)'
```

Each of the processes in the above example labeled `not confined` might need a custom profile to confine it. Those labeled `confined` by are already protected by AppArmor.

ПОДСКАЗКА: For More Information

For more information about choosing the the right applications to profile, refer to Раздел 18.2, «Determining Programs to Immunize» (стр. 228).

17.4 Building and Modifying Profiles

AppArmor on ships with a preconfigured set of profiles for the most important applications. In addition, you can use AppArmor to create your own profiles for any application you want.

There are two ways of managing profiles. One is to use the graphical front-end provided by the YaST AppArmor modules and the other is to use the command line tools provided by the AppArmor suite itself. Both methods basically work the same way.

For each application, perform the following steps to create a profile:

- 1 As `root`, let AppArmor create a rough outline of the application's profile by running `aa-genprof programname`

or

Outline the basic profile by running *YaST > AppArmor > Add Profile Wizard* and specifying the complete path to the application you want to profile.

A basic profile is outlined and AppArmor is put into learning mode, which means that it logs any activity of the program you are executing, but does not yet restrict it.

- 2 Run the full range of the application's actions to let AppArmor get a very specific picture of its activities.
- 3 Let AppArmor analyze the log files generated in IIIar 2 (стр. 222) by typing `S` in `aa-genprof`.

or

Analyze the logs by clicking *Scan System Log for AppArmor Events* in the *Add Profile Wizard* and following the instructions given in the wizard until the profile is completed.

AppArmor scans the logs it recorded during the application's run and asks you to set the access rights for each event that was logged. Either set them for each file or use globbing.

- 4 Depending on the complexity of your application, it might be necessary to repeat Шаг 2 (стр. 222) and Шаг 3 (стр. 222). Confine the application, exercise it under the confined conditions, and process any new log events. To properly confine the full range of an application's capabilities, you might be required to repeat this procedure often.
- 5 Once all access permissions are set, your profile is set to enforce mode. The profile is applied and AppArmor restricts the application according to the profile just created.

If you started `aa-genprof` on an application that had an existing profile that was in complain mode, this profile remains in learning mode upon exit of this learning cycle. For more information about changing the mode of a profile, refer to Раздел 22.6.3.2, «`aa-complain`—Entering Complain or Learning Mode» (стр. 287) and Раздел 22.6.3.3, «`aa-enforce`—Entering Enforce Mode» (стр. 288).

Test your profile settings by performing every task you need with the application you just confined. Normally, the confined program runs smoothly and you do not notice AppArmor activities at all. However, if you notice certain misbehavior with your application, check the system logs and see if AppArmor is too tightly confining your application. Depending on the log mechanism used on your system, there are several places to look for AppArmor log entries:

```
/var/log/audit/audit.log
/var/log/messages
dmesg
```

To adjust the profile, analyze the log messages relating to this application again as described in Шаг 3 (стр. 222). Determine the access rights or restrictions when prompted.

ПОДСКАЗКА: For More Information

For more information about profile building and modification, refer to Глава 19, *Profile Components and Syntax* (стр. 235), Глава 21, *Building and Managing Profiles with YaST* (стр. 259), and Глава 22, *Building Profiles from the Command Line* (стр. 279).

17.5 Updating Your Profiles

Software and system configurations change over time. As a result, your profile setup for AppArmor might need some fine-tuning from time to time. AppArmor checks your system log for policy violations or other AppArmor events and lets you adjust your profile set accordingly. Any application behavior that is outside of any profile definition can also be addressed using the *Update Profile Wizard*.

To update your profile set, proceed as follows:

- 1 Start YaST and choose *AppArmor > Update Profile Wizard*.
- 2 Adjust access or execute rights to any resource or for any executable that has been logged when prompted.
- 3 Leave YaST after you have answered all questions. Your changes are applied to the respective profiles.

ПОДСКАЗКА: For More Information

For more information about updating your profiles from the system logs, refer to Раздел 21.5, «Updating Profiles from Log Entries» (стр. 275).

Immunizing Programs

Effective hardening of a computer system requires minimizing the number of programs that mediate privilege, then securing the programs as much as possible. With AppArmor, you only need to profile the programs that are exposed to attack in your environment, which drastically reduces the amount of work required to harden your computer. AppArmor profiles enforce policies to make sure that programs do what they are supposed to do, but nothing else.

AppArmor® provides immunization technologies that protect applications from the inherent vulnerabilities they possess. After installing AppArmor, setting up AppArmor profiles, and rebooting the computer, your system becomes immunized because it begins to enforce the AppArmor security policies. Protecting programs with AppArmor is referred to as *immunizing*.

Administrators need only concern themselves with the applications that are vulnerable to attacks, and generate profiles for these. Hardening a system thus comes down to building and maintaining the AppArmor profile set and monitoring any policy violations or exceptions logged by AppArmor's reporting facility.

Users should not notice AppArmor at all. It runs «behind the scenes» and does not require any user interaction. Performance is not noticeably affected by AppArmor. If some activity of the application is not covered by an AppArmor profile or if some activity of the application is prevented by AppArmor, the administrator needs to adjust the profile of this application to cover this kind of behavior.

AppArmor sets up a collection of default application profiles to protect standard Linux services. To protect other applications, use the AppArmor tools to create profiles for the applications that you want protected. This chapter

introduces the philosophy of immunizing programs. Proceed to Глава 19, *Profile Components and Syntax* (стр. 235), Глава 21, *Building and Managing Profiles with YaST* (стр. 259), or Глава 22, *Building Profiles from the Command Line* (стр. 279) if you are ready to build and manage AppArmor profiles.

AppArmor provides streamlined access control for network services by specifying which files each program is allowed to read, write, and execute, and which type of network it is allowed to access. This ensures that each program does what it is supposed to do, and nothing else. AppArmor quarantines programs to protect the rest of the system from being damaged by a compromised process.

AppArmor is a host intrusion prevention or mandatory access control scheme. Previously, access control schemes were centered around users because they were built for large timeshare systems. Alternatively, modern network servers largely do not permit users to log in, but instead provide a variety of network services for users (such as Web, mail, file, and print servers). AppArmor controls the access given to network services and other programs to prevent weaknesses from being exploited.

ПОДСКАЗКА: Background Information for AppArmor

To get a more in-depth overview of AppArmor and the overall concept behind it, refer to Раздел 16.1, «Background Information on AppArmor Profiling» (стр. 218).

18.1 Introducing the AppArmor Framework

This section provides a very basic understanding of what is happening «behind the scenes» (and under the hood of the YaST interface) when you run AppArmor.

An AppArmor profile is a plain text file containing path entries and access permissions. See Раздел 19.1, «Breaking a AppArmor Profile into Its Parts» (стр. 236) for a detailed reference profile. The directives contained in this text file are then enforced by the AppArmor routines to quarantine the process or program.

The following tools interact in the building and enforcement of AppArmor profiles and policies:

`aa-unconfined / unconfined`

`aa-unconfined` detects any application running on your system that listens for network connections and is not protected by an AppArmor profile. Refer to Раздел 22.6.3.8, «`aa-unconfined`—Identifying Unprotected Processes» (стр. 302) for detailed information about this tool.

`aa-autodep / autodep`

`aa-autodep` creates a basic framework of a profile that needs to be fleshed out before it is put to use in production. The resulting profile is loaded and put into complain mode, reporting any behavior of the application that is not (yet) covered by AppArmor rules. Refer to Раздел 22.6.3.1, «`aa-autodep`—Creating Approximate Profiles» (стр. 286) for detailed information about this tool.

`aa-genprof / genprof`

`aa-genprof` generates a basic profile and asks you to refine this profile by executing the application and generating log events that need to be taken care of by AppArmor policies. You are guided through a series of questions to deal with the log events that have been triggered during the application's execution. After the profile has been generated, it is loaded and put into enforce mode. Refer to Раздел 22.6.3.4, «`aa-genprof`—Generating Profiles» (стр. 289) for detailed information about this tool.

`aa-logprof / logprof`

`aa-logprof` interactively scans and reviews the log entries generated by an application that is confined by an AppArmor profile in complain mode. It assists you in generating new entries in the profile concerned. Refer to Раздел 22.6.3.5, «`aa-logprof`—Scanning the System Log» (стр. 297) for detailed information about this tool.

`aa-complain / complain`

`aa-complain` toggles the mode of an AppArmor profile from enforce to complain. Exceptions to rules set in a profile are logged, but the profile is not enforced. Refer to Раздел 22.6.3.2, «`aa-complain`—Entering Complain or Learning Mode» (стр. 287) for detailed information about this tool.

`aa-enforce / enforce`

`aa-enforce` toggles the mode of an AppArmor profile from complain to enforce. Exceptions to rules set in a profile are logged, but not permitted—the profile is enforced. Refer to Раздел 22.6.3.3, «`aa-enforce`—Entering Enforce Mode» (стр. 288) for detailed information about this tool.

Once a profile has been built and is loaded, there are two ways in which it can get processed:

`aa-complain / complain`

In complain mode, violations of AppArmor profile rules, such as the profiled program accessing files not permitted by the profile, are detected. The violations are permitted, but also logged. To improve the profile, turn complain mode on, run the program through a suite of tests to generate log events that characterize the program's access needs, then postprocess the log with the AppArmor tools (YaST or `aa-logprof`) to transform log events into improved profiles.

`aa-enforce / enforce`

In enforce mode, violations of AppArmor profile rules, such as the profiled program accessing files not permitted by the profile, are detected. The violations are logged and not permitted. The default is for enforce mode to be enabled. To log the violations only, but still permit them, use complain mode. Enforce toggles with complain mode.

18.2 Determining Programs to Immunize

Now that you have familiarized yourself with AppArmor, start selecting the applications for which to build profiles. Programs that need profiling are those that mediate privilege. The following programs have access to resources that the person using the program does not have, so they grant the privilege to the user when used:

cron Jobs

Programs that are run periodically by cron. Such programs read input from a variety of sources and can run with special privileges, sometimes with as much as root privilege. For example, cron can run `/usr/sbin/logrotate` daily to rotate, compress, or even mail system logs. For instructions for finding these types of programs, refer to Раздел 18.3, «Immunizing cron Jobs» (стр. 229).

Web Applications

Programs that can be invoked through a Web browser, including CGI Perl scripts, PHP pages, and more complex Web applications. For instructions for finding these types of programs, refer to Раздел 18.4.1, «Immunizing Web Applications» (стр. 231).

Network Agents

Programs (servers and clients) that have open network ports. User clients, such as mail clients and Web browsers mediate privilege. These programs run with the privilege to write to the user's home directory and they process input from potentially hostile remote sources, such as hostile Web sites and e-mailed malicious code. For instructions for finding these types of programs, refer to Раздел 18.4.2, «Immunizing Network Agents» (стр. 233).

Conversely, unprivileged programs do not need to be profiled. For instance, a shell script might invoke the `cp` program to copy a file. Because `cp` does not have its own profile, it inherits the profile of the parent shell script, so can copy any files that the parent shell script's profile can read and write.

18.3 Immunizing cron Jobs

To find programs that are run by cron, inspect your local cron configuration. Unfortunately, cron configuration is rather complex, so there are numerous files to inspect. Periodic cron jobs are run from these files:

```
/etc/crontab
/etc/cron.d/*
/etc/cron.daily/*
/etc/cron.hourly/*
/etc/cron.monthly/*
/etc/cron.weekly/*
```

For root's cron jobs, edit the tasks with `crontab -e` and list root's cron tasks with `crontab -l`. You must be root for these to work.

Once you find these programs, you can use the *Add Profile Wizard* to create profiles for them. Refer to Раздел 21.1, «Adding a Profile Using the Wizard» (стр. 261).

18.4 Immunizing Network Applications

An automated method for finding network server daemons that should be profiled is to use the `aa-unconfined` tool.

The `aa-unconfined` tool uses the command `netstat -nlp` to inspect your open ports from inside your computer, detect the programs associated with

those ports, and inspect the set of AppArmor profiles that you have loaded. `aa-unconfined` then reports these programs along with the AppArmor profile associated with each program, or reports «none» (if the program is not confined).

ПРИМЕЧАНИЕ

If you create a new profile, you must restart the program that has been profiled to have it be effectively confined by AppArmor.

Below is a sample `aa-unconfined` output:

```
2325 /sbin/portmap not confined
3702❶ /usr/sbin/sshd❷ confined
      by '/usr/sbin/sshd❸ (enforce)'
4040 /usr/sbin/ntpd confined by '/usr/sbin/ntpd (enforce)'
4373 /usr/lib/postfix/master confined by '/usr/lib/postfix/master
(enforce)'
4505 /usr/sbin/httpd2-prefork confined by '/usr/sbin/httpd2-prefork
(enforce)'
5274 /sbin/dhcpd not confined
5592 /usr/bin/ssh not confined
7146 /usr/sbin/cupsd confined by '/usr/sbin/cupsd (complain)'
```

- ❶ The first portion is a number. This number is the process ID number (PID) of the listening program.
- ❷ The second portion is a string that represents the absolute path of the listening program
- ❸ The final portion indicates the profile confining the program, if any.

ПРИМЕЧАНИЕ

`aa-unconfined` requires `root` privileges and should not be run from a shell that is confined by an AppArmor profile.

`aa-unconfined` does not distinguish between one network interface and another, so it reports all unconfined processes, even those that might be listening to an internal LAN interface.

Finding user network client applications is dependent on your user preferences. The `aa-unconfined` tool detects and reports network ports opened by client applications, but only those client applications that are running at the time the `aa-unconfined` analysis is performed. This is a problem because network services tend to be running all the time, while network client applications tend only to be running when the user is interested in them.

Applying AppArmor profiles to user network client applications is also dependent on user preferences. Therefore, we leave the profiling of user network client applications as an exercise for the user.

To aggressively confine desktop applications, the `aa-unconfined` command supports a `paranoid` option, which reports all processes running and the corresponding AppArmor profiles that might or might not be associated with each process. The user can then decide whether each of these programs needs an AppArmor profile.

If you have new or modified profiles, you can submit them to the `apparmor-general@forge.novell.com` [<mailto:apparmor-general@forge.novell.com>] mailing list along with a use case for the application behavior that you exercised. The AppArmor team reviews and may submit the work into . We cannot guarantee that every profile will be included, but we make a sincere effort to include as much as possible so that end users can contribute to the security profiles that ship in .

Alternatively, use the AppArmor profile repository to make your profiles available to other users and to download profiles created by other AppArmor users and the AppArmor developers. Refer to Глава 20, *AppArmor Profile Repositories* (стр. 257) for more information on how to use the AppArmor profile repository.

18.4.1 Immunizing Web Applications

To find Web applications, investigate your Web server configuration. The Apache Web server is highly configurable and Web applications can be stored in many directories, depending on your local configuration. , by default, stores Web applications in `/srv/www/cgi-bin/`. To the maximum extent possible, each Web application should have an AppArmor profile.

Once you find these programs, you can use the AppArmor *Add Profile Wizard* to create profiles for them. Refer to Раздел 21.1, «Adding a Profile Using the Wizard» (стр. 261).

Because CGI programs are executed by the Apache Web server, the profile for Apache itself, `usr.sbin.httpd2-prefork` for Apache2 on , must be modified to add execute permissions to each of these programs. For instance, adding the line `/srv/www/cgi-bin/my_hit_counter.pl rpx` grants Apache permission to execute the Perl script `my_hit_counter.pl` and requires that there be a

dedicated profile for `my_hit_counter.pl`. If `my_hit_counter.pl` does not have a dedicated profile associated with it, the rule should say `/srv/www/cgi-bin/my_hit_counter.pl rix` to cause `my_hit_counter.pl` to inherit the `usr.sbin.httpd2-prefork` profile.

Some users might find it inconvenient to specify execute permission for every CGI script that Apache might invoke. Instead, the administrator can grant controlled access to collections of CGI scripts. For instance, adding the line `/srv/www/cgi-bin/*.{pl,py,pyc} rix` allows Apache to execute all files in `/srv/www/cgi-bin/` ending in `.pl` (Perl scripts) and `.py` or `.pyc` (Python scripts). As above, the `ix` part of the rule causes Python scripts to inherit the Apache profile, which is appropriate if you do not want to write individual profiles for each Python script.

ПРИМЕЧАНИЕ

If you want the subprocess confinement module (`apache2-mod-apparmor`) functionality when Web applications handle Apache modules (`mod_perl` and `mod_php`), use the `ChangeHat` features when you add a profile in YaST or at the command line. To take advantage of the subprocess confinement, refer to Раздел 23.1, «Apache ChangeHat» (стр. 306).

Profiling Web applications that use `mod_perl` and `mod_php` requires slightly different handling. In this case, the «program» is a script interpreted directly by the module within the Apache process, so no `exec` happens. Instead, the AppArmor version of Apache calls `change_hat()` using a subprofile (a «hat») corresponding to the name of the URI requested.

ПРИМЕЧАНИЕ

The name presented for the script to execute might not be the URI, depending on how Apache has been configured for where to look for module scripts. If you have configured your Apache to place scripts in a different place, the different names appear in log file when AppArmor complains about access violations. See Глава 25, *Managing Profiled Applications* (стр. 319).

For `mod_perl` and `mod_php` scripts, this is the name of the Perl script or the PHP page requested. For example, adding this subprofile allows the `localtime.php` page to execute and access the local system time:

```
/usr/bin/httpd2-prefork {  
    # ...
```

```

^/cgi-bin/localtime.php {
    /etc/localtime           r,
    /srv/www/cgi-bin/localtime.php r,
    /usr/lib/locale/**       r,
}
}

```

If no subprofile has been defined, the AppArmor version of Apache applies the `DEFAULT_URI` hat. This subprofile is basically sufficient to display an HTML Web page. The `DEFAULT_URI` hat that AppArmor provides by default is the following:

```

^DEFAULT_URI {
    /usr/sbin/suexec2           mixr,
    /var/log/apache2/**         rwl,
    @{HOME}/public_html        r,
    @{HOME}/public_html/**     r,
    /srv/www/htdocs            r,
    /srv/www/htdocs/**         r,
    /srv/www/icons/*.{gif,jpg,png} r,
    /srv/www/vhosts            r,
    /srv/www/vhosts/**         r,
    /usr/share/apache2/**      r,
    /var/lib/php/sess_*        rwl }

```

To use a single AppArmor profile for all Web pages and CGI scripts served by Apache, a good approach is to edit the `DEFAULT_URI` subprofile.

18.4.2 Immunizing Network Agents

To find network server daemons and network clients (such as fetchmail, Firefox, Amarok or Banshee) that need to be profiled, you should inspect the open ports on your machine, consider the programs that are answering on those ports, and provide profiles for as many of those programs as possible. If you provide profiles for all programs with open network ports, an attacker cannot get to the file system on your machine without passing through a AppArmor profile policy.

Scan your server for open network ports manually from outside the machine using a scanner (such as `nmap`), or from inside the machine using the `netstat --inet -n -p` command. Then, inspect the machine to determine which programs are answering on the discovered open ports.

ПОДСКАЗКА

Refer to the man page of the `netstat` command for a detailed reference of all possible options.

Profile Components and Syntax

Building AppArmor profiles to confine an application is very straightforward and intuitive. AppArmor ships with several tools that assist in profile creation. It does not require you to do any programming or script handling. The only task that is required of the administrator is to determine a policy of strictest access and execute permissions for each application that needs to be hardened.

Updates or modifications to the application profiles are only required if the software configuration or the desired range of activities changes. AppArmor offers intuitive tools to handle profile updates and modifications.

You are ready to build AppArmor profiles after you select the programs to profile. To do so, it is important to understand the components and syntax of profiles. AppArmor profiles contain several building blocks that help build simple and reusable profile code:

`#include` Files

`#include` statements are used to pull in parts of other AppArmor profiles to simplify the structure of new profiles.

Abstractions

Abstractions are `#include` statements grouped by common application tasks.

Program Chunks

Program chunks are `#include` statements that contain chunks of profiles that are specific to program suites.

Capability Entries

Capability entries are profile entries for any of the POSIX.1e Linux capabilities allowing a fine-grained control over what a confined process is allowed to do through system calls that require privileges.

Network Access Control Entries

Network Access Control Entries mediate network access based on the address type and family.

Local Variable Definitions

Local variables define shortcuts for paths.

File Access Control Entries

File Access Control Entries specify the set of files an application can access.

rlimit Entries

rlimit entries set and control an application's resource limits.

For help determining the programs to profile, refer to Раздел 18.2, «Determining Programs to Immunize» (стр. 228). To start building AppArmor profiles with YaST, proceed to Глава 21, *Building and Managing Profiles with YaST* (стр. 259). To build profiles using the AppArmor command line interface, proceed to Глава 22, *Building Profiles from the Command Line* (стр. 279).

19.1 Breaking a AppArmor Profile into Its Parts

The easiest way of explaining what a profile consists of and how to create one is to show the details of a sample profile, in this case for a hypothetical application called `/usr/bin/foo`:

```
#include <tunables/global>❶

# a comment naming the application to confine
/usr/bin/foo❷
{
    #include <abstractions/base>❸❹

    capability setgid❺,
    network inet tcp❻,

    link /etc/sysconfig/foo -> /etc/foo.conf,❼
```

```

/bin/mount                ux,
/dev/{,u}②random          r,
/etc/ld.so.cache          r,
/etc/foo/*                r,
/lib/ld-*.so*             mr,
/lib/lib*.so*            mr,
/proc/[0-9]**            r,
/usr/lib/**              mr,
/tmp/⑨                   r,
/tmp/foo.pid             wr,
/tmp/foo.*               lrw,
/@@{HOME}⑩/.foo_file      rw,
/@@{HOME}⑩/.foo_lock      kw,
owner⑪ /shared/foo/**    rw,
/usr/bin/foobar          cx,⑫
/bin/**                  px -> bin_generic,⑬

# a comment about foo's local (children)profile for /usr/bin/foobar.

profile /usr/bin/foobar⑭ {
    /bin/bash             rmix,
    /bin/cat              rmix,
    /bin/more             rmix,
    /var/log/foobar*      rwl,
    /etc/foobar           r,
}

# foo's hat, bar.
^bar⑮ {
    /lib/ld-*.so*         mr,
    /usr/bin/bar          px,
    /var/spool/*          rwl,
}
}

```

- ❶ This loads a file containing variable definitions.
- ❷ The normalized path to the program that is confined.
- ❸ The curly braces ({ }) serve as a container for include statements, subprofiles, path entries, capability entries, and network entries.
- ❹ This directive pulls in components of AppArmor profiles to simplify profiles.
- ❺ Capability entry statements enable each of the 29 POSIX.1e draft capabilities.
- ❻ A directive determining the kind of network access allowed to the application. For details, refer to Раздел 19.5, «Network Access Control» (стр. 243).
- ❼ A link pair rule specifying the source and the target of a link. See Раздел 19.7.6, «Link Pair» (стр. 248) for more information.
- ❽ The curly braces ({ }) make this rule apply to the path both with and without the content enclosed by the braces.

- ⑨ A path entry specifying what areas of the file system the program can access. The first part of a path entry specifies the absolute path of a file (including regular expression globbing) and the second part indicates permissible access modes (for example `r` for read, `w` for write, and `x` for execute). A whitespace of any kind (spaces or tabs) can precede pathnames or separate the pathname from the access modes. Spaces between the access mode and the trailing comma are optional. Find a comprehensive overview of the available access modes in Раздел 19.7, «File Permission Access Modes» (стр. 246).
- ⑩ This variable expands to a value that can be changed without changing the entire profile.
- ⑪ An owner conditional rule, granting read and write permission on files owned by the user. Refer to Раздел 19.7.7, «Owner Conditional Rules» (стр. 248) for more information.
- ⑫ This entry defines a transition to the local profile `/usr/bin/foobar`. Find a comprehensive overview of the available execute modes in Раздел 19.8, «Execute Modes» (стр. 249).
- ⑬ A named profile transition to the profile `bin_generic` located in the global scope. See Раздел 19.8.7, «Named Profile Transitions» (стр. 252) for details.
- ⑭ The local profile `/usr/bin/foobar` is defined in this section.
- ⑮ This section references a «hat» subprofile of the application. For more details on AppArmor's ChangeHat feature, refer to Глава 23, *Profiling Your Web Applications Using ChangeHat* (стр. 305).

When a profile is created for a program, the program can access only the files, modes, and POSIX capabilities specified in the profile. These restrictions are in addition to the native Linux access controls.

Example: To gain the capability `CAP_CHOWN`, the program must have both access to `CAP_CHOWN` under conventional Linux access controls (typically, be a `root`-owned process) and have the capability `chown` in its profile. Similarly, to be able to write to the file `/foo/bar` the program must have both the correct user ID and mode bits set in the files attributes (see the `chmod` and `chown` man pages) and have `/foo/bar w` in its profile.

Attempts to violate AppArmor rules are recorded in `/var/log/audit/audit.log` if the `audit` package is installed or otherwise in `/var/log/messages`. In many cases, AppArmor rules prevent an attack from working because necessary files are not accessible and, in all cases, AppArmor confinement restricts the damage that the attacker can do to the set of files permitted by AppArmor.

19.2 Profile Types

AppArmor knows four different types of profiles: standard profiles, unattached profiles, local profiles and hats. Standard and unattached profiles are stand-alone profiles, each stored in a file under `/etc/apparmor.d/`. Local profiles and hats are children profiles embedded inside of a parent profile used to provide tighter or alternate confinement for a subtask of an application.

19.2.1 Standard Profiles

The default AppArmor profile is attached to a program by its name, so a profile name must match the path to the application it is to confine.

```
/usr/bin/foo {  
...  
}
```

This profile will be automatically used whenever an unconfined process executes `/usr/bin/foo`.

19.2.2 Unattached Profiles

Unattached profiles do not reside in the file system namespace and therefore are not automatically attached to an application. The name of an unattached profile is preceded by the keyword `profile`. You can freely choose a profile name, except for the following limitations: the name must not begin with a `:` or `.` character. If it contains a whitespace, it must be quoted. If the name begins with a `/`, the profile is considered to be a standard profile, so the following two profiles are identical:

```
profile /usr/bin/foo {  
...  
}  
/usr/bin/foo {  
...  
}
```

Unattached profiles are never used automatically, nor can they be transitioned to through a `px` rule. They need to be attached to a program by either using a named profile transition (see Раздел 19.8.7, «Named Profile Transitions» (стр. 252)) or with the `change_profile` rule (see Раздел 19.2.5, «Change rules» (стр. 240)).

Unattached profiles are useful for specialized profiles for system utilities that generally should not be confined by a system wide profile (for example, `/bin/bash`). They can also be used to set up roles or to confine a user.

19.2.3 Local Profiles

Local profiles provide a convenient way to provide specialized confinement for utility programs launched by a confined application. They are specified just like standard profiles except they are embedded in a parent profile and begin with the `profile` keyword:

```
/parent/profile {  
    ...  
    profile local/profile {  
        ...  
    }  
}
```

To transition to a local profile, either use a `cx` rule (see Раздел 19.8.2, «Discrete Local Profile Execute Mode (cx)» (стр. 250)) or a named profile transition (see Раздел 19.8.7, «Named Profile Transitions» (стр. 252)).

19.2.4 Hats

AppArmor "hats" are a local profiles with some additional restrictions and an implicit rule allowing for `change_hat` to be used to transition to them. Refer to Глава 23, *Profiling Your Web Applications Using ChangeHat* (стр. 305) for a detailed description.

19.2.5 Change rules

AppArmor provides `change_hat` and `change_profile` rules that control domain transitioning. `change_hat` are specified by defining hats in a profile, while `change_profile` rules refer to another profile and start with the keyword `change_profile`:

```
change_profile /usr/bin/foobar,
```

Both `change_hat` and `change_profile` provide for an application directed profile transition, without having to launch a separate application.

`change_profile` provides a generic one way transition between any of the loaded profiles. `change_hat` provides for a returnable parent child transition where an application can switch from the parent profile to the hat profile and if it provides the correct secret key return to the parent profile at a later time.

`change_profile` is best used in situations where an application goes through a trusted setup phase and then can lower its privilege level. Any resources mapped or opened during the start-up phase may still be accessible after the profile change, but the new profile will restrict the opening of new resources, and will even limit some of the resources opened before the switch. Specifically, memory resources will still be available while capability and file resources (as long as they are not memory mapped) can be limited.

`change_hat` is best used in situations where an application runs a virtual machine or an interpreter that does not provide direct access to the applications resources (e.g. Apache's `mod_php`). Since `change_hat` stores the return secret key in the application's memory the phase of reduced privilege should not have direct access to memory. It is also important that file access is properly separated, since the hat can restrict accesses to a file handle but does not close it. If an application does buffering and provides access to the open files with buffering, the accesses to these files may not be seen by the kernel and hence not restricted by the new profile.

ПРЕДУПРЕЖДЕНИЕ: Safety of Domain Transitions

The `change_hat` and `change_profile` domain transitions are less secure than a domain transition done through an `exec` because they do not affect a processes memory mappings, nor do they close resources that have already been opened.

19.3 #include Statements

`#include` statements are directives that pull in components of other AppArmor profiles to simplify profiles. Include files retrieve access permissions for programs. By using an include, you can give the program access to directory paths or files that are also required by other programs. Using includes can reduce the size of a profile.

By default, AppArmor adds `/etc/apparmor.d` to the path in the `#include` statement. AppArmor expects the include files to be located in `/etc/`

`apparmor.d`. Unlike other profile statements (but similar to C programs), `#include` lines do not end with a comma.

To assist you in profiling your applications, AppArmor provides three classes of `#includes`: abstractions, program chunks and tunables.

19.3.1 Abstractions

Abstractions are `#includes` that are grouped by common application tasks. These tasks include access to authentication mechanisms, access to name service routines, common graphics requirements, and system accounting. Files listed in these abstractions are specific to the named task. Programs that require one of these files usually require some of the other files listed in the abstraction file (depending on the local configuration as well as the specific requirements of the program). Find abstractions in `/etc/apparmor.d/abstractions`.

19.3.2 Program Chunks

The program-chunks directory (`/etc/apparmor.d/program-chunks`) contains some chunks of profiles that are specific to program suites and not generally useful outside of the suite, thus are never suggested for use in profiles by the profile wizards (`aa-logprof` and `aa-genprof`). Currently, program chunks are only available for the postfix program suite.

19.3.3 Tunables

The tunables directory (`/etc/apparmor.d/tunables`) contains global variable definitions. When used in a profile, these variables expand to a value that can be changed without changing the entire profile. Add all the tunables definitions that should be available to every profile to `/etc/apparmor.d/tunables/global`.

19.4 Capability Entries (POSIX.1e)

Capability statements are simply the word `capability` followed by the name of the POSIX.1e capability as defined in the `capabilities(7)` man page.

19.5 Network Access Control

AppArmor allows mediation of network access based on the address type and family. The following illustrates the network access rule syntax:

```
network [[<domain>❶] [<type>❷] [<protocol>❸]]
```

- ❶ Supported domains: inet, ax25, ipx, appletalk, netrom, bridge, x25, inet6, rose, netbeui, security, key, packet, ash, econet, atmshvc, sna, irda, pppox, wanpipe, bluetooth
- ❷ Supported types: stream, dgram, seqpacket, rdm, raw, packet
- ❸ Supported protocols: tcp, udp, icmp

The AppArmor tools support only family and type specification. The AppArmor module emits only `network domain type` in «access denied» messages. And only these are output by the profile generation tools, both YaST and command line.

The following examples illustrate possible network-related rules to be used in AppArmor profiles. Note that the syntax of the last two are not currently supported by the AppArmor tools.

```
network❶,  
network inet❷,  
network inet6❸,  
network inet stream❹,  
network inet tcp❺,  
network tcp❻,
```

- ❶ Allow all networking. No restrictions applied with regards to domain, type, or protocol.
- ❷ Allow general use of IPv4 networking.
- ❸ Allow general use of IPv6 networking.
- ❹ Allow the use of IPv4 TCP networking.
- ❺ Allow the use of IPv4 TCP networking, paraphrasing the rule above.
- ❻ Allow the use of both IPv4 and IPv6 TCP networking.

19.6 Paths and Globbing

AppArmor explicitly distinguishes directory path names from file path names. Use a trailing `/` for any directory path that needs to be explicitly distinguished:

```
/some/random/example/* r
```

Allow read access to files in the `/some/random/example` directory.

```
/some/random/example/ r
```

Allow read access to the directory only.

```
/some/**/ r
```

Give read access to any directories below `/some`.

```
/some/random/example/** r
```

Give read access to files and directories under `/some/random/example`.

```
/some/random/example/**[^/] r
```

Give read access to files under `/some/random/example`. Explicitly exclude directories (`[^/]`).

Globbing (or regular expression matching) is when you modify the directory path using wild cards to include a group of files or subdirectories. File resources can be specified with a globbing syntax similar to that used by popular shells, such as `csh`, `Bash`, and `zsh`.

<code>*</code>	<p>Substitutes for any number of any characters, except <code>/</code>.</p> <p>Example: An arbitrary number of file path elements.</p>
<code>**</code>	<p>Substitutes for any number of characters, including <code>/</code>.</p> <p>Example: An arbitrary number of path elements, including entire directories.</p>
<code>?</code>	<p>Substitutes for any single character, except <code>/</code>.</p>
<code>[abc]</code>	<p>Substitutes for the single character <code>a</code>, <code>b</code>, or <code>c</code>.</p> <p>Example: a rule that matches <code>/home[01]/*/.plan</code> allows <code>a</code></p>

	program to access <code>.plan</code> files for users in both <code>/home0</code> and <code>/home1</code> .
<code>[a-c]</code>	Substitutes for the single character <code>a</code> , <code>b</code> , or <code>c</code> .
<code>{ab, cd}</code>	Expands to one rule to match <code>ab</code> and one rule to match <code>cd</code> . Example: a rule that matches <code>{usr, www}/pages/**</code> grants access to Web pages in both <code>/usr/pages</code> and <code>/www/pages</code> .
<code>[^a]</code>	Substitutes for any character except <code>a</code> .

19.6.1 Using Variables in Profiles

AppArmor allows to use variables holding paths in profiles. Use global variables to make your profiles portable and local variables to create shortcuts for paths.

A typical example of when global variables come in handy are network scenarios in which user home directories are mounted in different locations. Instead of rewriting paths to home directories in all affected profiles, you only need to change the value of a variable. Global variables are defined under `/etc/apparmor.d/tunables` and have to be made available via an `#include` statement. Find the variable definitions for this use case (`@{HOME}` and `@{HOMEDIRS}`) in the `/etc/apparmor.d/tunables/home` file.

Local variables are defined at the head of a profile. This is useful to provide the base of for a chrooted path, for example:

```
@{CHROOT_BASE}=/tmp/foo
/sbin/syslog-ng {
...
# chrooted applications
@{CHROOT_BASE}/var/lib/*/dev/log w,
@{CHROOT_BASE}/var/log/** w,
...
}
```

ПРИМЕЧАНИЕ

With the current AppArmor tools, variables can only be used when manually editing and maintaining a profile.

19.6.2 Alias rules

Alias rules provide an alternative way to manipulate profile path mappings to site specific layouts. They are an alternative form of path rewriting to using variables, and are done post variable resolution:

```
alias /home/ -> /mnt/users/
```

ПРИМЕЧАНИЕ

With the current AppArmor tools, alias rules can only be used when manually editing and maintaining a profile. Whats more, they are deactivated by disabled. Enable alias rules by editing `/etc/apparmor.d/tunables/alias`

19.7 File Permission Access Modes

File permission access modes consist of combinations of the following modes:

r	Read mode
w	Write mode (mutually exclusive to a)
a	Append mode (mutually exclusive to w)
k	File locking mode
l	Link mode
link file -> target	Link pair rule (cannot be combined with other access modes)

19.7.1 Read Mode (r)

Allows the program to have read access to the resource. Read access is required for shell scripts and other interpreted content and determines if an executing process can core dump.

19.7.2 Write Mode (w)

Allows the program to have write access to the resource. Files must have this permission if they are to be unlinked (removed).

19.7.3 Append Mode (a)

Allows a program to write to the end of a file. In contrast to the `w` mode, the append mode does not include the ability to overwrite data, to rename, or to remove a file. The append permission is typically used with applications who need to be able to write to log files, but which should not be able to manipulate any existing data in the log files. As the append permission is just a subset of the permissions associated with the write mode, the `w` and `a` permission flags cannot be used together and are mutually exclusive.

19.7.4 File Locking Mode (k)

The application can take file locks. Former versions of AppArmor allowed files to be locked if an application had access to them. By using a separate file locking mode, AppArmor makes sure locking is restricted only to those files which need file locking and tightens security as locking can be used in several denial of service attack scenarios.

19.7.5 Link Mode (l)

The link mode mediates access to hard links. When a link is created, the target file must have the same access permissions as the link created (with the exception that the destination does not need link access).

19.7.6 Link Pair

The link mode grants permission to create links to arbitrary files, provided the link has a subset of the permissions granted by the target (subset permission test). By specifying origin and destination, the link pair rule provides greater control over how hard links are created. Link pair rules by default do not enforce the link subset permission test that the standard rules link permission requires. To force the rule to require the test the `subset` keyword is used. The following rules are equivalent:

```
/link    l,  
link subset /link -> /**,
```

ПРИМЕЧАНИЕ

Currently link pair rules are not supported by YaST and the command line tools. Manually edit your profiles to use them. Updating such profiles using the tools is safe, because the link pair entries will not be touched.

19.7.7 Owner Conditional Rules

The file rules can be extended so that they can be conditional upon the the user being the owner of the file (the `fsuid` has to match the file's `uid`). For this purpose the `owner` keyword is prepended to the rule. Owner conditional rules accumulate just as regular file rules.

```
owner /home/**/* rw
```

When using file ownership conditions with link rules the ownership test is done against the target file so the user must own the file to be able to link to it.

ПРИМЕЧАНИЕ: Precedence of Regular File Rules

Owner conditional rules are considered a subset of regular file rules. If a regular file rule overlaps with an owner conditional file rule, the resultant permissions will be that of the regular file rule.

19.7.8 Deny Rules

Deny rules can be used to annotate or quiet known rejects. The profile generating tools will not ask about a known reject treated with a deny rule. Such a reject will

also not show up in the audit logs when denied, keeping the log files lean. If this is not desired, prepend the deny entry with the keyword `audit`.

It is also possible to use deny rules in combination with allow rules. This allows you to specify a broad allow rule, and then subtract a few known files that should not be allowed. Deny rules can also be combined with owner rules, to deny files owned by the user. The following example allows read/write access to everything in a users directory except write access to the `.ssh/` files:

```
deny /home/*/ .ssh/** w,  
/home/*/** rw,
```

The extensive use of deny rules is generally not encouraged, because it makes it much harder to understand what a profile does. However a judicious use of deny rules can simplify profiles. Therefore the tools only generate profiles denying specific files and will not make use of globbing in deny rules. Manually edit your profiles to add deny rules using globbing. Updating such profiles using the tools is safe, because the deny entries will not be touched.

19.8 Execute Modes

Execute modes, also named profile transitions, consist of the following modes:

px	Discrete profile execute mode
cx	Discrete local profile execute mode
ux	Unconstrained execute mode
ix	Inherit execute mode
m	Allow <code>PROT_EXEC</code> with <code>mmap (2)</code> calls

19.8.1 Discrete Profile Execute Mode (px)

This mode requires that a discrete security profile is defined for a resource executed at an AppArmor domain transition. If there is no profile defined, the access is denied.

ПРЕДУПРЕЖДЕНИЕ: Using the Discrete Profile Execute Mode

`px` does not scrub the environment of variables such as `LD_PRELOAD`. As a result, the calling domain may have an undue amount of influence over the called item.

Incompatible with `Ux`, `ux`, `Px`, and `ix`.

19.8.2 Discrete Local Profile Execute Mode (`cx`)

As `px`, but instead of searching the global profile set, `cx` only searches the local profiles of the current profile. This profile transition provides a way for an application to have alternate profiles for helper applications.

ПРИМЕЧАНИЕ: Limitations of the Discrete Local Profile Execute Mode (`cx`)

Currently, `cx` transitions are limited to top level profiles and can not be used in hats and children profiles. This restriction will be removed in the future.

Incompatible with `Ux`, `ux`, `Px`, `px`, `Cx`, and `ix`.

19.8.3 Unconstrained Execute Mode (`ux`)

Allows the program to execute the resource without any AppArmor profile applied to the executed resource. This mode is useful when a confined program needs to be able to perform a privileged operation, such as rebooting the machine. By placing the privileged section in another executable and granting unconstrained execution rights, it is possible to bypass the mandatory constraints imposed on all confined processes. For more information about what is constrained, see the `apparmor(7)` man page.

ПРЕДУПРЕЖДЕНИЕ: Using Unconstrained Execute Mode (`ux`)

Use `ux` only in very special cases. It enables the designated child processes to be run without any AppArmor protection. `ux` does not scrub the environment of variables such as `LD_PRELOAD`. As a result, the calling

domain may have an undue amount of influence over the called resource. Use this mode only if the child absolutely must be run unconfined and `LD_PRELOAD` must be used. Any profile using this mode provides negligible security. Use at your own risk.

This mode is incompatible with `Ux`, `px`, `Px`, and `ix`.

19.8.4 Clean Exec modes

The clean exec modes allow the named program to run in `px`, `cx` and `ux` mode, but AppArmor invokes the Linux kernel's `unsafe_exec` routines to scrub the environment, similar to `setuid` programs. The clean exec modes are specified with an uppercase letter: `Px`, `Cx` and `Ux`. See the man page of `ld.so(8)` for some information about `setuid` and `setgid` environment scrubbing.

19.8.5 Inherit Execute Mode (ix)

`ix` prevents the normal AppArmor domain transition on `execve(2)` when the profiled program executes the named program. Instead, the executed resource inherits the current profile.

This mode is useful when a confined program needs to call another confined program without gaining the permissions of the target's profile or losing the permissions of the current profile. There is no version to scrub the environment because `ix` executions do not change privileges.

Incompatible with `cx`, `ux`, and `px`. Implies `m`.

19.8.6 Allow Executable Mapping (m)

This mode allows a file to be mapped into memory using `mmap(2)`'s `PROT_EXEC` flag. This flag marks the pages executable. It is used on some architectures to provide non executable data pages, which can complicate exploit attempts. AppArmor uses this mode to limit which files a well-behaved program (or all programs on architectures that enforce non executable memory access controls) may use as libraries, to limit the effect of invalid `-L` flags given to `ld(1)` and `LD_PRELOAD`, `LD_LIBRARY_PATH`, given to `ld.so(8)`.

19.8.7 Named Profile Transitions

By default, the `px` and `cx` (and their `clean exec` variants, too) transition to a profile whose name matches the executable name. With named profile transitions, you can specify a profile to be transitioned to. This is useful if multiple binaries need to share a single profile, or if they need to use a different profile than their name would specify. Named profile transitions can be used in conjunction with `cx`, `Cx`, `px` and `Px`. Currently there is a limit of twelve named profile transitions per profile.

Named profile transitions use `->` to indicate the name of the profile that needs to be transitioned to:

```
/usr/bin/foo
{
  /bin/** px -> shared_profile,
  ...
  /usr/*bash cx -> local_profile,
  ...
  profile local_profile
  {
    ...
  }
}
```

ПРИМЕЧАНИЕ: Difference Between Normal and Named Transitions

When used with globbing, normal transitions provide a «one to many» relationship—`/bin/** px` will transition to `/bin/ping`, `/bin/cat`, etc, depending on the program being run.

Named transitions provide a «many to one» relationship—all programs that match the rule regardless of their name will transition to the specified profile.

Named profile transitions show up in the log as having the mode `Nx`. The name of the profile to be changed to is listed in the `name2` field.

19.8.8 Inheritance Fallback for Profile Transitions

The `px` and `cx` transitions specify a hard dependency (if the specified profile does not exist, the `exec` will fail). With the inheritance fallback, the execution will succeed but inherit the current profile. To specify inheritance fallback, `ix` is combined with

`cx`, `Cx`, `px` and `Px` into the modes `cix`, `Cix`, `pix` and `Pix`. The fallback modes can be used with named profile transitions, too.

19.8.9 Variable Settings in Execution Modes

When choosing one of the `Px`, `Cx` or `Ux` execution modes, take into account that the following environment variables are removed from the environment before the child process inherits it. As a consequence, applications or processes relying on any of these variables do not work anymore if the profile applied to them carries `Px`, `Cx` or `Ux` flags:

- `GCONV_PATH`
- `GETCONF_DIR`
- `HOSTALIASES`
- `LD_AUDIT`
- `LD_DEBUG`
- `LD_DEBUG_OUTPUT`
- `LD_DYNAMIC_WEAK`
- `LD_LIBRARY_PATH`
- `LD_ORIGIN_PATH`
- `LD_PRELOAD`
- `LD_PROFILE`
- `LD_SHOW_AUXV`
- `LD_USE_LOAD_BIAS`
- `LOCALDOMAIN`
- `LOCPATH`

- MALLOC_TRACE
- NLSPATH
- RESOLV_HOST_CONF
- RES_OPTIONS
- TMPDIR
- TZDIR

19.9 Resource Limit Control

AppArmor provides the ability to set and control an application's resource limits (rlimits, also known as ulimits). By default AppArmor does not control applications rlimits, and it will only control those limits specified in the confining profile. For more information about resource limits, refer to the `setrlimit(2)`, `ulimit(1)`, or `ulimit(3)` man pages.

AppArmor leverages the system's rlimits and as such does not provide an additional auditing that would normally occur. It also cannot raise rlimits set by the system, AppArmor rlimits can only reduce an application's current resource limits.

The values will be inherited by the children of a process and will remain even if a new profile is transitioned to or the application becomes unconfined. So when an application transitions to a new profile, that profile has the ability to further reduce the applications rlimits.

AppArmor's rlimit rules will also provide mediation of setting an application's hard limits, should it try to raise them. The application will not be able to raise its hard limits any further than specified in the profile. The mediation of raising hard limits is not inherited as the set value is, so that once the application transitions to a new profile it is free to raise its limits as specified in the profile.

AppArmor's rlimit control does not affect an application's soft limits beyond ensuring that they are less than or equal to the application's hard limits.

AppArmor's hard limit rules have the general form of:

```
set rlimit resource <= value,
```

where *resource* and *value* are to be replaced with the following values:

`cpu`

currently not supported

`fsize, data, stack, core, rss, as, memlock, msgqueue`

a number in bytes, or a number with a suffix where the suffix can be K (kilobytes), M (megabytes), G (gigabytes), for example

`rlimit data <= 100M,`

`fsize, nofile, locks, sigpending, nproc*, rtprio`

a number greater or equal to 0

`nice`

a value between -20 and 19

^{*}The `nproc` rlimit is handled different than all the other rlimits. Instead of indicating the standard process rlimit it controls the maximum number of processes that can be running under the profile at any given time. Once the limit is exceeded the creation of new processes under the profile will fail until the number of currently running processes is reduced.

ПРИМЕЧАНИЕ

Currently the tools can not be used to add rlimit rules to profiles. The only way to add rlimit controls to a profile is to manually edit the profile with a text editor. The tools will still work with profiles containing rlimit rules and will not remove them, so it is safe to use the tools to update profiles containing them.

19.10 Auditing Rules

AppArmor provides the ability to audit given rules so that when they are matched an audit message will appear in the audit log. To enable audit messages for a given rule, the `audit` keyword is prepended to the rule:

```
audit /etc/foo/*          rw,
```

If it is desirable to audit only a given permission the rule can be split into two rules. The following example will result in audit messages when files are opened for writing, but not when they are opened for just reading:

```
audit /etc/foo/* w,  
/etc/foo/* r,
```

ПРИМЕЧАНИЕ

Audit messages are not generated for every read or write of a file but only when a file is opened for read or write.

Audit control can be combined with owner conditional file rules to provide auditing when users access files they own (at the moment it is not possible to audit files they don't own):

```
audit owner /home/*/.ssh/** rw,
```

AppArmor Profile Repositories

AppArmor ships a set of profiles enabled by default and created by the AppArmor developers, and kept under the `/etc/apparmor.d`. In addition to these profiles, ships profiles for individual applications together with the relevant application. These profiles are not enabled by default, and reside under another directory than the standard AppArmor profiles, `/etc/apparmor/profiles/extras`.

20.1 Using the Local Repository

The AppArmor tools (YaST and `aa-genprof` and `aa-logprof`) support the use of a local repository. Whenever you start to create a new profile from scratch, and there already is one inactive profile in your local repository, you are asked whether you would like to use the existing inactive one from `/etc/apparmor/profiles/extras` and whether you want to base your efforts on it. If you decide to use this profile, it gets copied over to the directory of profiles enabled by default (`/etc/apparmor.d`) and loaded whenever AppArmor is started. Any further further adjustments will be done to the active profile under `/etc/apparmor.d`.

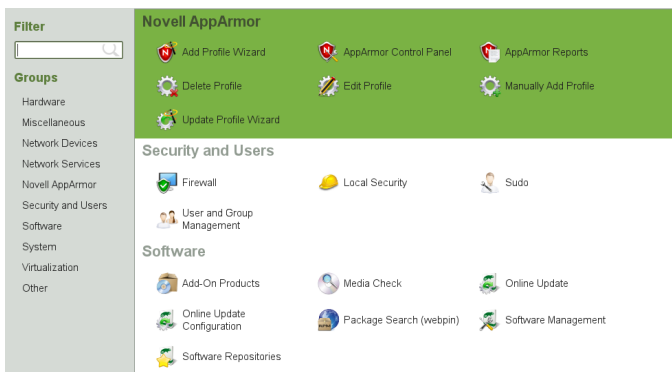
Building and Managing Profiles with YaST

21

YaST provides an easy way to build profiles and manage AppArmor®. It provides two interfaces: a graphical one and a text-based one. The text-based interface consumes less resources and bandwidth, making it a better choice for remote administration, or for times when a local graphical environment is inconvenient. Although the interfaces have differing appearances, they offer the same functionality in similar ways. Another alternative is to use AppArmor commands, which can control AppArmor from a terminal window or through remote connections. The command line tools are described in Глава 22, *Building Profiles from the Command Line* (стр. 279).

Start YaST from the main menu and enter your `root` password when prompted for it. Alternatively, start YaST by opening a terminal window, logging in as `root`, and entering `yast2` for the graphical mode or `yast` for the text-based mode.

Рисунок 21.1 *YaST Controls for AppArmor*



The right frame shows the AppArmor options:

Add Profile Wizard

For detailed steps, refer to Раздел 21.1, «Adding a Profile Using the Wizard» (стр. 261).

Manually Add Profile

Add a AppArmor profile for an application on your system without the help of the wizard. For detailed steps, refer to Раздел 21.2, «Manually Adding a Profile» (стр. 268).

Edit Profile

Edits an existing AppArmor profile on your system. For detailed steps, refer to Раздел 21.3, «Editing Profiles» (стр. 269).

Delete Profile

Deletes an existing AppArmor profile from your system. For detailed steps, refer to Раздел 21.4, «Deleting a Profile» (стр. 274).

Update Profile Wizard

For detailed steps, refer to Раздел 21.5, «Updating Profiles from Log Entries» (стр. 275).

AppArmor Control Panel

For detailed steps, refer to Раздел 21.6, «Managing AppArmor» (стр. 275).

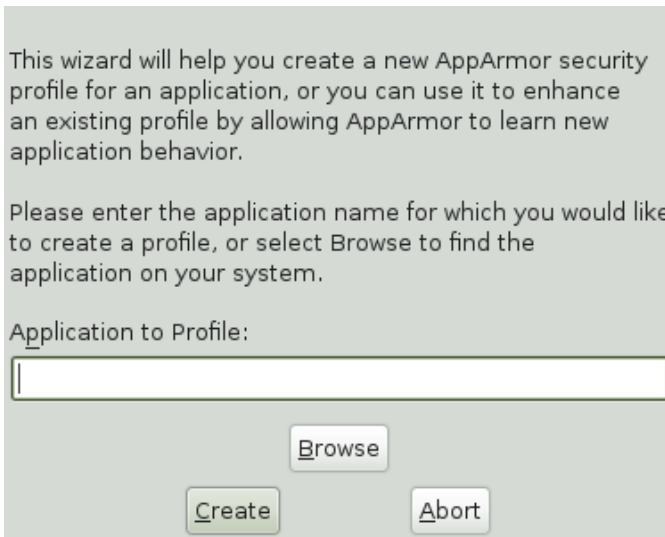
21.1 Adding a Profile Using the Wizard

Add Profile Wizard is designed to set up AppArmor profiles using the AppArmor profiling tools, `aa-genprof` (generate profile) and `aa-logprof` (update profiles from learning mode log file). For more information about these tools, refer to Раздел 22.6.3, «Summary of Profiling Tools» (стр. 285).

- 1 Stop the application before profiling it to ensure that application start-up is included in the profile. To do this, make sure that the application or daemon is not running.

For example, enter `rcPROGRAM stop` (or `/etc/init.d/PROGRAM stop`) in a terminal window while logged in as `root`, replacing `PROGRAM` with the name of the program to profile.

- 2 Start YaST and select *AppArmor > Add Profile Wizard*.



This wizard will help you create a new AppArmor security profile for an application, or you can use it to enhance an existing profile by allowing AppArmor to learn new application behavior.

Please enter the application name for which you would like to create a profile, or select Browse to find the application on your system.

Application to Profile:

- 3 Enter the name of the application or browse to the location of the program.
- 4 Click *Create*. This runs an AppArmor tool named `aa-autodep`, which performs a static analysis of the program to profile and loads an approximate profile into

the AppArmor module. For more information about aa-autodep, refer to Раздел 22.6.3.1, «aa-autodep—Creating Approximate Profiles» (стр. 286).

Depending on whether the profile you are about to create already exists either in the local profile repository (see Раздел 20.1, «Using the Local Repository» (стр. 257)) or in the external profile repository (see Глава 20, *AppArmor Profile Repositories* (стр. 257)) or whether it does not exist yet, proceed with one of the following options:

- Determine whether you want to use or fine-tune an already existing profile from your local profile repository, as outlined in Шаг 5 (стр. 262).
- Determine whether you want to use or fine-tune an already existing profile from the external profile repository, as outlined in Шаг 6 (стр. 262).
- Create the profile from scratch and proceed with Шаг 7 (стр. 262) and beyond.

- 5** If the profile already exists in the local profile repository under `/etc/apparmor/profiles/extra`, YaST informs you that there is an inactive profile which you can either use as a base for your own efforts or which you can just accept as is.

Alternatively, you can choose not to use the local version at all and start creating the profile from scratch. In any case, proceed with Шаг 7 (стр. 262).

- 6** If the profile already exists in the external profile repository and this is the first time you tried to create a profile that already exists in the repository, configure your access to the server and determine how to use it:

6a Determine whether you want to enable access to the external repository or postpone this decision. In case you have selected *Enable Repository*, determine the access mode (download/upload) in a next step. In case you want to postpone the decision, select *Ask Me Later* and proceed directly to Шаг 7 (стр. 262).

6b Provide username and password for your account on the profile repository server and register at the server.

6c Select the profile to use and proceed to Шаг 7 (стр. 262).

- 7** Run the application to profile.

- 8 Perform as many of the application functions as possible, so that learning mode can log the files and directories to which the program requires access to function properly. Be sure to include restarting and stopping the program in the exercised functions. AppArmor needs to handle these events, as well as any other program function.
- 9 Click *Scan system log for AppArmor events* to parse the learning mode log files. This generates a series of questions that you must answer to guide the wizard in generating the security profile.

If requests to add hats appear, proceed to Глава 23, *Profiling Your Web Applications Using ChangeHat* (стр. 305).

The questions fall into two categories:

- A resource is requested by a profiled program that is not in the profile (see Рисунок 21.2, «Learning Mode Exception: Controlling Access to Specific Resources» (стр. 264)). Allow or deny access to a specific resource.
- A program is executed by the profiled program and the security domain transition has not been defined (see Рисунок 21.3, «Learning Mode Exception: Defining Execute Permissions for an Entry» (стр. 265)). Define execute permissions for an entry.

Each of these cases results in a series of questions that you must answer to add the resource to the profile or to add the program to the profile. For an example of each case, see Рисунок 21.2, «Learning Mode Exception: Controlling Access to Specific Resources» (стр. 264) and Рисунок 21.3, «Learning Mode Exception: Defining Execute Permissions for an Entry» (стр. 265). Subsequent steps describe your options in answering these questions.

ПРИМЕЧАНИЕ: Varying Processing Options

Depending on the type of entry processed, the available options vary.

Рисунок 21.2 *Learning Mode Exception: Controlling Access to Specific Resources*

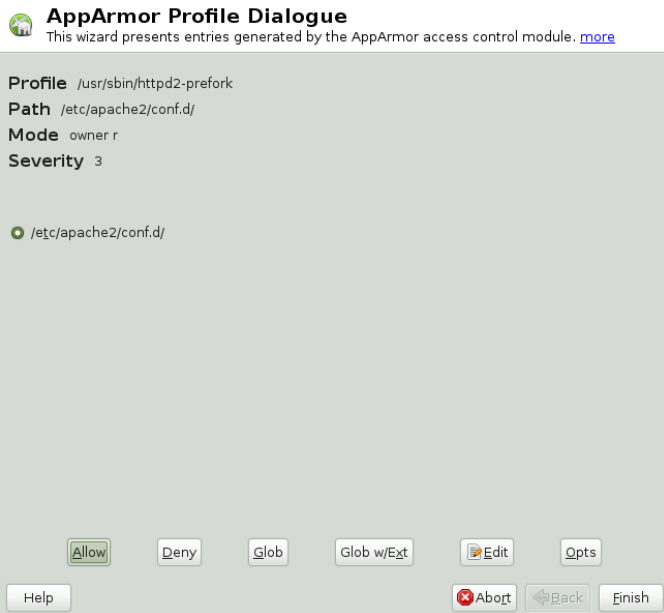


Рисунок 21.3 *Learning Mode Exception: Defining Execute Permissions for an Entry*



- 10 The *Add Profile Wizard* begins suggesting directory path entries that have been accessed by the application profiled (as seen in Рисунок 21.2, «Learning Mode Exception: Controlling Access to Specific Resources» (стр. 264)) or requires you to define execute permissions for entries (as seen in Рисунок 21.3, «Learning Mode Exception: Defining Execute Permissions for an Entry» (стр. 265)).
- For Рисунок 21.2: Learning Mode Exception: Controlling Access to Specific Resources: Select the option that satisfies the request for access, which could be a suggested include, a particular globbed version of the path, or the actual pathname. Depending on the situation, these options are available:

`#include`

The section of a AppArmor profile that refers to an include file. Include files give access permissions for programs. By using an include, you can give the program access to directory paths or files that are also required

by other programs. Using includes can reduce the size of a profile. It is good practice to select includes when suggested.

Globbered Version

Accessed by clicking *Glob*. For information about globbing syntax, refer to Раздел 19.6, «Paths and Globbing» (стр. 243).

Actual Pathname

Literal path that the program needs to access to run properly.

After selecting a directory path, process it as an entry to the AppArmor profile by clicking *Allow* or *Deny*. If you are not satisfied with the directory path entry as it is displayed, you can also *Glob* or *Edit* it.

The following options are available to process the learning mode entries and build the profile:

Allow

Grant the program access to the specified directory path entries. The *Add Profile Wizard* suggests file permission access. For more information about this, refer to Раздел 19.7, «File Permission Access Modes» (стр. 246).

Deny

Click *Deny* to prevent the program from accessing the specified paths.

Glob

Clicking this modifies the directory path (using wild cards) to include all files in the suggested directory. Double-clicking it grants access to all files and subdirectories beneath the one shown. For more information about globbing syntax, refer to Раздел 19.6, «Paths and Globbing» (стр. 243).

Glob w/Ext

Modify the original directory path while retaining the filename extension. A single click causes `/etc/apache2/file.ext` to become `/etc/apache2/*.ext`, adding the wild card (asterisk) in place of the filename. This allows the program to access all files in the suggested directories that end with the `.ext` extension. When you double-click it, access is granted to all files with the particular extension and subdirectories beneath the one shown.

Edit

Edit the highlighted line. The new edited line appears at the bottom of the list.

Abort

Abort aa-logprof, losing all rule changes entered so far and leaving all profiles unmodified.

Finish

Close aa-logprof, saving all rule changes entered so far and modifying all profiles.

Click *Allow* or *Deny* for each learning mode entry. These help build the AppArmor profile.

ПРИМЕЧАНИЕ

The number of learning mode entries corresponds to the complexity of the application.

- For Рисунок 21.3: Learning Mode Exception: Defining Execute Permissions for an Entry: From the following options, select the one that satisfies the request for access. For detailed information about the options available, refer to Раздел 19.7, «File Permission Access Modes» (стр. 246).

Inherit

Stay in the same security profile (parent's profile).

Profile

Require a separate profile to exist for the executed program. When selecting this option, also select whether AppArmor should sanitize the environment when switching profiles by removing certain environment variables that can modify the execution behavior of the child process. Unless these variables are absolutely required to properly execute the child process, always choose the more secure, sanitized option.

Unconfined

Execute the program without a security profile. When prompted, have AppArmor sanitize the environment to avoid adding security risks by inheriting certain environmental variables from the parent process.

ПРЕДУПРЕЖДЕНИЕ: Risks of Running Unconfined

Unless absolutely necessary, do not run unconfined. Choosing the *Unconfined* option executes the new program without any protection from AppArmor.

Deny

Click *Deny* to prevent the program from accessing the specified paths.

Abort

Abort aa-logprof, losing all rule changes entered so far, and leaving all profiles unmodified.

Finish

Close aa-logprof, saving all rule changes entered so far, and modifying all profiles.

- 11 Repeat the previous steps if you need to execute more functionality of the application.

When you are done, click *Finish*. Choose to apply your changes to the local profile set. If you have previously chosen to upload your profile to the external profile repository, provide a brief change log entry describing your work and upload the profile. If you had postponed the decision on whether to upload the profile or not, YaST asks you again and you can create an account the upload the profile now or not upload it at all.

As soon as you exit the *Profile Creation Wizard*, the profile is saved both locally and on the repository server, if you have chosen to upload it. The profile is then loaded into the AppArmor module.

21.2 Manually Adding a Profile

AppArmor enables you to create a AppArmor profile by manually adding entries into the profile. Select the application for which to create a profile then add entries.

- 1 Start YaST and select *AppArmor > Manually Add Profile*.
- 2 Browse your system to find the application for which to create a profile.

- 3 When you find the application, select it and click *Open*. A basic, empty profile appears in the *AppArmor Profile Dialog* window.
- 4 In *AppArmor Profile Dialog*, add, edit, or delete AppArmor profile entries by clicking the corresponding buttons and referring to Раздел 21.3.1, «Adding an Entry» (стр. 271), Раздел 21.3.2, «Editing an Entry» (стр. 274), or Раздел 21.3.3, «Deleting an Entry» (стр. 274).
- 5 When finished, click *Done*.

21.3 Editing Profiles

AppArmor enables you to edit AppArmor profiles manually by adding, editing, or deleting entries. To edit a profile, proceed as follows:

- 1 Start YaST and select *AppArmor > Edit Profile*.



Edit Profile - Choose profile to edit

Please make a selection from the listed profiles and press Next to edit the profile.

Profile Name:

```
/bin/ping
/sbin/klogd
/sbin/syslog-ng
/sbin/syslogd
/usr/sbin/avahi-daemon
/usr/sbin/httpd2-prefork
/usr/sbin/identd
/usr/sbin/mdnsd
/usr/sbin/nscd
/usr/sbin/ntpd
/usr/sbin/traceroute
```

Help

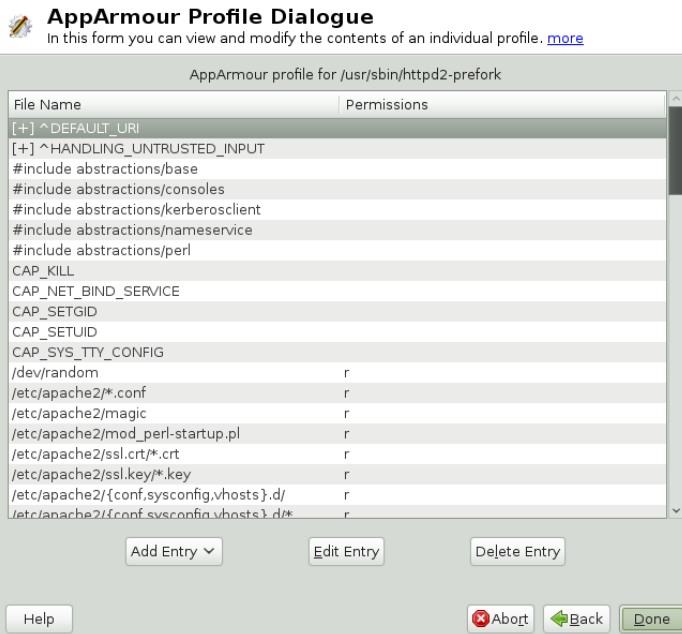
Abort

Back

Next

- 2 From the list of profiled applications, select the profile to edit.

- 3 Click *Next*. The *AppArmor Profile Dialog* window displays the profile.



- 4 In the *AppArmor Profile Dialog* window, add, edit, or delete AppArmor profile entries by clicking the corresponding buttons and referring to Раздел 21.3.1, «Adding an Entry» (стр. 271), Раздел 21.3.2, «Editing an Entry» (стр. 274), or Раздел 21.3.3, «Deleting an Entry» (стр. 274).
- 5 When you are finished, click *Done*.
- 6 In the pop-up that appears, click *Yes* to confirm your changes to the profile and reload the AppArmor profile set.

ПОДСКАЗКА: Syntax Checking in AppArmor

AppArmor contains a syntax check that notifies you of any syntax errors in profiles you are trying to process with the YaST AppArmor tools. If an error occurs, edit the profile manually as `root` and reload the profile set with `rcapparmor reload`.

21.3.1 Adding an Entry

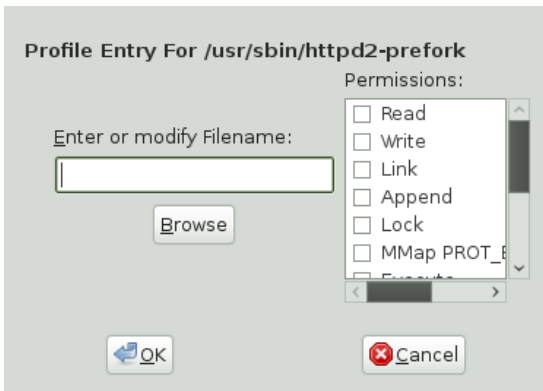
The *Add Entry* option can be found in Раздел 21.2, «Manually Adding a Profile» (стр. 268) or Раздел 21.3, «Editing Profiles» (стр. 269). When you select *Add Entry*, a list shows the types of entries you can add to the AppArmor profile.

From the list, select one of the following:

File

In the pop-up window, specify the absolute path of a file, including the type of access permitted. When finished, click *OK*.

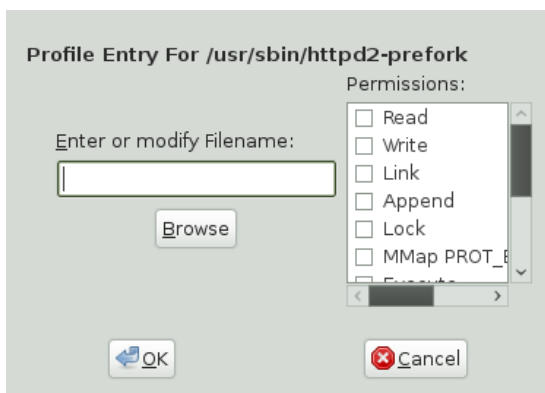
You can use globbing if necessary. For globbing information, refer to Раздел 19.6, «Paths and Globbing» (стр. 243). For file access permission information, refer to Раздел 19.7, «File Permission Access Modes» (стр. 246).



Directory

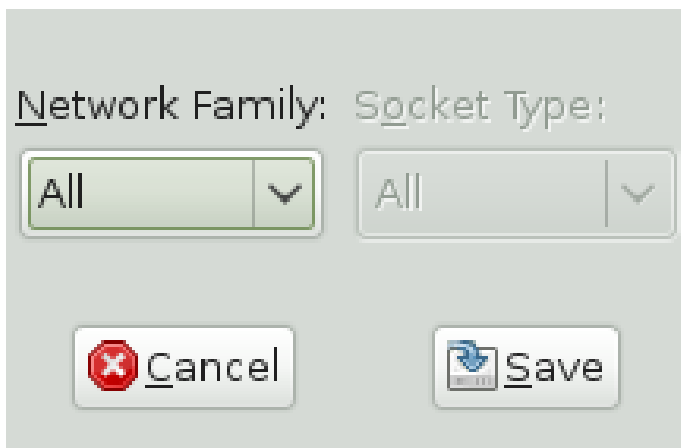
In the pop-up window, specify the absolute path of a directory, including the type of access permitted. You can use globbing if necessary. When finished, click *OK*.

For globbing information, refer to Раздел 19.6, «Paths and Globbing» (стр. 243). For file access permission information, refer to Раздел 19.7, «File Permission Access Modes» (стр. 246).



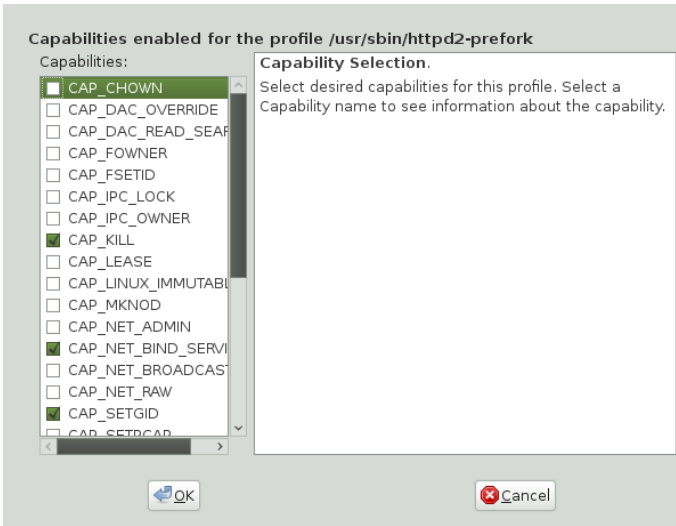
Network Rule

In the pop-up window, select the appropriate network family and the socket type. For more information, refer to Раздел 19.5, «Network Access Control» (срп. 243).



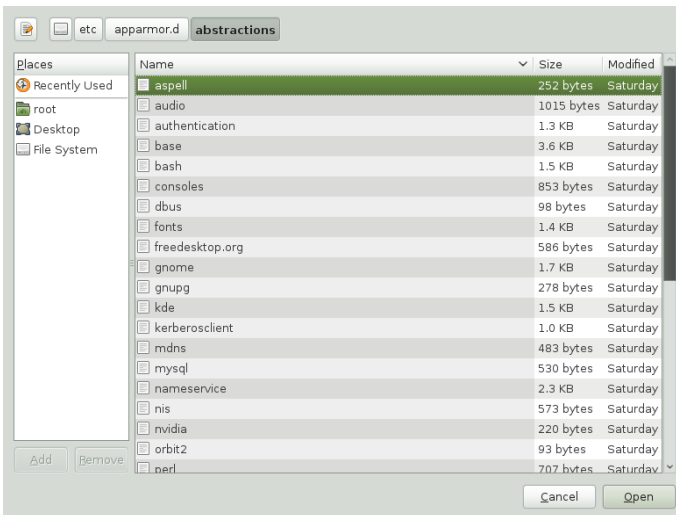
Capability

In the pop-up window, select the appropriate capabilities. These are statements that enable each of the 32 POSIX.1e capabilities. Refer to Раздел 19.4, «Capability Entries (POSIX.1e)» (срп. 242) for more information about capabilities. When finished making your selections, click *OK*.



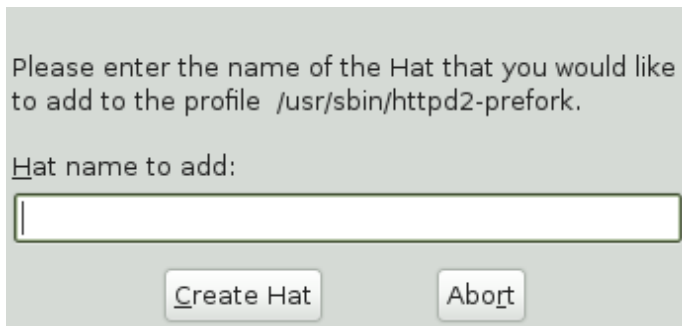
Include

In the pop-up window, browse to the files to use as includes. Includes are directives that pull in components of other AppArmor profiles to simplify profiles. For more information, refer to Раздел 19.3, «`#include Statements`» (стр. 241).



Hat

In the pop-up window, specify the name of the subprofile (*hat*) to add to your current profile and click *Create Hat*. For more information, refer to Глава 23, *Profiling Your Web Applications Using ChangeHat* (стр. 305).



Please enter the name of the Hat that you would like to add to the profile `/usr/sbin/httpd2-prefork`.

Hat name to add:

21.3.2 Editing an Entry

When you select *Edit Entry*, the file browser pop-up window opens. From here, edit the selected entry.

In the pop-up window, specify the absolute path of a file, including the type of access permitted. You can use globbing if necessary. When finished, click *OK*.

For globbing information, refer to Раздел 19.6, «Paths and Globbing» (стр. 243). For file access permission information, refer to Раздел 19.7, «File Permission Access Modes» (стр. 246).

21.3.3 Deleting an Entry

To delete an entry in a given profile, select *Delete Entry*. AppArmor removes the selected profile entry.

21.4 Deleting a Profile

AppArmor enables you to delete an AppArmor profile manually. Simply select the application for which to delete a profile then delete it as follows:

- 1 Start YaST and select *AppArmor > Delete Profile*.
- 2 Select the profile to delete.
- 3 Click *Next*.
- 4 In the pop-up that opens, click *Yes* to delete the profile and reload the AppArmor profile set.

21.5 Updating Profiles from Log Entries

The AppArmor profile wizard uses *aa-logprof*, the tool that scans log files and enables you to update profiles. *aa-logprof* tracks messages from the AppArmor module that represent exceptions for all profiles running on your system. These exceptions represent the behavior of the profiled application that is outside of the profile definition for the program. You can add the new behavior to the relevant profile by selecting the suggested profile entry.

- 1 Start YaST and select *AppArmor > Update Profile Wizard*.

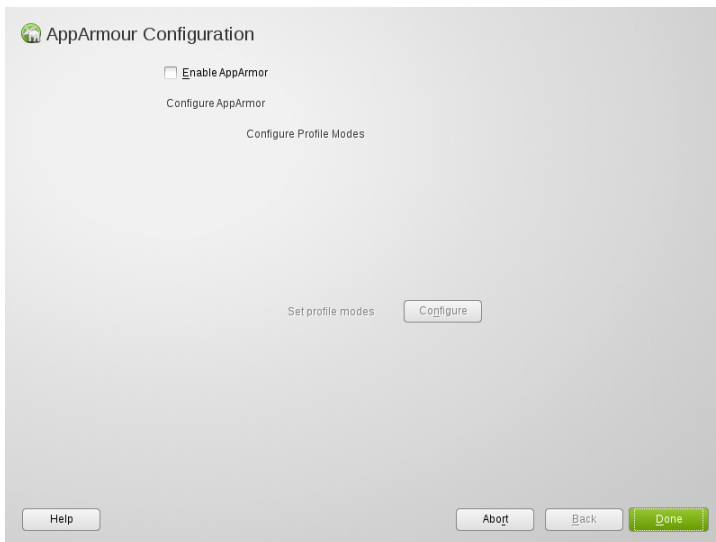
Running *Update Profile Wizard* (*aa-logprof*) parses the learning mode log files. This generates a series of questions that you must answer to guide *aa-logprof* to generate the security profile. The exact procedure is the same as with creating a new profile. Refer to [Шар 9 \(стр. 263\)](#) in [Раздел 21.1](#), «Adding a Profile Using the Wizard» (стр. 261) for details.

- 2 When you are done, click *Finish*. In the following pop-up, click *Yes* to exit the *Add Profile Wizard*. The profile is saved and loaded into the AppArmor module.

21.6 Managing AppArmor

You can change the status of AppArmor by enabling or disabling it. Enabling AppArmor protects your system from potential program exploitation. Disabling AppArmor, even if your profiles have been set up, removes protection from your

system. To change the status of AppArmor, start YaST and select *AppArmor > AppArmor Control Panel*.



To change the status of AppArmor, continue as described in Раздел 21.6.1, «Changing AppArmor Status» (стр. 276). To change the mode of individual profiles, continue as described in Раздел 21.6.2, «Changing the Mode of Individual Profiles» (стр. 277).

21.6.1 Changing AppArmor Status

When you change the status of AppArmor, set it to enabled or disabled. When AppArmor is enabled, it is installed, running, and enforcing the AppArmor security policies.

- 1 Start YaST and select *AppArmor > AppArmor Control Panel*.
- 2 Enable AppArmor by checking *Enable AppArmor* or disable AppArmor by deselecting it.
- 3 Click *Done* in the *AppArmor Configuration* window.
- 4 Click *File > Quit* in the YaST Control Center.

21.6.2 Changing the Mode of Individual Profiles

AppArmor can apply profiles in two different modes. In *complain* or *learning* mode, violations of AppArmor profile rules, such as the profiled program accessing files not permitted by the profile, are detected. The violations are permitted, but also logged. This mode is convenient for developing profiles and is used by the AppArmor tools for generating profiles. Loading a profile in *enforce* mode enforces the policy defined in the profile and reports policy violation attempts to syslogd.

The *Profile Modes* dialog allows you to view and edit the mode of currently loaded AppArmor profiles. This feature is useful for determining the status of your system during profile development. During the course of systemic profiling (see Параграф 22.6.2, «Systemic Profiling» (стр. 283)), you can use this tool to adjust and monitor the scope of the profiles for which you are learning behavior.

To edit an application's profile mode, proceed as follows:

- 1 Start YaST and select *AppArmor > AppArmor Control Panel*.
- 2 In the *Configure Profile Modes* section, select *Configure*.
- 3 Select the profile for which to change the mode.
- 4 Select *Toggle Mode* to set this profile to *complain* mode or to *enforce* mode.
- 5 Apply your settings and leave YaST with *Done*.

To change the mode of all profiles, use *Set All to Enforce* or *Set All to Complain*.

ПОДСКАЗКА: Listing the Profiles Available

By default, only active profiles are listed (any profile that has a matching application installed on your system). To set up a profile before installing the respective application, click *Show All Profiles* and select the profile to configure from the list that appears.

Building Profiles from the Command Line

AppArmor® provides the user the ability to use a command line interface rather than a graphical interface to manage and configure the system security. Track the status of AppArmor and create, delete, or modify AppArmor profiles using the AppArmor command line tools.

ПОДСКАЗКА: Background Information

Before starting to manage your profiles using the AppArmor command line tools, check out the general introduction to AppArmor given in Глава 18, *Immunizing Programs* (стр. 225) and Глава 19, *Profile Components and Syntax* (стр. 235).

22.1 Checking the AppArmor Module Status

An AppArmor module can be in any one of three states:

Unloaded

The AppArmor module is not loaded into the kernel.

Running

The AppArmor module is loaded into the kernel and is enforcing AppArmor program policies.

Stopped

The AppArmor module is loaded into the kernel, but no policies are enforced.

Detect the state of the AppArmor module by inspecting `/sys/kernel/security/apparmor/profiles`. If `cat /sys/kernel/security/apparmor/profiles` reports a list of profiles, AppArmor is running. If it is empty and returns nothing, AppArmor is stopped. If the file does not exist, AppArmor is unloaded.

Manage AppArmor through the script `rcapparmor`, which can perform the following operations:

`rcapparmor start`

Behavior depends on the AppArmor module state. If it is unloaded, `start` loads the module and starts it, putting it in the running state. If it is stopped, `start` causes the module to rescan the AppArmor profiles usually found in `/etc/apparmor.d` and puts the module in the running state. If the module is already running, `start` reports a warning and takes no action.

`rcapparmor stop`

Stops the AppArmor module if it is running by removing all profiles from kernel memory, effectively disabling all access controls, and putting the module into the stopped state. If the AppArmor module is unloaded or already stopped, `stop` tries to unload the profiles again, but nothing happens.

`rcapparmor restart`

Causes the AppArmor module to rescan the profiles in `/etc/apparmor.d` without unconfining running processes. Freshly created profiles are enforced and recently deleted ones are removed from the `/etc/apparmor.d` directory.

`rcapparmor kill`

Unconditionally removes the AppArmor module from the kernel. However, unloading modules from the Linux kernel is unsafe. This command is provided only for debugging and emergencies (when the module might need to be removed).

ПРЕДУПРЕЖДЕНИЕ

AppArmor is a powerful access control system and it is possible to lock yourself out of your own machine to the point where you must boot the machine from a rescue medium (such as the first medium of) to regain control.

To prevent such a problem, always ensure that you have a running, unconfined, `root` login on the machine being configured when you restart the AppArmor module. If you damage your system to the point where logins are no longer possible (for example, by breaking the profile associated with the SSH daemon), you can repair the damage using your running `root` prompt then restarting the AppArmor module.

22.2 Building AppArmor Profiles

The AppArmor module profile definitions are stored in the `/etc/apparmor.d` directory as plain text files. For a detailed description of the syntax of these files, refer to Глава 19, *Profile Components and Syntax* (стр. 235).

All files in the `/etc/apparmor.d` directory are interpreted as profiles and are loaded as such. Renaming files in that directory is not an effective way of preventing profiles from being loaded. You must remove profiles from this directory to prevent them from being read and evaluated effectively.

You can use a text editor, such as `vim`, to access and make changes to these profiles. The following options contain detailed steps for building profiles:

Adding or Creating AppArmor Profiles

Refer to Раздел 22.3, «Adding or Creating an AppArmor Profile» (стр. 281)

Editing AppArmor Profiles

Refer to Раздел 22.4, «Editing an AppArmor Profile» (стр. 282)

Deleting AppArmor Profiles

Refer to Раздел 22.5, «Deleting an AppArmor Profile» (стр. 282)

22.3 Adding or Creating an AppArmor Profile

To add or create an AppArmor profile for an application, you can use a systemic or stand-alone profiling method, depending on your needs. Learn more about these two approaches in Раздел 22.6, «Two Methods of Profiling» (стр. 282).

22.4 Editing an AppArmor Profile

The following steps describe the procedure for editing an AppArmor profile:

- 1 If you are not currently logged in as `root`, enter `su` in a terminal window.
- 2 Enter the `root` password when prompted.
- 3 Go to the profile directory with `cd /etc/apparmor.d/`.
- 4 Enter `ls` to view all profiles currently installed.
- 5 Open the profile to edit in a text editor, such as `vim`.
- 6 Make the necessary changes then save the profile.
- 7 Restart AppArmor by entering `rcapparmor restart` in a terminal window.

22.5 Deleting an AppArmor Profile

The following steps describe the procedure for deleting an AppArmor profile.

- 1 If you are not currently logged in as `root`, enter `su` in a terminal window.
- 2 Enter the `root` password when prompted.
- 3 Go to the AppArmor directory with `cd /etc/apparmor.d/`.
- 4 Enter `ls` to view all the AppArmor profiles that are currently installed.
- 5 Delete the profile with `rm filename`.
- 6 Restart AppArmor by entering `rcapparmor restart` in a terminal window.

22.6 Two Methods of Profiling

Given the syntax for AppArmor profiles in Глава 19, *Profile Components and Syntax* (стр. 235), you could create profiles without using the tools. However, the

effort involved would be substantial. To avoid such a hassle, use the AppArmor tools to automate the creation and refinement of profiles.

There are two ways to approach AppArmor profile creation. Tools are available for both methods.

Stand-Alone Profiling

A method suitable for profiling small applications that have a finite run time, such as user client applications like mail clients. For more information, refer to Раздел 22.6.1, «Stand-Alone Profiling» (стр. 283).

Systemic Profiling

A method suitable for profiling large numbers of programs all at once and for profiling applications that may run for days, weeks, or continuously across reboots, such as network server applications like Web servers and mail servers. For more information, refer to Раздел 22.6.2, «Systemic Profiling» (стр. 283).

Automated profile development becomes more manageable with the AppArmor tools:

- 1 Decide which profiling method suits your needs.
- 2 Perform a static analysis. Run either `aa-genprof` or `aa-autodep`, depending on the profiling method chosen.
- 3 Enable dynamic learning. Activate learning mode for all profiled programs.

22.6.1 Stand-Alone Profiling

Stand-alone profile generation and improvement is managed by a program called `aa-genprof`. This method is easy because `aa-genprof` takes care of everything, but is limited because it requires `aa-genprof` to run for the entire duration of the test run of your program (you cannot reboot the machine while you are still developing your profile).

To use `aa-genprof` for the stand-alone method of profiling, refer to Раздел 22.6.3.4, «`aa-genprof`—Generating Profiles» (стр. 289).

22.6.2 Systemic Profiling

This method is called *systemic profiling* because it updates all of the profiles on the system at once, rather than focusing on the one or few targeted by `aa-genprof` or

stand-alone profiling. With systemic profiling, profile construction and improvement are somewhat less automated, but more flexible. This method is suitable for profiling long-running applications whose behavior continues after rebooting, or a large number of programs all at once.

Build an AppArmor profile for a group of applications as follows:

1 Create profiles for the individual programs that make up your application.

Although this approach is systemic, AppArmor only monitors those programs with profiles and their children. To get AppArmor to consider a program, you must at least have `aa-autodep` create an approximate profile for it. To create this approximate profile, refer to Раздел 22.6.3.1, «`aa-autodep`—Creating Approximate Profiles» (стр. 286).

2 Put relevant profiles into learning or complain mode.

Activate learning or complain mode for all profiled programs by entering `aa-complain /etc/apparmor.d/*` in a terminal window while logged in as `root`. This functionality is also available through the YaST Profile Mode module, described in Раздел 21.6.2, «Changing the Mode of Individual Profiles» (стр. 277).

When in learning mode, access requests are not blocked, even if the profile dictates that they should be. This enables you to run through several tests (as shown in Иллар 3 (стр. 284)) and learn the access needs of the program so it runs properly. With this information, you can decide how secure to make the profile.

Refer to Раздел 22.6.3.2, «`aa-complain`—Entering Complain or Learning Mode» (стр. 287) for more detailed instructions for using learning or complain mode.

3 Exercise your application.

Run your application and exercise its functionality. How much to exercise the program is up to you, but you need the program to access each file representing its access needs. Because the execution is not being supervised by `aa-genprof`, this step can go on for days or weeks and can span complete system reboots.

4 Analyze the log.

In systemic profiling, run `aa-logprof` directly instead of letting `aa-genprof` run it (as in stand-alone profiling). The general form of `aa-logprof` is:

```
aa-logprof [ -d /path/to/profiles ] [ -f /path/to/logfile ]
```

Refer to Раздел 22.6.3.5, «`aa-logprof`—Scanning the System Log» (стр. 297) for more information about using `aa-logprof`.

5 Repeat Шаг 3 (стр. 284) and Шаг 4 (стр. 284).

This generates optimum profiles. An iterative approach captures smaller data sets that can be trained and reloaded into the policy engine. Subsequent iterations generate fewer messages and run faster.

6 Edit the profiles.

You might want to review the profiles that have been generated. You can open and edit the profiles in `/etc/apparmor.d/` using `vim`.

7 Return to enforce mode.

This is when the system goes back to enforcing the rules of the profiles, not just logging information. This can be done manually by removing the `flags=(complain)` text from the profiles or automatically by using the `aa-enforce` command, which works identically to the `aa-complain` command, except it sets the profiles to enforce mode. This functionality is also available through the YaST Profile Mode module, described in Раздел 21.6.2, «Changing the Mode of Individual Profiles» (стр. 277).

To ensure that all profiles are taken out of complain mode and put into enforce mode, enter `aa-enforce /etc/apparmor.d/*`.

8 Rescan all profiles.

To have AppArmor rescan all of the profiles and change the enforcement mode in the kernel, enter `rcapparmor restart`.

22.6.3 Summary of Profiling Tools

All of the AppArmor profiling utilities are provided by the `apparmor-utils` RPM package and are stored in `/usr/sbin`. Each tool has a different purpose.

22.6.3.1 aa-autodep—Creating Approximate Profiles

This creates an approximate profile for the program or application selected. You can generate approximate profiles for binary executables and interpreted script programs. The resulting profile is called «approximate» because it does not necessarily contain all of the profile entries that the program needs to be properly confined by AppArmor. The minimum aa-autodep approximate profile has, at minimum, a base include directive, which contains basic profile entries needed by most programs. For certain types of programs, aa-autodep generates a more expanded profile. The profile is generated by recursively calling `ldd(1)` on the executables listed on the command line.

To generate an approximate profile, use the aa-autodep program. The program argument can be either the simple name of the program, which aa-autodep finds by searching your shell's path variable, or it can be a fully qualified path. The program itself can be of any type (ELF binary, shell script, Perl script, etc.). aa-autodep generates an approximate profile to improve through the dynamic profiling that follows.

The resulting approximate profile is written to the `/etc/apparmor.d` directory using the AppArmor profile naming convention of naming the profile after the absolute path of the program, replacing the forward slash (/) characters in the path with period (.) characters. The general form of aa-autodep is to enter the following in a terminal window when logged in as `root`:

```
aa-autodep [ -d /path/to/profiles ] [program1 program2...]
```

If you do not enter the program name or names, you are prompted for them. */path/to/profiles* overrides the default location of `/etc/apparmor.d`, should you keep profiles in a location other than the default.

To begin profiling, you must create profiles for each main executable service that is part of your application (anything that might start without being a child of another program that already has a profile). Finding all such programs depends on the application in question. Here are several strategies for finding such programs:

Directories

If all the programs to profile are in one directory and there are no other programs in that directory, the simple command `aa-autodep /path/to/your/programs/*` creates basic profiles for all programs in that directory.

ps command

You can run your application and use the standard Linux `ps` command to find all processes running. Then manually hunt down the location of these programs and run the `aa-autodep` for each one. If the programs are in your path, `aa-autodep` finds them for you. If they are not in your path, the standard Linux command `find` might be helpful in finding your programs. Execute `find / -name 'my_application' -print` to determine an application's path (*my_application* being an example application). You may use wild cards if appropriate.

22.6.3.2 aa-complain—Entering Complain or Learning Mode

The complain or learning mode tool (`aa-complain`) detects violations of AppArmor profile rules, such as the profiled program accessing files not permitted by the profile. The violations are permitted, but also logged. To improve the profile, turn complain mode on, run the program through a suite of tests to generate log events that characterize the program's access needs, then postprocess the log with the AppArmor tools to transform log events into improved profiles.

Manually activating complain mode (using the command line) adds a flag to the top of the profile so that `/bin/foo` becomes `/bin/foo flags=(complain)`. To use complain mode, open a terminal window and enter one of the following lines as root:

- If the example program (*program1*) is in your path, use:

```
aa-complain [program1 program2 ...]
```

- If the program is not in your path, specify the entire path as follows:

```
aa-complain /sbin/program1
```

- If the profiles are not in `/etc/apparmor.d`, use the following to override the default location:

```
aa-complain /path/to/profiles/ program1
```

- Specify the profile for *program1* as follows:

```
aa-complain /etc/apparmor.d/sbin.program1
```

Each of the above commands activates the complain mode for the profiles or programs listed. If the program name does not include its entire path, `aa-complain` searches `$PATH` for the program. For instance, `aa-complain /usr/sbin/*` finds profiles associated with all of the programs in `/usr/sbin` and puts them into complain mode. `aa-complain /etc/apparmor.d/*` puts all of the profiles in `/etc/apparmor.d` into complain mode.

ПОДСКАЗКА: Toggling Profile Mode with YaST

YaST offers a graphical front-end for toggling complain and enforce mode. See Раздел 21.6.2, «Changing the Mode of Individual Profiles» (стр. 277) for information.

22.6.3.3 aa-enforce—Entering Enforce Mode

The enforce mode detects violations of AppArmor profile rules, such as the profiled program accessing files not permitted by the profile. The violations are logged and not permitted. The default is for enforce mode to be enabled. To log the violations only, but still permit them, use complain mode. Enforce toggles with complain mode.

Manually activating enforce mode (using the command line) adds a flag to the top of the profile so that `/bin/foo` becomes `/bin/foo flags=(enforce)`. To use enforce mode, open a terminal window and enter one of the following lines as `root`.

- If the example program (*program1*) is in your path, use:

```
aa-enforce [program1 program2 ...]
```

- If the program is not in your path, specify the entire path, as follows:

```
aa-enforce /sbin/program1
```

- If the profiles are not in `/etc/apparmor.d`, use the following to override the default location:

```
aa-enforce /path/to/profiles/program1
```

- Specify the profile for *program1* as follows:

```
aa-enforce /etc/apparmor.d/sbin.program1
```

Each of the above commands activates the enforce mode for the profiles and programs listed.

If you do not enter the program or profile names, you are prompted to enter one. `/path/to/profiles` overrides the default location of `/etc/apparmor.d`.

The argument can be either a list of programs or a list of profiles. If the program name does not include its entire path, `aa-enforce` searches `$PATH` for the program.

ПОДСКАЗКА: Toggling Profile Mode with YaST

YaST offers a graphical front-end for toggling complain and enforce mode. See Раздел 21.6.2, «Changing the Mode of Individual Profiles» (стр. 277) for information.

22.6.3.4 aa-genprof—Generating Profiles

`aa-genprof` is AppArmor's profile generating utility. It runs `aa-autodep` on the specified program, creating an approximate profile (if a profile does not already exist for it), sets it to complain mode, reloads it into AppArmor, marks the log, and prompts the user to execute the program and exercise its functionality. Its syntax is as follows:

```
aa-genprof [ -d /path/to/profiles ] program
```

To create a profile for the the Apache Web server program `httpd2-prefork`, do the following as `root`:

- 1 Enter `rcapache2 stop`.
- 2 Next, enter `aa-genprof httpd2-prefork`.

Now `aa-genprof` does the following:

1. Resolves the full path of `httpd2-prefork` using your shell's path variables. You can also specify a full path. On , the default full path is `/usr/sbin/httpd2-prefork`.
2. Checks to see if there is an existing profile for `httpd2-prefork`. If there is one, it updates it. If not, it creates one using the `aa-autodep` as described in Раздел 22.6.3, «Summary of Profiling Tools» (стр. 285).
3. Puts the profile for this program into learning or complain mode so that profile violations are logged, but are permitted to proceed. A log event looks like this (see `/var/log/audit/audit.log`):

```
type=APPARMOR_ALLOWED msg=audit(1189682639.184:20816):  
operation="file_mmap" requested_mask="::r" denied_mask="::r" fsuid=30  
name="/srv/www/htdocs/index.html" pid=27471 profile="null-complain-  
profile"
```

If you are not running the audit daemon, the AppArmor events are logged to /var/log/messages:

```
Sep 13 13:20:30 K23 kernel: audit(1189682430.672:20810):  
operation="file_mmap" requested_mask="::r" denied_mask="::r" fsuid=30  
name="/srv/www/htdocs/phpsysinfo/templates/bulix/form.tpl" pid=30405  
profile="/usr/sbin/httpd2-prefork//phpsysinfo/"
```

They also can be viewed using the `dmesg` command:

```
audit(1189682430.672:20810): operation="file_mmap" requested_mask="::r"  
denied_mask="::r" fsuid=30 name="/srv/www/htdocs/phpsysinfo/templates/  
bulix/form.tpl" pid=30405 profile="/usr/sbin/httpd2-prefork//  
phpsysinfo/"
```

4. Marks the log with a beginning marker of log events to consider. For example:

```
Sep 13 17:48:52 figwit root: GenProf: e2ff78636296f16d0b5301209a04430d
```

- 3 When prompted by the tool, run the application to profile in another terminal window and perform as many of the application functions as possible. Thus, the learning mode can log the files and directories to which the program requires access in order to function properly. For example, in a new terminal window, enter `rcapache2 start`.
- 4 Select from the following options that are available in the `aa-logprof` terminal window after you have executed the program function:
 - **S** runs `aa-logprof` on the system log from where it was marked when `aa-genprof` was started and reloads the profile. If system events exist in the log, AppArmor parses the learning mode log files. This generates a series of questions that you must answer to guide `aa-genprof` in generating the security profile.
 - **F** exits the tool and returns to the main menu.

ПРИМЕЧАНИЕ

If requests to add hats appear, proceed to Глава 23, *Profiling Your Web Applications Using ChangeHat* (стр. 305).

5 Answer two types of questions:

- A resource is requested by a profiled program that is not in the profile (see Пример 22.1, «Learning Mode Exception: Controlling Access to Specific Resources» (стр. 291)).
- A program is executed by the profiled program and the security domain transition has not been defined (see Пример 22.2, «Learning Mode Exception: Defining Execute Permissions for an Entry» (стр. 293)).

Each of these categories results in a series of questions that you must answer to add the resource or program to the profile. Пример 22.1, «Learning Mode Exception: Controlling Access to Specific Resources» (стр. 291) and Пример 22.2, «Learning Mode Exception: Defining Execute Permissions for an Entry» (стр. 293) provide examples of each one. Subsequent steps describe your options in answering these questions.

- Dealing with execute accesses is complex. You must decide how to proceed with this entry regarding which execute permission type to grant to this entry:

Пример 22.1 *Learning Mode Exception: Controlling Access to Specific Resources*

```
Reading log entries from /var/log/audit/audit.log.  
Updating AppArmor profiles in /etc/apparmor.d.
```

```
Profile: /usr/sbin/xinetd  
Program: xinetd  
Execute: /usr/lib/cups/daemon/cups-lpd  
Severity: unknown
```

```
[(I)nherit] / (P)rofile / (U)nconfined / (D)eny / Abo(r)t / (F)inish
```

Inherit (ix)

The child inherits the parent's profile, running with the same access controls as the parent. This mode is useful when a confined program needs to call another confined program without gaining the permissions of the target's profile or losing the permissions of the current profile. This mode is often used when the child program is a *helper application*, such as the `/usr/bin/mail` client using `less` as a pager or the Mozilla* Web browser using Adobe Acrobat* to display PDF files.

Profile (px)

The child runs using its own profile, which must be loaded into the kernel. If the profile is not present, attempts to execute the child fail with permission denied. This is most useful if the parent program is invoking a global service, such as DNS lookups or sending mail with your system's MTA.

Choose the *profile with clean exec* (Px) option to scrub the environment of environment variables that could modify execution behavior when passed to the child process.

Unconfined (ux)

The child runs completely unconfined without any AppArmor profile applied to the executed resource.

Choose the *unconfined with clean exec* (Ux) option to scrub the environment of environment variables that could modify execution behavior when passed to the child process. This option introduces a security vulnerability that could be used to exploit AppArmor. Only use it as a last resort.

mmap (m)

This permission denotes that the program running under the profile can access the resource using the mmap system call with the flag `PROT_EXEC`. This means that the data mapped in it can be executed. You are prompted to include this permission if it is requested during a profiling run.

Deny

Prevents the program from accessing the specified directory path entries. AppArmor then continues to the next event.

Abort

Aborts aa-logprof, losing all rule changes entered so far and leaving all profiles unmodified.

Finish

Closes aa-logprof, saving all rule changes entered so far and modifying all profiles.

- Пример 22.2, «Learning Mode Exception: Defining Execute Permissions for an Entry» (стр. 293) shows AppArmor suggesting directory path entries that

have been accessed by the application being profiled. It might also require you to define execute permissions for entries.

Пример 22.2 *Learning Mode Exception: Defining Execute Permissions for an Entry*

Adding /bin/ps ix to profile.

```
Profile: /usr/sbin/xinetd
Path:    /etc/hosts.allow
New Mode: r
```

```
[1 - /etc/hosts.allow]
```

```
[(A)llow] / (D)eny / (N)ew / (G)lob / Glob w/(E)xt / Abo(r)t / (F)inish
```

AppArmor provides one or more paths or includes. By entering the option number, select the desired options then proceed to the next step.

ПРИМЕЧАНИЕ

All of these options are not always presented in the AppArmor menu.

`#include`

This is the section of an AppArmor profile that refers to an include file, which procures access permissions for programs. By using an include, you can give the program access to directory paths or files that are also required by other programs. Using includes can reduce the size of a profile. It is good practice to select includes when suggested.

Globbed Version

This is accessed by selecting *Glob* as described in the next step. For information about globbing syntax, refer to Раздел 19.6, «Paths and Globbing» (стр. 243).

Actual Path

This is the literal path to which the program needs access so that it can run properly.

After you select the path or include, process it as an entry into the AppArmor profile by selecting *Allow* or *Deny*. If you are not satisfied with the directory path entry as it is displayed, you can also *Glob* it.

The following options are available to process the learning mode entries and build the profile:

Select

Allows access to the selected directory path.

Allow

Allows access to the specified directory path entries. AppArmor suggests file permission access. For more information, refer to Раздел 19.7, «File Permission Access Modes» (стр. 246).

Deny

Prevents the program from accessing the specified directory path entries. AppArmor then continues to the next event.

New

Prompts you to enter your own rule for this event, allowing you to specify a regular expression. If the expression does not actually satisfy the event that prompted the question in the first place, AppArmor asks for confirmation and lets you reenter the expression.

Glob

Select a specific path or create a general rule using wild cards that match a broader set of paths. To select any of the offered paths, enter the number that is printed in front of the path then decide how to proceed with the selected item.

For more information about globbing syntax, refer to Раздел 19.6, «Paths and Globbing» (стр. 243).

Glob w/Ext

This modifies the original directory path while retaining the filename extension. For example, `/etc/apache2/file.ext` becomes `/etc/apache2/* .ext`, adding the wild card (asterisk) in place of the filename. This allows the program to access all files in the suggested directory that end with the `.ext` extension.

Abort

Aborts aa-logprof, losing all rule changes entered so far and leaving all profiles unmodified.

Finish

Closes aa-logprof, saving all rule changes entered so far and modifying all profiles.

- 6 To view and edit your profile using vim, enter `vim /etc/apparmor.d/profilename` in a terminal window.
- 7 Restart AppArmor and reload the profile set including the newly created one using the `rcapparmor restart` command.

Like the graphical front-end for building AppArmor profiles, the YaST Add Profile Wizard, aa-genprof also supports the use of the local profile repository under `/etc/apparmor/profiles/extras` and the remote AppArmor profile repository.

To use a profile from the local repository, proceed as follows:

- 1 Start aa-genprof as described above.

If aa-genprof finds an inactive local profile, the following lines appear on your terminal window:

```
Profile: /usr/bin/opera
```

```
[1 - Inactive local profile for /usr/bin/opera]
```

```
[(V)iew Profile] / (U)se Profile / (C)reate New Profile / Abo(r)t /  
(F)inish
```

- 2 If you want to just use this profile, hit **U** (*Use Profile*) and follow the profile generation procedure outlined above.

If you want to examine the profile before activating it, hit **V** (*View Profile*).

If you want to ignore the existing profile, hit **C** (*Create New Profile*) and follow the profile generation procedure outlined above to create the profile from scratch.

- 3 Leave aa-genprof by hitting **F** (*Finish*) when you are done and save your changes.

To use the remote AppArmor profile repository with aa-genprof, proceed as follows:

- 1 Start aa-genprof as described above.

If aa-genprof detects a suitable profile on the repository server, the following lines appear on your terminal window:

```
Repository: http://apparmor.opensuse.org/backend/api

Would you like to enable access to the profile repository?

(E)nable Repository / (D)isable Repository / Ask Me (L)ater
```

2 Hit **E** (*Enable Repository*) to enable the repository.

3 Determine whether you want to aa-genprof to upload any profiles to the repository server:

```
Would you like to upload newly created and changed profiles to
the profile repository?

(Y)es / (N)o / Ask Me (L)ater
```

Hit **Y** (*Yes*), if you want to enable profile upload or select **N** (*No*), if you want aa-genprof to just pull profiles from the repository, but not to upload any.

4 Create a new user on the profile repository server to be able to upload profiles. Provide username and password.

5 Determine whether you want to use the profile downloaded from the server or whether you would just like to review it:

```
Profile: /usr/bin/opera

[1 - novell]

[(V)iew Profile] / (U)se Profile / (C)reate New Profile / Abo(r)t /
(F)inish
```

If you want to just use this profile, hit **U** (*Use Profile*) and follow the profile generation procedure outlined above.

If you want to examine the profile before activating it, hit **V** (*View Profile*).

If you want to ignore the existing profile, hit **C** (*Create New Profile*) and follow the profile generation procedure outlined above to create the profile from scratch.

6 Leave aa-genprof by hitting **F** (*Finish*) when you are done and save the profile.

If you opted for uploading your profile, provide a short change log and push it to the repository.

22.6.3.5 aa-logprof—Scanning the System Log

aa-logprof is an interactive tool used to review the learning or complain-mode output found in the log entries in `/var/log/audit/audit.log` or `/var/log/messages` (if auditd is not running) and generate new entries in AppArmor security profiles.

When you run aa-logprof, it begins to scan the log files produced in learning or complain mode and, if there are new security events that are not covered by the existing profile set, it gives suggestions for modifying the profile. The learning or complain mode traces program behavior and enters it in the log. aa-logprof uses this information to observe program behavior.

If a confined program forks and executes another program, aa-logprof sees this and asks the user which execution mode should be used when launching the child process. The execution modes *ix*, *px*, *Px*, *ux*, and *Ux* are options for starting the child process. If a separate profile exists for the child process, the default selection is *px*. If one does not exist, the profile defaults to *ix*. Child processes with separate profiles have aa-autodep run on them and are loaded into AppArmor, if it is running.

When aa-logprof exits, profiles are updated with the changes. If the AppArmor module is running, the updated profiles are reloaded and, if any processes that generated security events are still running in the null-complain-profile, those processes are set to run under their proper profiles.

To run aa-logprof, enter aa-logprof into a terminal window while logged in as root. The following options can be used for aa-logprof:

```
aa-logprof -d /path/to/profile/directory/
```

Specifies the full path to the location of the profiles if the profiles are not located in the standard directory, `/etc/apparmor.d/`.

```
aa-logprof -f /path/to/logfile/
```

Specifies the full path to the location of the log file if the log file is not located in the default directory, `/var/log/audit/audit.log` or `/var/log/messages` (if auditd is not running).

```
aa-logprof -m "string marker in logfile"
```

Marks the starting point for aa-logprof to look in the system log. aa-logprof ignores all events in the system log before the specified mark. If the mark contains spaces, it must be surrounded by quotes to work correctly. For example:

```
aa-logprof -m"17:04:21"
```

or

```
logprof -m e2ff78636296f16d0b5301209a04430d
```

aa-logprof scans the log, asking you how to handle each logged event. Each question presents a numbered list of AppArmor rules that can be added by pressing the number of the item on the list.

By default, aa-logprof looks for profiles in `/etc/apparmor.d/` and scans the log in `/var/log/messages`. In many cases, running aa-logprof as root is enough to create the profile.

However, there might be times when you need to search archived log files, such as if the program exercise period exceeds the log rotation window (when the log file is archived and a new log file is started). If this is the case, you can enter `zcat -f `ls -ltr /var/log/messages*` | aa-logprof -f -`.

22.6.3.6 aa-logprof Example 1

The following is an example of how aa-logprof addresses httpd2-prefork accessing the file `/etc/group`. `[]` indicates the default option.

In this example, the access to `/etc/group` is part of httpd2-prefork accessing name services. The appropriate response is 1, which includes a predefined set of AppArmor rules. Selecting 1 to `#include` the name service package resolves all of the future questions pertaining to DNS lookups and also makes the profile less brittle in that any changes to DNS configuration and the associated name service profile package can be made just once, rather than needing to revise many profiles.

```
Profile: /usr/sbin/httpd2-prefork
Path:    /etc/group
New Mode: r

[1 - #include <abstractions/nameservice>]
 2 - /etc/group
[(A)llow] / (D)eny / (N)ew / (G)lob / Glob w/(E)xt / Abo(r)t / (F)inish
```

Select one of the following responses:

Select

Triggers the default action, which is, in this example, allowing access to the specified directory path entry.

Allow

Allows access to the specified directory path entries. AppArmor suggests file permission access. For more information about this, refer to Раздел 19.7, «File Permission Access Modes» (срп. 246).

Deny

Prevents the program from accessing the specified directory path entries. AppArmor then continues to the next event.

New

Prompts you to enter your own rule for this event, allowing you to specify whatever form of regular expression you want. If the expression entered does not actually satisfy the event that prompted the question in the first place, AppArmor asks for confirmation and lets you reenter the expression.

Glob

Select either a specific path or create a general rule using wild cards that matches on a broader set of paths. To select any of the offered paths, enter the number that is printed in front of the paths then decide how to proceed with the selected item.

For more information about globbing syntax, refer to Раздел 19.6, «Paths and Globbing» (срп. 243).

Glob w/Ext

This modifies the original directory path while retaining the filename extension. For example, `/etc/apache2/file.ext` becomes `/etc/apache2/*.ext`, adding the wild card (asterisk) in place of the filename. This allows the program to access all files in the suggested directory that end with the `.ext` extension.

Abort

Aborts aa-logprof, losing all rule changes entered so far and leaving all profiles unmodified.

Finish

Closes aa-logprof, saving all rule changes entered so far and modifying all profiles.

22.6.3.7 aa-logprof Example 2

For example, when profiling vsftpd, see this question:

```
Profile: /usr/sbin/vsftpd
Path: /y2k.jpg
New Mode: r
```

```
[1 - /y2k.jpg]
```

```
(A)llow / [(D)eny] / (N)ew / (G)lob / Glob w/(E)xt / Abo(r)t / (F)inish
```

Several items of interest appear in this question. First, note that vsftpd is asking for a path entry at the top of the tree, even though vsftpd on `srv` serves FTP files from `/srv/ftp` by default. This is because `httpd2-prefork` uses `chroot` and, for the portion of the code inside the `chroot` jail, AppArmor sees file accesses in terms of the `chroot` environment rather than the global absolute path.

The second item of interest is that you might want to grant FTP read access to all JPEG files in the directory, so you could use *Glob w/Ext* and use the suggested path of `/*.jpg`. Doing so collapses all previous rules granting access to individual `.jpg` files and forestalls any future questions pertaining to access to `.jpg` files.

Finally, you might want to grant more general access to FTP files. If you select *Glob* in the last entry, `aa-logprof` replaces the suggested path of `/y2k.jpg` with `/*`. Alternatively, you might want to grant even more access to the entire directory tree, in which case you could use the *New* path option and enter `/**/*.jpg` (which would grant access to all `.jpg` files in the entire directory tree) or `/**` (which would grant access to all files in the directory tree).

These items deal with read accesses. Write accesses are similar, except that it is good policy to be more conservative in your use of regular expressions for write accesses. Dealing with execute accesses is more complex. Find an example in Пример 22.1, «Learning Mode Exception: Controlling Access to Specific Resources» (стр. 291).

In the following example, the `/usr/bin/mail` mail client is being profiled and `aa-logprof` has discovered that `/usr/bin/mail` executes `/usr/bin/less` as a helper application to «page» long mail messages. Consequently, it presents this prompt:

```
/usr/bin/nail -> /usr/bin/less
(I)nherit / (P)rofile / (U)nconfined / (D)eny
```

ПОДСКАЗКА

The actual executable file for `/usr/bin/mail` turns out to be `/usr/bin/nail`, which is not a typographical error.

The program `/usr/bin/less` appears to be a simple one for scrolling through text that is more than one screen long and that is in fact what `/usr/bin/mail` is using it for. However, `less` is actually a large and powerful program that makes use of many other helper applications, such as `tar` and `rpm`.

ПОДСКАЗКА

Run `less` on a tar file or an RPM file and it shows you the inventory of these containers.

You do not want to run `rpm` automatically when reading mail messages (that leads directly to a Microsoft* Outlook-style virus attack, because `rpm` has the power to install and modify system programs), so, in this case, the best choice is to use *Inherit*. This results in the `less` program executed from this context running under the profile for `/usr/bin/mail`. This has two consequences:

- You need to add all of the basic file accesses for `/usr/bin/less` to the profile for `/usr/bin/mail`.
- You can avoid adding the helper applications, such as `tar` and `rpm`, to the `/usr/bin/mail` profile so that when `/usr/bin/mail` runs `/usr/bin/less` in this context, the `less` program is far less dangerous than it would be without AppArmor protection.

In other circumstances, you might instead want to use the *Profile* option. This has two effects on `aa-logprof`:

- The rule written into the profile uses `px`, which forces the transition to the child's own profile.
- `aa-logprof` constructs a profile for the child and starts building it, in the same way that it built the parent profile, by assigning events for the child process to the child's profile and asking the `aa-logprof` user questions.

If a confined program forks and executes another program, `aa-logprof` sees this and asks the user which execution mode should be used when launching the child process. The execution modes of `inherit`, `profile`, `unconfined` or an option to deny the execution are presented.

If a separate profile exists for the child process, the default selection is `profile`. If a profile does not exist, the default is `inherit`. The `inherit` option, or `ix`, is described in Раздел 19.7, «File Permission Access Modes» (стр. 246).

The profile option indicates that the child program should run in its own profile. A secondary question asks whether to sanitize the environment that the child program inherits from the parent. If you choose to sanitize the environment, this places the execution modifier `Px` in your AppArmor profile. If you select not to sanitize, `px` is placed in the profile and no environment sanitizing occurs. The default for the execution mode is `px` if you select profile execution mode.

The unconfined execution mode is not recommended and should only be used in cases where there is no other option to generate a profile for a program reliably. Selecting unconfined opens a warning dialog asking for confirmation of the choice. If you are sure and choose *Yes*, a second dialog ask whether to sanitize the environment. Choosing *Yes* uses the execution mode `Ux` in your profile. Choosing *No* uses the execution mode `ux` for your profile. The default value selected is `Ux` for unconfined execution mode.

ВАЖНО: Running Unconfined

Choosing `ux` is very dangerous and provides no enforcement of policy (from a security perspective) of the resulting execution behavior of the child program.

22.6.3.8 aa-unconfined—Identifying Unprotected Processes

The `aa-unconfined` command examines open network ports on your system, compares that to the set of profiles loaded on your system, and reports network services that do not have AppArmor profiles. It requires `root` privileges and that it not be confined by an AppArmor profile.

`aa-unconfined` must be run as `root` to retrieve the process executable link from the `/proc` file system. This program is susceptible to the following race conditions:

- An unlinked executable is mishandled
- A process that dies between `netstat(8)` and further checks is mishandled

ПРИМЕЧАНИЕ

This program lists processes using TCP and UDP only. In short, this program is unsuitable for forensics use and is provided only as an aid to profiling all network-accessible processes in the lab.

22.7 Important Filenames and Directories

The following list contains the most important files and directories used by the AppArmor framework. If you intend to manage and troubleshoot your profiles manually, make sure that you know about these files and directories:

`/sys/kernel/security/apparmor/profiles`

Virtualized file representing the currently loaded set of profiles.

`/etc/apparmor/`

Location of AppArmor configuration files.

`/etc/apparmor/profiles/extras/`

A local repository of profiles shipped with AppArmor, but not enabled by default.

`/etc/apparmor.d/`

Location of profiles, named with the convention of replacing the `/` in paths with `.` (not for the root `/`) so profiles are easier to manage. For example, the profile for the program `/usr/sbin/ntpd` is named `usr.sbin.ntpd`.

`/etc/apparmor.d/abstractions/`

Location of abstractions.

`/etc/apparmor.d/program-chunks/`

Location of program chunks.

`/proc/*/attr/current`

Check this file to review the confinement status of a process and the profile that is used to confine the process. The `ps auxZ` command retrieves this information automatically.

Profiling Your Web Applications Using ChangeHat

23

A AppArmor® profile represents the security policy for an individual program instance or process. It applies to an executable program, but if a portion of the program needs different access permissions than other portions, the program can «change hats» to use a different security context, distinctive from the access of the main program. This is known as a *hat* or *subprofile*.

ChangeHat enables programs to change to or from a *hat* within a AppArmor profile. It enables you to define security at a finer level than the process. This feature requires that each application be made «ChangeHat aware», meaning that it is modified to make a request to the AppArmor module to switch security domains at arbitrary times during the application execution. Two examples for ChangeHat-aware applications are the Apache Web server and Tomcat.

A profile can have an arbitrary number of subprofiles, but there are only two levels: a subprofile cannot have further sub-subprofiles. A subprofile is written as a separate profile and named as the containing profile followed by the subprofile name, separated by a `^`. Subprofiles must be stored in the same file as the parent profile.

Note that the security of hats is considerably weaker than that of full profiles. That is to say, if attackers can find just the right kind of bug in a program, they may be able to escape from a hat into the containing profile. This is because the security of hats is determined by a secret key handled by the containing process, and the code running in the hat must not have access to the key. Thus `change_hat` is most useful in conjunction with application servers, where a language interpreter (such as PERL, PHP, or Java) is isolating pieces of code such that they do not have direct access to the memory of the containing process.

The rest of this chapter describes using `change_hat` in conjunction with Apache, to contain web server components run using `mod_perl` and `mod_php`. Similar approaches can be used with any application server by providing an application module similar to the `mod_apparmor` described next in Раздел 23.2.2, «Location and Directory Directives» (стр. 313).

ПРИМЕЧАНИЕ: For More Information

For more information, see the `change_hat` man page.

23.1 Apache ChangeHat

AppArmor provides a `mod_apparmor` module (package `apache2-mod_apparmor`) for the Apache program. This module makes the Apache Web server ChangeHat aware. Install it along with Apache.

When Apache is ChangeHat aware, it checks for the following customized AppArmor security profiles in the order given for every URI request that it receives.

- URI-specific hat. For example, `^phpsysinfo/templates/classic/images/bar_left.gif`
- `DEFAULT_URI`
- `HANDLING_UNTRUSTED_INPUT`

ПРИМЕЧАНИЕ: Apache Configuration

If you install `apache2-mod_apparmor`, make sure the module gets loaded in Apache by executing the following command:

```
a2enmod apparmor
```

23.1.1 Managing ChangeHat-Aware Applications

As with most of the AppArmor tools, you can use two methods for managing ChangeHat, YaST or the command line interface. Managing ChangeHat-aware

applications from the command line is much more flexible, but the process is also more complicated. Both methods allow you to manage the hats for your application and populate them with profile entries.

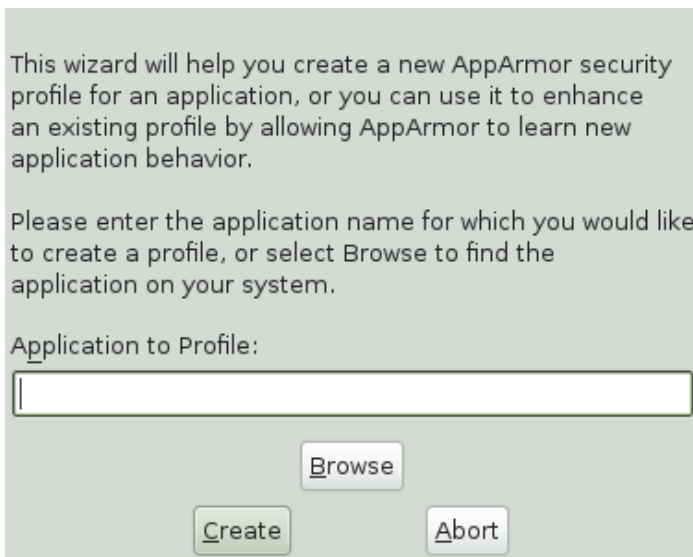
The following steps are a demonstration that adds hats to an Apache profile using YaST. In the *Add Profile Wizard*, the AppArmor profiling utilities prompt you to create new hats for distinct URI requests. Choosing to create a new hat allows you to create individual profiles for each URI. You can create very tight rules for each request.

If the URI that is processed does not represent significant processing or otherwise does not represent a significant security risk, safely select *Use Default Hat* to process this URI in the default hat, which is the default security profile.

This example creates a new hat for the URI `phpsysinfo` and its subsequent accesses. Using the profiling utilities, delegate what to add to this new hat. The resulting hat becomes a tight-security container that encompasses all the processing on the server that occurs when the `phpsysinfo` URI is passed to the Apache Web server.

The URI runs the application `phpsysinfo` (refer to <http://phpsysinfo.sourceforge.net> for more information). The `phpsysinfo` package is assumed to be installed in `/srv/www/htdocs/phpsysinfo` in a clean (new) installation of `PHP` and AppArmor.

- 1** Once `phpsysinfo` is installed, you are ready to add hats to the Apache profile. From the AppArmor GUI, select *Add Profile Wizard*.
- 2** In *Application to Profile*, enter `httpd2-prefork`.
- 3** Click *Create Profile*.



- 4 Restart Apache by entering `rcapache2 restart` in a terminal window.

Restart any program you are profiling at this point.

- 5 Open `http://localhost/phpsysinfo/` in a Web browser window. The browser window should display network usage and system information.

ПРИМЕЧАНИЕ: Data Caching

To ensure that this request is processed by the server and you do not review cached data in your browser, refresh the page. To do this, click the browser *Refresh* button to make sure that Apache processes the request for the `phpsysinfo` URI.

- 6 Click *Scan System Log for Entries to Add to Profiles*. AppArmor launches the `aa-logprof` tool, which scans the information learned in the previous step. It begins to prompt you with profile questions.
- 7 `aa-logprof` first prompts with *Add Requested Hat* or *Use Default Hat* because it noticed that the `phpsysinfo` URI was accessed. Select *Add Requested Hat*.
- 8 Click *Allow*.

Choosing *Add Requested Hat* in the previous step creates a new hat in the profile and specifies that the results of subsequent questions about the script's actions are added to the newly created hat rather than the default hat for this application.

In the next screen, AppArmor displays an external program that the script executed. You can specify that the program should run confined by the `phpsysinfo` hat (choose *Inherit*), confined by a separate profile (choose *Profile*), or that it should run unconfined or without any security profile (choose *Unconfined*). For the case of the *Profile* option, a new profile is created for the program if one does not already exist.

ПРИМЕЧАНИЕ: Security Considerations

Selecting *Unconfined* can create a significant security hole and should be done with caution.

- 8a** Select *Inherit* for the `/bin/bash` path. This adds `/bin/bash` (accessed by Apache) to the `phpsysinfo` hat profile with the necessary permissions.
- 8b** Click *Allow*.
- 9** The remaining questions prompt you to generate new hats and add entries to your profile and its hats. The process of adding entries to profiles is covered in detail in the Раздел 21.1, «Adding a Profile Using the Wizard» (срп. 261).

When all profiling questions are answered, click *Finish* to save your changes and exit the wizard.

The following is an example `phpsysinfo` hat.

Пример 23.1 Example `phpsysinfo` Hat

```
/usr/sbin/httpd2-prefork {  
...  
^phpsysinfo {  
    #include <abstractions/bash>  
    #include <abstractions/nameservice>  
  
    /bin/basename          ixr,  
    /bin/bash              ixr,  
    /bin/df                ixr,  
    /bin/grep              ixr,  
    /bin/mount             Ux,
```

```

/bin/sed                ixr,
/dev/bus/usb/           r,
/dev/bus/usb/**         r,
/dev/null               w,
/dev/tty                rw,
/dev/urandom            r,
/etc/SuSE-release       r,
/etc/ld.so.cache        r,
/etc/lsb-release        r,
/etc/lsb-release.d/     r,
/lib/ld-2.6.1.so        ixr,
/proc/**                r,
/sbin/lspci             ixr,
/srv/www/htdocs/phpsysinfo/** r,
/sys/bus/pci/**         r,
/sys/bus/scsi/devices/  r,
/sys/devices/**         r,
/usr/bin/cut            ixr,
/usr/bin/getopt         ixr,
/usr/bin/head           ixr,
/usr/bin/lsb_release    ixr,
/usr/bin/lsscsi         ixr,
/usr/bin/tr             ixr,
/usr/bin/who            ixr,
/usr/lib/lib*so*        mr,
/usr/lib/locale/**      r,
/usr/sbin/lusb          ixr,
/usr/share/locale/**    r,
/usr/share/pci.ids      r,
/usr/share/usb.ids      r,
/var/log/apache2/access_log w,
/var/run/utmp           kr,
}
}

```

ПРИМЕЧАНИЕ: Hat and Parent Profile Relationship

The profile `^phpsysinfo` is only valid in the context of a process running under the parent profile `httpd2-prefork`.

23.1.2 Adding Hats and Entries to Hats

When you use the *Edit Profile* dialog (for instructions, refer to Раздел 21.3, «Editing Profiles» (стр. 269)) or when you add a new profile using *Manually Add Profile* (for instructions, refer to Раздел 21.2, «Manually Adding a Profile» (стр. 268)), you are given the option of adding hats (subprofiles) to your AppArmor profiles. Add a ChangeHat subprofile from the *AppArmor Profile Dialog* window as in the following.



AppArmor Profile Dialogue

In this form you can view and modify the contents of an individual profile. [more](#)

AppArmor profile for /usr/sbin/httpd2-prefork

File Name	Permissions
[+] ^DEFAULT_URI	
[+] ^HANDLING_UNTRUSTED_INPUT	
#include abstractions/base	
#include abstractions/consoles	
#include abstractions/kerberosclient	
#include abstractions/nameservice	
#include abstractions/perl	
CAP_KILL	
CAP_NET_BIND_SERVICE	
CAP_SETGID	
CAP_SETUID	
CAP_SYS_TTY_CONFIG	
/dev/random	r
/etc/apache2/*conf	r
/etc/apache2/magic	r
/etc/apache2/mod_perl-startup.pl	r
/etc/apache2/ssl.crt/*:crt	r
/etc/apache2/ssl.key/*:key	r
/etc/apache2/{conf.sysconfig.vhosts}.d/	r
/etc/apache2/{conf.sysconfig.vhosts}.d/*	r

Add Entry ▼ Edit Entry Delete Entry

Help Abort Back Done

- 1 From the *AppArmor Profile Dialog* window, click *Add Entry* then select *Hat*. The *Enter Hat Name* dialog box opens:

Please enter the name of the Hat that you would like to add to the profile /usr/sbin/httpd2-prefork.

Hat name to add:

Create Hat Abort

- 2 Enter the name of the hat to add to the AppArmor profile. The name is the URI that, when accessed, receives the permissions set in the hat.
- 3 Click *Create Hat*. You are returned to the *AppArmor Profile Dialog* screen.
- 4 After adding the new hat, click *Done*.

ПРИМЕЧАНИЕ: For More Information

For an example of an AppArmor profile, refer to Пример 23.1, «Example phpsysinfo Hat» (стр. 309).

23.2 Configuring Apache for mod_apparmor

Apache is configured by placing directives in plain text configuration files. The main configuration file is usually `httpd.conf`. When you compile Apache, you can indicate the location of this file. Directives can be placed in any of these configuration files to alter the way Apache behaves. When you make changes to the main configuration files, you need to start or restart Apache, so the changes are recognized.

23.2.1 Virtual Host Directives

Virtual host directives control whether requests that contain trailing pathname information following an actual filename (or that refer to a nonexistent file in an existing directory) are accepted or rejected. For Apache documentation on virtual host directives, refer to <http://httpd.apache.org/docs/2.2/mod/core.html#virtualhost>.

The ChangeHat-specific configuration keyword is `AADefaultHatName`. It is used similarly to `AAHatName`, for example, `AADefaultHatName My_Funky_Default_Hat`.

The configuration option is actually based on a server directive, which enables you to use the keyword outside of other options, setting it for the default server. Virtual hosts are considered internally within Apache to be separate «servers,» so you can set a default hat name for the default server as well as one for each virtual host, if desired.

When a request comes in, the following steps reflect the sequence in which `mod_apparmor` attempts to apply hats.

1. A location or directory hat as specified by the `AAHatName` keyword
2. A hat named by the entire URI path

3. A default server hat as specified by the `AADefaultHatName` keyword
4. `DEFAULT_URI` (if none of those exist, it goes back to the «parent» Apache hat)

23.2.2 Location and Directory Directives

Location and directory directives specify hat names in the program configuration file so the program calls the hat regarding its security. For Apache, you can find documentation about the location and directory directives at <http://httpd.apache.org/docs/2.2/sections.html>.

The location directive example below specifies that, for a given location, `mod_apparmor` should use a specific hat:

```
<Location /foo/> AAHatName MY_HAT_NAME </Location>
```

This tries to use `MY_HAT_NAME` for any URI beginning with `/foo/` (`/foo/`, `/foo/bar`, `/foo/cgi/path/blah_blah/blah`, etc.).

The directory directive works similarly to the location directive, except it refers to a path in the file system as in the following example:

```
<Directory "/srv/www/www.immunix.com/docs">
  # Note lack of trailing slash
  AAHatName immunix.com
</Directory>
```

Example: The program `phpsysinfo` is used to illustrate a location directive in the following example. The tarball can be downloaded from <http://phpsysinfo.sourceforge.net>.

- 1 After downloading the tarball, install it into `/srv/www/htdocs/phpsysinfo`.
- 2 Create `/etc/apache2/conf.d/phpsysinfo.conf` and add the following text to it:

```
<Location "/phpsysinfo">
  AAHatName phpsysinfo
</Location>
```

The following hat should then work for `phpsysinfo`:

```
/usr/sbin/httpd2-prefork {
```

```

...
^phpsysinfo {
    #include <abstractions/bash>
    #include <abstractions/nameservice>

    /bin/basename                ixr,
    /bin/bash                    ixr,
    /bin/df                      ixr,
    /bin/grep                    ixr,
    /bin/mount                   Ux,
    /bin/sed                     ixr,
    /dev/bus/usb/                r,
    /dev/bus/usb/**              r,
    /dev/null                    w,
    /dev/tty                     rw,
    /dev/urandom                 r,
    /etc/SuSE-release            r,
    /etc/ld.so.cache             r,
    /etc/lsb-release             r,
    /etc/lsb-release.d/         r,
    /lib/ld-2.6.1.so            ixr,
    /proc/**                     r,
    /sbin/lspci                 ixr,
    /srv/www/htdocs/phpsysinfo/** r,
    /sys/bus/pci/**              r,
    /sys/bus/scsi/devices/       r,
    /sys/devices/**             r,
    /usr/bin/cut                 ixr,
    /usr/bin/getopt              ixr,
    /usr/bin/head                ixr,
    /usr/bin/lsb_release         ixr,
    /usr/bin/lsscsi              ixr,
    /usr/bin/tr                  ixr,
    /usr/bin/who                 ixr,
    /usr/lib/lib*so*             mr,
    /usr/lib/locale/**          r,
    /usr/sbin/lusb               ixr,
    /usr/share/locale/**        r,
    /usr/share/pci.ids           r,
    /usr/share/usb.ids           r,
    /var/log/apache2/access_log  w,
    /var/run/utmp                kr,
}
}

```

3 Reload AppArmor profiles by entering `rcapparmor restart` at a terminal window as `root`.

4 Restart Apache by entering `rcapache2 restart` at a terminal window as `root`.

- 5** Enter `http://hostname/phpsysinfo/` into a browser to receive the system information that `phpsysinfo` delivers.
- 6** Locate configuration errors by going to `/var/log/audit/audit.log` or running `dmesg` and looking for any rejections in the output.

Confining Users with `pam_apparmor`

An AppArmor profile applies to an executable program; if a portion of the program needs different access permissions than other portions need, the program can change hats via `change_hat` to a different role, also known as a subprofile. The `pam_apparmor` PAM module allows applications to confine authenticated users into subprofiles based on group names, user names, or a default profile. To accomplish this, `pam_apparmor` needs to be registered as a PAM session module.

The package `pam_apparmor` may not be installed by default, you may need to install it using YaST or `zypper`. Details about how to set up and configure `pam_apparmor` can be found in `/usr/share/doc/packages/pam_apparmor/README` after the package has been installed. For details on PAM, refer to Глава 2, *Автоматизация с помощью PAM* (стр. 19).

`pam_apparmor` allows you to set up role-based access control (RBAC). A detailed HOWTO on setting up RBAC with AppArmor is available at <http://wiki.apparmor.net/index.php/AppArmorRBAC>.

Managing Profiled Applications

25

After creating profiles and immunizing your applications, becomes more efficient and better protected as long as you perform AppArmor® profile maintenance (which involves analyzing log files, refining your profiles, backing up your set of profiles and keeping it up-to-date). You can deal with these issues before they become a problem by setting up event notification by e-mail, running periodic reports, updating profiles from system log entries by running the `aa-logprof` tool through YaST, and dealing with maintenance issues.

25.1 Reacting to Security Event Rejections

When you receive a security event rejection, examine the access violation and determine if that event indicated a threat or was part of normal application behavior. Application-specific knowledge is required to make the determination. If the rejected action is part of normal application behavior, run `aa-logprof` at the command line or the *Update Profile Wizard* in AppArmor to update your profile.

If the rejected action is not part of normal application behavior, this access should be considered a possible intrusion attempt (that was prevented) and this notification should be passed to the person responsible for security within your organization.

25.2 Maintaining Your Security Profiles

In a production environment, you should plan on maintaining profiles for all of the deployed applications. The security policies are an integral part of your deployment. You should plan on taking steps to back up and restore security policy files, plan for software changes, and allow any needed modification of security policies that your environment dictates.

25.2.1 Backing Up Your Security Profiles

Backing up profiles might save you from having to reprofile all your programs after a disk crash. Also, if profiles are changed, you can easily restore previous settings by using the backed up files.

Back up profiles by copying the profile files to a specified directory.

- 1 You should first archive the files into one file. To do this, open a terminal window and enter the following as `root`:

```
tar zclpf profiles.tgz /etc/apparmor.d
```

The simplest method to ensure that your security policy files are regularly backed up is to include the directory `/etc/apparmor.d` in the list of directories that your backup system archives.

- 2 You can also use `scp` or a file manager like Konqueror or Nautilus to store the files on some kind of storage media, the network, or another computer.

25.2.2 Changing Your Security Profiles

Maintenance of security profiles includes changing them if you decide that your system requires more or less security for its applications. To change your profiles in AppArmor, refer to Раздел 21.3, «Editing Profiles» (стр. 269).

25.2.3 Introducing New Software into Your Environment

When you add a new application version or patch to your system, you should always update the profile to fit your needs. You have several options, depending on your company's software deployment strategy. You can deploy your patches and upgrades into a test or production environment. The following explains how to do this with each method.

If you intend to deploy a patch or upgrade in a test environment, the best method for updating your profiles is one of the following:

- Run the profiling wizard by selecting *Add Profile Wizard* in YaST. This creates a new profile for the added or patched application. For step-by-step instructions, refer to Раздел 21.1, «Adding a Profile Using the Wizard» (стр. 261).
- Run `aa-genprof` by typing `aa-genprof` in a terminal while logged in as `root`. For detailed instructions, refer to Раздел 22.6.3.4, «aa-genprof—Generating Profiles» (стр. 289).

If you intend to deploy a patch or upgrade directly into a production environment, the best method for updating your profiles is one of the following:

- Monitor the system frequently to determine if any new rejections should be added to the profile and update as needed using `aa-logprof`. For detailed instructions, refer to Раздел 22.6.3.5, «aa-logprof—Scanning the System Log» (стр. 297).
- Run the YaST *Update Profile Wizard* to learn the new behavior (high security risk as all accesses are allowed and logged, not rejected). For step-by-step instructions, refer to Раздел 21.5, «Updating Profiles from Log Entries» (стр. 275).

Support

This chapter outlines maintenance-related tasks. Learn how to update AppArmor® and get a list of available man pages providing basic help for using the command line tools provided by AppArmor. Use the troubleshooting section to learn about some common problems encountered with AppArmor and their solutions. Report defects or enhancement requests for AppArmor following the instructions in this chapter.

26.1 Updating AppArmor Online

Updates for AppArmor packages are provided in the same way as any other update for . Retrieve and apply them exactly like for any other package that ships as part of .

26.2 Using the Man Pages

There are man pages available for your use. In a terminal, enter `man apparmor` to open the apparmor man page. Man pages are distributed in sections numbered 1 through 8. Each section is specific to a category of documentation:

Таблица 26.1 *Man Pages: Sections and Categories*

Section	Category
1	User commands

Section	Category
2	System calls
3	Library functions
4	Device driver information
5	Configuration file formats
6	Games
7	High level concepts
8	Administrator commands

The section numbers are used to distinguish man pages from each other. For example, `exit(2)` describes the `exit` system call, while `exit(3)` describes the `exit` C library function.

The AppArmor man pages are:

- `unconfined(8)`
- `autodep(1)`
- `complain(1)`
- `enforce(1)`
- `genprof(1)`
- `logprof(1)`
- `change_hat(2)`
- `logprof.conf(5)`
- `apparmor.conf(5)`
- `apparmor.d(5)`

- `apparmor.vim(5)`
- `apparmor(7)`
- `apparmor_parser(8)`

26.3 For More Information

Find more information about the AppArmor product at: <http://wiki.apparmor.net>. Find the product documentation for AppArmor in the installed system at `/usr/share/doc/manual`.

There is a mailing lists for AppArmor that users can post to or join to communicate with developers. See <https://lists.ubuntu.com/mailman/listinfo/apparmor> for details.

26.4 Troubleshooting

This section lists the most common problems and error messages that may occur using AppArmor.

26.4.1 How to React to odd Application Behavior?

If you notice odd application behavior or any other type of application problem, you should first check the reject messages in the log files to see if AppArmor is too closely constricting your application. If you detect reject messages that indicate that your application or service is too closely restricted by AppArmor, update your profile to properly handle your use case of the application. Do this with the *Update Profile Wizard* in YaST, as described in Раздел 21.5, «Updating Profiles from Log Entries» (стр. 275).

If you decide to run your application or service without AppArmor protection, remove the application's profile from `/etc/apparmor.d` or move it to another location.

26.4.2 My Profiles do not Seem to Work Anymore ...

If you have been using previous versions of AppArmor and have updated your system (but kept your old set of profiles) you might notice some applications which seemed to work perfectly before you updated behaving strangely, or not working at all .

This version of AppArmor introduces a set of new features to the profile syntax and the AppArmor tools that might cause trouble with older versions of the AppArmor profiles. Those features are:

- File Locking
- Network Access Control
- The `SYS_PTRACE` Capability
- Directory Path Access

The current version of AppArmor mediates file locking and introduces a new permission mode (`k`) for this. Applications requesting file locking permission might misbehave or fail altogether if confined by older profiles which do not explicitly contain permissions to lock files. If you suspect this being the case, check the log file under `/var/log/audit/audit.log` for entries like the following:

```
type=APPARMOR_DENIED msg=audit(1188913493.299:9304): operation="file_lock"
requested_mask="::k" denied_mask="::k" fsuid=1000 name="/home/
tux/.qt/.qtrc.lock" pid=25736 profile="/usr/bin/opera"
```

Update the profile using the YaST Update Profile Wizard or the `aa-logprof` command as outlined below.

The new network access control syntax based on the network family and type specification, described in Раздел 19.5, «Network Access Control» (стр. 243), might cause application misbehavior or even stop applications from working. If you notice a network-related application behaving strangely, check the log file under `/var/log/audit/audit.log` for entries like the following:

```
type=APPARMOR_DENIED msg=audit(1188894313.206:9123):
operation="socket_create" family="inet" sock_type="raw" protocol=1
pid=23810 profile="/bin/ping"
```

This log entry means that our example application, `/bin/ping` in this case, failed to get AppArmor's permission to open a network connection. This permission has to be explicitly stated to make sure that an application has network access. To update the profile to the new syntax, use the YaST Update Profile Wizard or the `aa-logprof` command as outlined below.

The current kernel requires the `SYS_PTRACE` capability, if a process tries to access files in `/proc/pid/fd/*`. New profiles need an entry for the file and the capability, where old profiles only needed the file entry. For example:

```
/proc/*/fd/**  rw,
```

in the old syntax would translate to the following rules in the new syntax:

```
capability SYS_PTRACE,  
/proc/*/fd/**  rw,
```

To update the profile to the new syntax, use the YaST Update Profile Wizard or the `aa-logprof` command as outlined below.

With this version of AppArmor, a few changes have been made to the profile rule syntax to better distinguish directory from file access. Therefore, some rules matching both file and directory paths in the previous version might now just match a file path. This could lead to AppArmor not being able to access a crucial directory at all, and thus trigger misbehavior of your application and various log messages. The following examples highlight the most important changes to the path syntax.

Using the old syntax, the following rule would allow access to files and directories in `/proc/net`. It would allow directory access only to read the entries in the directory, but not give access to files or directories under the directory, e.g. `/proc/net/dir/foo` would be matched by the asterisk (*), but as `foo` is a file or directory under `dir`, it cannot be accessed.

```
/proc/net/*  r,
```

To get the same behavior using the new syntax, you need two rules instead of one. The first allows access to the file under `/proc/net` and the second allows access to directories under `/proc/net`. Directory access can only be used for listing the contents, not actually accessing files or directories underneath the directory.

```
/proc/net/*  r,  
/proc/net/*/  r,
```

The following rule works similarly both under the old and the new syntax, and allows access to both files and directories under `/proc/net`:

```
/proc/net/** r,
```

To distinguish file access from directory access using the above expression in the new syntax, use the following two rules. The first one only allows to recursively access directories under `/proc/net` while the second one explicitly allows for recursive file access only.

```
/proc/net/**/ r,  
/proc/net/**[^/] r,
```

The following rule works similarly both under the old and the new syntax and allows access to both files and directories beginning with `foo` under `/proc/net`:

```
/proc/net/foo** r,
```

To distinguish file access from directory access in the new syntax and use the `**` globbing pattern, use the following two rules. The first one would have matched both files and directories in the old syntax, but only matches files in the new syntax due to the missing trailing slash. The second rule matched neither file nor directory in the old syntax, but matches directories only in the new syntax:

```
/proc/net/**foo r,  
/proc/net/**foo/ r,
```

The following rules illustrate how the use of the `?` globbing pattern has changed. In the old syntax, the first rule would have matched both files and directories (four characters, last character could be any but a slash). In the new syntax, it matches only files (trailing slash is missing). The second rule would match nothing in the old profile syntax, but matches directories only in the new syntax. The last rule matches explicitly matches a file called `bar` under `/proc/net/foo?`. Using the old syntax, this rule would have applied to both files and directories:

```
/proc/net/foo? r,  
/proc/net/foo?/ r,  
/proc/net/foo?/bar r,
```

To find and resolve issues related to syntax changes, take some time after the update to check the profiles you want to keep and proceed as follows for each application you kept the profile for:

- 1 Make sure that AppArmor is running and that the application's profile is loaded.
- 2 Start the YaST AppArmor Control Panel and put the application's profile into complain mode. Log entries are made for any actions violating the current profile, but the profile is not enforced and the application's behavior not restricted.

- 3 Run the application covering all the tasks you need this application to be able to perform.
- 4 Start the YaST Update Profile Wizard to update the application's profile according to the log entries generated while running the application.
- 5 Once the profile is updated, put it back into enforce mode via the YaST AppArmor Control Panel.

Using the AppArmor command line tools, you would proceed as follows:

- 1 Put the application's profile into complain mode:

```
aa-complain /path/to/application
```

- 2 Run the application.
- 3 Update the profile according to the log entries made while running the application:

```
aa-logprof /path/to/application
```

- 4 Put the resulting profile back into enforce mode:

```
aa-enforce /path/to/application
```

26.4.3 How to Confine KDE Applications with AppArmor?

Currently, it is not possible to confine KDE applications to the same extent as any other application, due to the way KDE manages its processes.

If you want to confine KDE applications, choose one of the following approaches, but note that none of them are really suited for a standard setup:

Create a Single Profile for the Entire KDE Desktop

As all KDE processes are children of one parent process and AppArmor cannot distinguish an individual application's process from the rest, create one huge profile to confine the entire desktop all at once. This approach is only feasible if your setup is a very limited (kiosk-type) one. Maintaining such a profile for

a standard KDE desktop (including all of its applications) would be close to impossible.

Modify KDE's process handling

Using `KDE_EXEC_SLAVES=1` and `KDE_IS_PRELINKED=1` variables force KDE to manage its processes in a way that allows AppArmor to distinguish individual applications from each other and apply profiles to them. This approach might slow down your desktop considerably, as it turns off a crucial optimization for speed. Note that the above mentioned environment variables have to be set before KDM/XDM/GDM or startx are started. One way to achieve this would be to add them to `/etc/security/pam_env.conf`.

26.4.4 How to Resolve Issues with Apache?

Apache is not starting properly or it is not serving Web pages and you just installed a new module or made a configuration change. When you install additional Apache modules (like `apache2-mod_apparmor`) or make configuration changes to Apache, you should profile Apache again to catch any additional rules that need to be added to the profile.

26.4.5 How to Exclude Certain Profiles from the List of Profiles Used?

Run `aa-disable PROGRAMNAME` to disable the profile for *PROGRAMNAME*. This command creates a symbolic link to the profile in `/etc/apparmor.d/disable/`. In order to reactivate the profile, just delete that link.

26.4.6 Can I Manage Profiles for Applications not Installed on my System?

Managing profiles with AppArmor requires you to have access to the log of the system on which the application is running. So you do not need to run the application on your profile, build host as long as you have access to the machine that runs the application. You can run the application on one system, transfer the logs (`/var/`

log/audit.log or, if audit is not installed, /var/log/messages) to your profile build host and run `aa-logprof -f path_to_logfile`.

26.4.7 How to Spot and fix AppArmor Syntax Errors?

Manually editing AppArmor profiles can introduce syntax errors. If you attempt to start or restart AppArmor with syntax errors in your profiles, error results are shown. This example shows the syntax of the entire parser error.

```
localhost:~ # rcapparmor start
Loading AppArmor profiles AppArmor parser error in /etc/apparmor.d/
usr.sbin.squid at line 410: syntax error, unexpected TOK_ID, expecting
TOK_MODE
Profile /etc/apparmor.d/usr.sbin.squid failed to load
```

Using the AppArmor YaST tools, a graphical error message indicates which profile contained the error and requests you to fix it.

Errors found in AppArmor profiles

These problems must be corrected before AppArmor can be started or the profile management tools can be used.

- /etc/apparmor.d/usr.sbin.traceroute contains syntax errors.
Line [foo]

You can find a description of AppArmor profile syntax by running `man apparmor.d`

Comprehensive documentation about AppArmor is available in the Administration guide. This is available in the directory:

/usr/share/doc/manual/suselinux-manual_LANGUAGE.

Please refer to this for more detailed information about AppArmor



To fix a syntax error, log in to a terminal window as `root`, open the profile, and correct the syntax. Reload the profile set with `rcapparmor reload`.

ПОДСКАЗКА: AppArmor Syntax Highlighting in `vi`

The editor `vi` on supports syntax highlighting for AppArmor profiles. Lines containing syntax errors will be displayed with a red background.

26.5 Reporting Bugs for AppArmor

The developers of AppArmor are eager to deliver products of the highest quality. Your feedback and your bug reports help us keep the quality high. Whenever you encounter a bug in AppArmor, file a bug report against this product:

1 Use your Web browser to go to <https://bugzilla.novell.com/index.cgi>.

2 Enter the account data of your Novell account and click *Login*

or

Create a new Novell account as follows:

2a Click *Create New Account* on the *Login to Continue* page.

2b Provide a username and password and additional address data and click *Create Login* to immediately proceed with the login creation.

or

Provide data on which other Novell accounts you maintain to sync all these to one account.

3 Check whether a problem similar to yours has already been reported by clicking *Search Reports*. Use a quick search against a given product and keyword or use the *Advanced Search*.

4 If your problem has already been reported, check this bug report and add extra information to it, if necessary.

5 If your problem has not been reported yet, select *New* from the top navigation bar and proceed to the *Enter Bug* page.

6 Select the product against which to file the bug. In your case, this would be your product's release. Click *Submit*.

7 Select the product version, component (AppArmor in this case), hardware platform, and severity.

- 8** Enter a brief headline describing your problem and add a more elaborate description including log files. You may create attachments to your bug report for screen shots, log files, or test cases.
- 9** Click *Submit* after you have entered all the details to send your report to the developers.

AppArmor Glossary

Apache

Apache is a freely-available UNIX-based Web server. It is currently the most commonly used Web server on the Internet. Find more information about Apache at the Apache Web site at <http://www.apache.org>.

application firewalling

AppArmor contains applications and limits the actions they are permitted to take. It uses privilege confinement to prevent attackers from using malicious programs on the protected server and even using trusted applications in unintended ways.

attack signature

Pattern in system or network activity that alerts of a possible virus or hacker attack. Intrusion detection systems might use attack signatures to distinguish between legitimate and potentially malicious activity.

By not relying on attack signatures, AppArmor provides "proactive" instead of "reactive" defense from attacks. This is better because there is no window of vulnerability where the attack signature must be defined for AppArmor as it does for products using attack signatures to secure their networks.

GUI

Graphical user interface. Refers to a software front-end meant to provide an attractive and easy-to-use interface between a computer user and application. Its elements include such things as windows, icons, buttons, cursors, and scroll bars.

globbing

Filename substitution.

HIP

Host intrusion prevention. Works with the operating system kernel to block abnormal application behavior in the expectation that the abnormal behavior represents an unknown attack. Blocks malicious packets on the host at the network level before they can «hurt» the application they target.

mandatory access control

A means of restricting access to objects that is based on fixed security attributes assigned to users, files, and other objects. The controls are mandatory in the sense that they cannot be modified by users or their programs.

profile foundation classes

Profile building blocks needed for common application activities, such as DNS lookup and user authentication.

RPM

The RPM Package Manager. An open packaging system available for anyone to use. It works on Red Hat Linux, , and other Linux and UNIX systems. It is capable of installing, uninstalling, verifying, querying, and updating computer software packages. See <http://www.rpm.org/> for more information.

SSH

Secure Shell. A service that allows you to access your server from a remote computer and issue text commands through a secure connection.

streamlined access control

AppArmor provides streamlined access control for network services by specifying which files each program is allowed to read, write, and execute. This ensures that each program does what it is supposed to do and nothing else.

URI

Universal resource identifier. The generic term for all types of names and addresses that refer to objects on the World Wide Web. A URL is one kind of URI.

URL

Uniform Resource Locator. The global address of documents and other resources on the World Wide Web.

The first part of the address indicates what protocol to use and the second part specifies the IP address or the domain name where the resource is located.

For example, in `http://www.novell.com`, `http` is the protocol to use.

vulnerabilities

An aspect of a system or network that leaves it open to attack. Characteristics of computer systems that allow an individual to keep it from correctly operating or that allows unauthorized users to take control of the system. Design, administrative, or implementation weaknesses or flaws in hardware, firmware, or software. If exploited, a vulnerability could lead to an unacceptable impact in the form of unauthorized access to information or the disruption of critical processing.



Лицензии GNU

Это приложение содержит GNU General Public License версии 2 и GNU Free Documentation License версии 1.2.

Универсальная Общественная Лицензия GNU (GNU General Public License)

Версия 2, июнь 1991 г.

Copyright (C) 1989, 1991 Free Software Foundation, Inc. 59 Temple Place - Suite 330, Boston, MA 02111-1307, USA

(C) Перевод. О.В. Кузина, В.М. Юфа, 1993 (C) Перевод. О.С. Тихонов, 1998

Этот документ можно копировать, а также распространять его дословные копии, однако вносить в него изменения запрещено.

Преамбула

Лицензии на большую часть программного обеспечения (ПО) составлены так, чтобы лишить вас свободы совместно использовать и изменять его. В противоположность этому, предназначение Универсальной Общественной Лицензии GNU состоит в том, чтобы гарантировать вашу свободу совместно использовать и изменять свободное ПО, т.е. обеспечить свободу ПО для всех его пользователей. Данная Универсальная Общественная Лицензия применима к большей части ПО Фонда Свободного ПО и ко всем другим программам, чьи авторы принимают на себя обязательство ее использовать. (Для некоторых программ Фонда Свободного ПО вместо нее применяется Универсальная Общественная Лицензия GNU для библиотек.) Вы тоже можете применить ее к своим программам.

Когда мы говорим о свободном ПО, мы имеем в виду свободу, а не бесплатность. Наши Универсальные Общественные Лицензии разрабатывались для того, чтобы гарантировать, что вы пользуетесь свободой распространять копии свободного ПО (и при желании получать за это вознаграждение); что вы получаете исходный код или можете получить его, если захотите; что вы можете изменять ПО или использовать его части в новых свободных программах; и что вы знаете обо всех этих правах.

Чтобы защитить ваши права, нам нужно ввести некоторые ограничения, которые запретят кому бы то ни было отказывать вам в этих правах или потребовать от вас отказаться от этих прав. Эти ограничения накладывают на вас некоторые обязательства, если вы распространяете копии ПО или изменяете его.

Например, если вы распространяете копии такой программы бесплатно или за вознаграждение, вы должны предоставить получателям все права, которыми обладаете вы сами. Вы должны гарантировать, что они тоже получат или смогут получить исходный код. Наконец, вы должны показать им текст данных условий, чтобы они знали о своих правах.

Мы защищаем ваши права в два этапа: (1) сохраняем авторские права на ПО и (2) предлагаем вам эту лицензию, которая дает вам законное право копировать, распространять и/или модифицировать ПО.

Кроме того, в целях защиты как каждого автора, так и нас, мы хотим удостовериться, что каждый понимает, что гарантий на это свободное ПО нет. Если ПО модифицируется и передается кем-то еще, мы хотим, чтобы получатели ПО знали, что то, что у них есть, — это не оригинал, чтобы любые проблемы, созданные другими, не отразились на репутации первоначальных авторов.

И наконец, каждой свободной программе постоянно угрожают патенты на ПО. Мы хотим избежать той опасности, что повторные распространители свободной программы самостоятельно получают патенты, делая программу таким образом частной собственностью. Чтобы предотвратить это, мы со всей определенностью заявляем, что любой патент должен быть либо предоставлен всем для свободного использования, либо не предоставлен никому.

Ниже следуют точные определения и условия для копирования, распространения и модификации.

ОПРЕДЕЛЕНИЯ И УСЛОВИЯ ДЛЯ КОПИРОВАНИЯ, РАСПРОСТРАНЕНИЯ И МОДИФИКАЦИИ

0. Эта Лицензия применима к любой программе или другому произведению, содержащему уведомление, помещенное держателем авторских прав и сообщающее о том, что оно может распространяться при условиях, оговоренных в данной Универсальной Общественной Лицензии. В дальнейшем термин «Программа» относится к любой такой программе или произведению, а термин «произведение, основанное на Программе» означает Программу или любое произведение, содержащее Программу или ее часть, дословную, или модифицированную, и/или переведенную на другой язык. (Здесь и далее перевод включается без ограничений в понятие «модификация».) Каждый обладатель лицензии адресуется как «вы».

Виды деятельности, не являющиеся копированием, распространением или модификацией, не охватываются данной Лицензией; они лежат за пределами ее влияния. Использование Программы по ее функциональному назначению не ограничено, а выходные данные Программы охватываются этой Лицензией, только если их содержание является произведением, основанным на Программе (вне зависимости от того, были ли они получены в процессе использования Программы). Являются ли они таковыми, зависит от того, что именно делает Программа.

1. Вы можете копировать и распространять дословные копии исходного кода Программы по его получению на любом носителе, при условии что вы соответствующим образом помещаете на видном месте в каждой копии соответствующее уведомление об авторских правах и отказ от предоставления гарантий; оставляете нетронутыми все уведомления, относящиеся к данной Лицензии и к отсутствию каких-либо гарантий; и передаете всем другим получателям Программы копию данной Лицензии вместе с Программой.

Вы можете назначить плату за физический акт передачи копии и можете по своему усмотрению предоставлять гарантии за вознаграждение.

2. Вы можете изменять свою копию или копии Программы или любой ее части, создавая таким образом произведение, основанное на Программе, и копировать и распространять эти модификации или произведение в соответствии с Разделом 1, приведенным выше, при условии, что вы выполните все нижеследующие условия:

- a)** Вы обязаны снабдить модифицированные файлы заметными уведомлениями, содержащими указания на то, что вы изменили файлы, и дату каждого изменения.
- b)** Вы обязаны предоставить всем третьим лицам лицензию на бесплатное использование каждого произведения, которое вы распространяете или публикуете, целиком, и которое полностью или частично содержит Программу или какую-либо ее часть, на условиях, оговоренных в данной Лицензии.
- c)** Если модифицированная программа обычно читает команды в интерактивном режиме работы, вы должны сделать так, чтобы при запуске для работы в таком интерактивном режиме обычным для нее способом она печатала или выводила на экран объявление, содержащее соответствующее уведомление об авторских правах и уведомление о том, что гарантий нет (или, наоборот, сообщающее о том, что вы обеспечиваете гарантии), и что пользователи могут повторно распространять программу при этих условиях, и указывающее пользователю, как просмотреть копию данной Лицензии. (Исключение: если сама Программа работает в интерактивном режиме, но обычно не выводит подобных сообщений, то ваше произведение, основанное на Программе, не обязано выводить объявление.)

Эти требования применяются к модифицированному произведению в целом. Если известные части этого произведения не были основаны на Программе и могут обоснованно считаться независимыми и самостоятельными произведениями, то эта Лицензия и ее условия не распространяются на эти части, если вы распространяете их как отдельные произведения. Но если вы распространяете эти части как часть целого произведения, основанного на Программе, то вы обязаны делать это в соответствии с условиями данной Лицензии, распространяя права получателей лицензии на все произведение и, таким образом, на каждую часть, вне зависимости от того, кто ее написал.

Таким образом, содержание этого раздела не имеет цели претендовать на ваши права на произведение, написанное полностью вами, или оспаривать их; цель скорее в том, чтобы реализовать право управлять распространением производных или коллективных произведений, основанных на Программе.

Кроме того, простое нахождение другого произведения, не основанного на этой Программе, совместно с Программой (или с произведением, основанным на этой Программе) на одном носителе для постоянного хранения или распространяемом носителе не распространяет действие этой Лицензии на другое произведение.

3. Вы можете копировать и распространять Программу (или произведение, основанное на ней) согласно Разделу 2) в объектном коде или в выполняемом виде в соответствии с Разделами 1 и 2, приведенными выше, при условии, что вы также выполните одно из следующих требований:

- a)** Сопроводите ее полным соответствующим машиночитаемым кодом, который должен распространяться в соответствии с Разделами 1 и 2, приведенными выше, на носителе, который обычно используется для обмена ПО; или,
- b)** Сопроводите ее письменным предложением, действительным по крайней мере в течение трех лет, предоставить любому третьему лицу за вознаграждение, не превышающее стоимость физического акта изготовления копии, полную машиночитаемую копию соответствующего исходного кода, подлежащую распространению в соответствии с Разделами 1 и 2, приведенными выше; или
- c)** Сопроводите ее информацией, полученной вами в качестве предложения распространить соответствующий исходный код. (Эта возможность допустима только для некоммерческого распространения, и только если вы получили программу в объектном коде или в исполняемом виде с предложением в соответствии с Пунктом b) выше.)

Исходный код для произведения означает его вид, предпочтительный для выполнения в нем модификаций. Для исполняемого произведения полный исходный код означает все исходные коды для всех модулей, которые он содержит, плюс любые связанные с произведением файлы определения интерфейса, плюс сценарии, используемые для управления компиляцией и установкой исполняемого произведения. Однако, в виде особого исключения распространяемый исходный код не обязан включать то, что обычно предоставляется (как в объектных, так и в исходных кодах) с основными компонентами (компилятор, ядро и так далее) операционной системы, под управлением которой работает исполняемое произведение, за исключением случая, когда сам компонент сопровождает исполняемое произведение.

Если распространение исполняемого произведения или объектного кода происходит путем предоставления доступа для копирования с обозначенного места, то предоставление доступа для копирования исходного кода с того же места считается распространением исходного кода, даже если третьи лица не принуждаются к копированию исходного кода вместе с объектным кодом.

4. Вы не можете копировать, изменять, повторно лицензировать, или распространять Программу никаким иным способом, кроме явно предусмотренных данной Лицензией. Любая попытка копировать, изменять или распространять Программу каким-либо другим способом или с измененной лицензией неправомерна и автоматически прекращает ваши права, данные вам этой Лицензией. Однако лицензии лиц, получивших от вас копии или права согласно данной Универсальной Общественной Лицензии, не прекращают своего действия, если эти лица полностью соблюдают условия.

5. Вы не обязаны соглашаться с этой Лицензией, так как вы не подписывали ее. Однако, ничто, кроме этой Лицензии, не дает вам право изменять или распространять эту Программу или основанные на ней произведения. Эти действия запрещены законом, если вы не принимаете к соблюдению эту Лицензию. А значит, изменяя или распространяя Программу (или произведение, основанное на Программе), вы извещаете свое согласие с этой Лицензией и всеми ее условиями о копировании, распространении или модификации Программы или основанных на ней произведений.

6. Каждый раз, когда вы повторно распространяете Программу (или любое произведение, основанное на Программе), получатель этого произведения автоматически получает от первоначального выдавшего лицензию лица свою лицензию на копирование, распространение или модификацию Программы, обсуждаемую в этих определениях и условиях. Вы не можете налагать каких-либо дополнительных ограничений на осуществление получателем прав, предоставленных данным документом. Вы не несете ответственности за соблюдение третьими лицами условий этой Лицензии.

7. Если в результате судебного разбирательства, или обвинения в нарушении патента или по любой другой причине (не обязательно связанной с патентами), вам навязаны условия, противоречащие данной Лицензии (по постановлению суда, по соглашению или иным способом), это не освобождает вас от соблюдения Лицензии. Если вы не можете заниматься распространением так, чтобы одновременно удовлетворить требованиям и этой Лицензии, и всем другим требованиям, то вы не должны заниматься распространением Программы. Например, если патент не позволяет безвозмездное повторное распространение Программы всем, кто получил копии от вас непосредственно или через посредников, то единственным способом удовлетворить и патенту, и этой Лицензии будет ваш полный отказ от распространения Программы.

Если какая-либо часть этого раздела не имеет силы или не может быть исполнена при некоторых конкретных обстоятельствах, то подразумевается, что имеет силу остальная часть раздела, а при других обстоятельствах имеет силу весь Раздел.

Цель этого раздела — не побудить вас делать заявления о нарушениях прав на патент, или заявлять о других претензиях на право собственности или оспаривать правильность подобных претензий; единственная цель этого раздела — защита целостности системы распространения свободного ПО, которая реализуется использованием общественных лицензий. Многие люди внесли щедрый вклад в широкий спектр ПО, распространяемого по этой системе, полагаясь на ее согласованное применение; только автору принадлежит право решать, хочет ли он или она распространять ПО в этой системе или в какой-то другой, и получатель лицензии не может влиять на принятие этого решения.

Этот раздел предназначен для того, чтобы тщательно прояснить, что полагается следствием из остальной части данной Лицензии.

8. Если распространение и/или применение Программы ограничено в ряде стран либо патентами, либо авторскими правами на интерфейсы, первоначальный обладатель авторских прав, выпускающий Программу с этой Лицензией, может добавить явное ограничение на географическое распространение, исключив такие страны, так что распространение разрешается только в тех странах, которые не были исключены. В этом случае данная Лицензия включает в себя это ограничение, как если бы оно было написано в тексте данной Лицензии.

9. Фонд Свободного ПО может время от времени публиковать пересмотренные и/или новые версии Универсальной Общественной Лицензии. Такие новые версии будут сходны по духу с настоящей версией, но могут отличаться в деталях, направленных на новые проблемы или обстоятельства.

Каждой версии придается отличительный номер. Если в Программе указывается, что к ней относится некоторый номер версии данной Лицензии и «любая последующая версия», вы можете по выбору следовать определениям и условиям либо данной версии, либо любой последующей версии, опубликованной Фондом Свободного ПО. Если в Программе не указан номер версии данной Лицензии, вы можете выбрать любую версию, когда-либо опубликованную Фондом Свободного ПО.

10. Если вы хотите встроить части Программы в другие свободные программы с иными условиями распространения, напишите автору с просьбой о разрешении. Для ПО, которое охраняется авторскими правами Фонда Свободного ПО, напишите в Фонд Свободного ПО; мы иногда делаем такие исключения. Наше решение будет руководствоваться двумя целями: сохранения свободного статуса всех производных нашего свободного ПО и содействия совместному и повторному использованию ПО вообще.

НИКАКИХ ГАРАНТИЙ

11. ПОСКОЛЬКУ ПРОГРАММА ПРЕДОСТАВЛЯЕТСЯ БЕСПЛАТНО, НА ПРОГРАММУ НЕТ ГАРАНТИЙ В ТОЙ МЕРЕ, КАКАЯ ДОПУСТИМА ПРИМЕНИМЫМ ЗАКОНОМ. ЗА ИСКЛЮЧЕНИЕМ ТЕХ СЛУЧАЕВ, КОГДА ОБРАТНОЕ ЗАЯВЛЕНО В ПИСЬМЕННОЙ ФОРМЕ, ДЕРЖАТЕЛИ АВТОРСКИХ ПРАВ И/ЛИ ДРУГИЕ СТОРОНЫ ПОСТАВЛЯЮТ ПРОГРАММУ “КАК ОНА ЕСТЬ” БЕЗ КАКОГО-ЛИБО ВИДА ГАРАНТИЙ, ВЫРАЖЕННЫХ ЯВНО ИЛИ ПОДРАЗУМЕВАЕМЫХ, ВКЛЮЧАЯ, НО НЕ ОГРАНИЧИВАЯСЬ ИМИ,

ПОДРАЗУМЕВАЕМЫЕ ГАРАНТИИ КОММЕРЧЕСКОЙ ЦЕННОСТИ И ПРИГОДНОСТИ ДЛЯ КОНКРЕТНОЙ ЦЕЛИ. ВЕСЬ РИСК В ОТНОШЕНИИ КАЧЕСТВА И ПРОИЗВОДИТЕЛЬНОСТИ ПРОГРАММЫ ОСТАЕТСЯ ПРИ ВАС. ЕСЛИ ПРОГРАММА ОКАЖЕТСЯ ДЕФЕКТНОЙ, ВЫ ПРИНИМАЕТЕ НА СЕБЯ СТОИМОСТЬ ВСЕГО НЕОБХОДИМОГО ОБСЛУЖИВАНИЯ, ВОССТАНОВЛЕНИЯ ИЛИ ИСПРАВЛЕНИЯ.

12. НИ В КОЕМ СЛУЧАЕ, ЕСЛИ НЕ ТРЕБУЕТСЯ СООТВЕТСТВУЮЩИМ ЗАКОНОМ, ИЛИ НЕ УСЛОВЛЕНО В ПИСЬМЕННОЙ ФОРМЕ, НИ ОДИН ДЕРЖАТЕЛЬ АВТОРСКИХ ПРАВ И НИ ОДНО ДРУГОЕ ЛИЦО, КОТОРОЕ МОЖЕТ ИЗМЕНЯТЬ И/ИЛИ ПОВТОРНО РАСПРОСТРАНЯТЬ ПРОГРАММУ, КАК БЫЛО РАЗРЕШЕНО ВЫШЕ, НЕ ОТВЕТСТВЕННЫ ПЕРЕД ВАМИ ЗА УБЫТКИ, ВКЛЮЧАЯ ЛЮБЫЕ ОБЩИЕ, СПЕЦИАЛЬНЫЕ, СЛУЧАЙНЫЕ ИЛИ ПОСЛЕДОВАВШИЕ УБЫТКИ, ПРОИСТЕКАЮЩИЕ ИЗ ИСПОЛЬЗОВАНИЯ ИЛИ НЕВОЗМОЖНОСТИ ИСПОЛЬЗОВАНИЯ ПРОГРАММЫ (ВКЛЮЧАЯ, НО НЕ ОГРАНИЧИВАЯСЬ ПОТЕРЕЙ ДАННЫХ, ИЛИ ДАННЫМИ, СТАВШИМИ НЕПРАВИЛЬНЫМИ, ИЛИ ПОТЕРЯМИ, ПОНЕСЕННЫМИ ИЗ-ЗА ВАС ИЛИ ТРЕТЬИХ ЛИЦ, ИЛИ ОТКАЗОМ ПРОГРАММЫ РАБОТАТЬ СОВМЕСТНО С ЛЮБЫМИ ДРУГИМИ ПРОГРАММАМИ), ДАЖЕ ЕСЛИ ТАКОЙ ДЕРЖАТЕЛЬ ИЛИ ДРУГОЕ ЛИЦО БЫЛИ ИЗВЕЩЕНЫ О ВОЗМОЖНОСТИ ТАКИХ УБЫТКОВ.

КОНЕЦ ОПРЕДЕЛЕНИЙ И УСЛОВИЙ

Как применять эти условия к вашим новым программам

Если вы разрабатываете новую программу и хотите, чтобы она принесла максимально возможную пользу обществу, лучший способ достичь этого — включить ее в свободное ПО, которое каждый может повторно распространять и изменять согласно данным условиям.

Чтобы сделать это, добавьте в программу следующие уведомления. Надежнее всего будет добавить их в начало каждого исходного файла, чтобы наиболее эффективно передать сообщение об отсутствии гарантий; каждый файл должен содержать по меньшей мере строку, содержащую «знак охраны авторского права» и указание на то, где находится полное уведомление.

```
одна строка, содержащая название программы и краткое описание того, что
она делает .
(С) наименование (имя) автора уууу
```

```
Это свободная программа; вы можете повторно распространять ее и/или
модифицировать ее в соответствии с Универсальной Общественной
Лицензией GNU, опубликованной Фондом Свободного ПО; либо версии 2,
либо (по вашему выбору) любой более поздней версии .
```

```
Эта программа распространяется в надежде, что она будет полезной,
но БЕЗ КАКИХ-ЛИБО ГАРАНТИЙ; даже без подразумеваемых гарантий
КОММЕРЧЕСКОЙ ЦЕННОСТИ или ПРИГОДНОСТИ ДЛЯ КОНКРЕТНОЙ ЦЕЛИ. Для
получения подробных сведений смотрите Универсальную Общественную
Лицензию GNU.
```

```
Вы должны были получить копию Универсальной Общественной Лицензии
GNU вместе с этой программой; если нет, напишите по адресу: Free
Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA
02111-1307 USA
```

Добавьте также сведения о том, как связаться с вами по электронной и обычной почте.

Если программа интерактивная, сделайте так, чтобы при запуске в интерактивном режиме она выдавала краткое уведомление вроде следующего:

```
Гномовизор, версия 69, (С) имя автора год
Гномовизор поставляется АБСОЛЮТНО БЕЗ КАКИХ-ЛИБО ГАРАНТИЙ;
для получения подробностей введите 'show w'. Это свободная
программа, и вы приглашаетесь повторно распространять ее при
определенных условиях; для получения подробностей введите 'show c'.
```

Гипотетические команды 'show w' и 'show c' должны показывать соответствующие части Универсальной Общественной Лицензии. Конечно, используемые вами команды могут называться как-нибудь иначе, нежели 'show w' и 'show c'; они даже могут выбираться с помощью мыши или быть пунктами меню — как больше подходит для вашей программы.

Вы также должны добиться того, чтобы ваш работодатель (если вы работаете программистом) или ваше учебное заведение, если таковое имеется, подписали в случае «отказ от имущественных прав» необходимости на эту программу. Вот образец; замените фамилии:

Компания Братья Ёдины настоящим отказывается от всех имущественных прав на программу 'Гномовизор' (которая делает пасты в сторону компиляторов) написанную Абстрактным К.И.

подпись Мага Ната, 1 апреля 1989 г
Маг Нат, Президент фирмы Вице.

Эта универсальная общественная лицензия не разрешает включать вашу программу в программы защищенные патентами. Если ваша программа — библиотека подпрограмм, вы можете посчитать более полезным разрешить компоновать собственные приложения с библиотекой. Если это вам подходит — используйте GNU Lesser General Public License [<http://www.fsf.org/licenses/lgpl.html>] вместо этой лицензии.

GNU Free Documentation License

Version 1.2, November 2002

Copyright (C) 2000,2001,2002 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document “free” in the sense of freedom; to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondly, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of «copyleft», which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The «Document», below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as «you». You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A «Modified Version» of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A «Secondary Section» is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The «Invariant Sections» are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The «Cover Texts» are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A «Transparent» copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not «Transparent» is called «Opaque».

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary

word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The «Title Page» means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, «Title Page» means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

A section «Entitled XYZ» means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as «Acknowledgements», «Dedications», «Endorsements», or «History».) To «Preserve the Title» of such a section when you modify the Document means that it remains a section «Entitled XYZ» according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties; any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

- A.** Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.
- B.** List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.
- C.** State on the Title page the name of the publisher of the Modified Version, as the publisher.
- D.** Preserve all the copyright notices of the Document.
- E.** Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
- F.** Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.
- G.** Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.

H. Include an unaltered copy of this License.

I. Preserve the section Entitled «History», Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled «History» in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.

J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the «History» section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.

K. For any section Entitled «Acknowledgements» or «Dedications», Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.

L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.

M. Delete any section Entitled «Endorsements». Such a section may not be included in the Modified Version.

N. Do not retitle any existing section to be Entitled «Endorsements» or to conflict in title with any Invariant Section.

O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled «Endorsements», provided it contains nothing but endorsements of your Modified Version by various parties—for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled «History» in the various original documents, forming one section Entitled «History»; likewise combine any sections Entitled «Acknowledgements», and any sections Entitled «Dedications». You must delete all sections Entitled «Endorsements».

COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an “aggregate” if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled «Acknowledgements», «Dedications», or «History», the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See <http://www.gnu.org/copyleft/>.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License «or any later version» applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

```
Copyright (c) YEAR YOUR NAME.
Permission is granted to copy, distribute and/or modify this document
under the terms of the GNU Free Documentation License, Version 1.2
or any later version published by the Free Software Foundation;
with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts.
A copy of the license is included in the section entitled "GNU
Free Documentation License".
```

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the “with...Texts.” line with this:

```
with the Invariant Sections being LIST THEIR TITLES, with the
Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.
```

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.