А. М. Каннер

LINUX: О ДОВЕРЕННОЙ ЗАГРУЗКЕ ЗАГРУЗЧИКА ОС

В последнее время операционные системы (ОС) семейства Linux все чаще встречаются как в портативных электронных устройствах, так и в крупных серверных решениях небезызвестных зарубежных фирм. Наиболее популярные «настольные» дистрибутивы Linux уже сегодня «повернулись» к пользователям лицом — они удобны, надежны и наконец-то просты в освоении, в том числе и неподготовленным пользователем. Так или иначе, Linux постепенно занимает свою нишу в жизни практически каждого человека, в связи с чем все стремительнее растет необходимость в защите данных, хранящихся и обрабатываемых в среде Linux.

Если говорить обобщенно, для защиты современных ПК/серверов (далее будем использовать понятие СВТ) от несанкционированного доступа (НСД) в общем случае принято использовать следующие средства защиты информации (СЗИ):

1. Аппаратный модуль доверенной загрузки (АМДЗ), который позволяет осуществлять «доверенную загрузку ОС» (этот термин не случайно взят полностью в кавычки, дальнейшее повествование раскроет их суть);

2. Подсистема разграничения доступа в ОС, дополнительно реализующая идентификацию/аутентификацию пользователей, политики разграничения доступа, процедуры статического и динамического контроля целостности и т. д. в рамках уже загруженной ОС (загруженного ядра ОС).

Общепринятым в области защиты информации РФ является тот факт, что совместное использование данных СЗИ НСД (каскадно, одно за одним) позволяет в полной мере обеспечить невозможность несанкционированного доступа к защищаемой информации (если не рассматривать возможность физического извлечения носителя информации, которая должна исключаться организационными мерами или с использованием средств криптографической защиты информации).

Рассмотрим подробнее процесс загрузки типового СВТ, на котором установлены аппаратный модуль доверенной загрузки и подсистема разграничения доступа в ОС:

- первоначально управление передается штатному BIOS СВТ, инициализируется оборудование (если установлена парольная защита — BIOS первоначально требует ввести пароль);
- управление перехватывает АМДЗ (сразу после процедуры RomScan), проводятся процедуры идентификации/аутентификации пользователя, контроль целостности/наличия аппаратных (в том числе целостность BIOS) и программных компонентов СВТ (в том числе ядра ОС и компонентов подсистемы разграничения доступа);
- в случае успешного прохождения проверочных процедур АМДЗ управление передается дальше штатному загрузчику ОС (при этом приоритеты загрузки с CD или другого носителя, выставленные в BIOS, игнорируются);

В описанной последовательности обойти или несанкционированно форсировать такой рубеж обороны невозможно. Без извлечения самого носителя информации нелегитимный пользователь не сможет получить доступ к данным.

Далее:

- загрузчик загружает ядро ОС, начинается ранний этап загрузки (тут дополнительно загружается подсистема разграничения доступа), происходят инициализация оборудования, монтирование файловых систем и т. д.;
- подсистема разграничения доступа дополнительно проводит статический контроль целостности данных и предлагает пройти пользователю процедуру идентификации/аутентификации;

- в случае успеха для пользователя загружается его сессия и начинают действовать правила разграничения доступа (ПРД) в соответствии с настройками подсистемы разграничения доступа.

В этой последовательности за счет того, что АМДЗ контролирует целостность всех компонентов подсистемы разграничения доступа (и ядра ОС), сама подсистема разграничения доступа фактически может уже «доверенно» разграничивать доступ пользователей к тем или иным данным в соответствии с выбранной политикой разграничения доступа. Немаловажным обстоятельством, подкрепляющим уверенность в том, что такая последовательность корректна, является тот факт, что подсистема разграничения доступа загружается на самом раннем этапе (фактически сразу после загрузки ядра ОС).

Доверенная загрузка ОС (?)

Примем за истину то, что прошивка/код BIOS и загрузчика являются доверенными (то, как мы этого добьемся, не относится к теме данной статьи). Также примем за истину то, что BIOS/АМДЗ/загрузчик/подсистема разграничения доступа работают корректно, в их коде нет ошибок, то есть представим идеальную ситуацию. Попробуем теперь рассмотреть, обеспечивается ли на самом деле **доверенная загрузка ОС** в общем случае.

После прохождения всех контрольных процедур АМДЗ управление передается коду загрузчика, а не ядру ОС. Именно загрузчик обеспечивает загрузку в память и дальнейшую передачу управления ядру ОС (а затем и загрузку подсистемы разграничения доступа). Таким образом, в общем случае уместнее говорить о **доверенной загрузке загрузчика ОС**.

В случае, например, со стандартным загрузчиком Windows (NTLDR) доверенная загрузка этого загрузчика фактически влечет за собой и доверенную загрузку ОС. Но когда на СВТ используются более «мощные» загрузчики, такие как grub/lilo и т. п., являющиеся стандартными для ОС семейства Linux, за доверенной загрузкой загрузчика ОС может в общем случае не идти доверенной загрузки ОС. Рассмотрим вкратце возможности таких загрузчиков на примере grub.

Загрузчик grub (GRand Unified Bootloader) представляет собой загрузчик, который:

1. соответствует стандарту Multiboot Specification [1], то есть позволяет загружать любую ОС, поддерживающую данный стандарт (Linux, FreeBSD, Solaris и т. д.);

2. позволяет передать управление «по цепочке» другому загрузчику (это происходит в том числе при загрузке Windows/DOS или OS/2);

3. позволяет модифицировать/задавать произвольные параметры при загрузке (параметры ядра, путь к самому ядру ОС или образу начальной загрузки initrd/initramfs Linux) real-time, то есть без модификации кода загрузчика, записанного в загрузочный сектор диска или раздела;

4. позволяет выбрать другой носитель информации (образ) для дальнейшей загрузки;

5. позволяет осуществлять сетевую загрузку (BOOTP, TFTP) и т. д.

Далее в качестве ОС, предустановленной на анализируемом нами абстрактном СВТ, будет рассматриваться ОС семейства Linux. Применительно к таким ОС попробуем выяснить способы использования возможностей загрузчика grub для осуществления недоверенной загрузки ОС при доверенной загрузке загрузчика ОС (то есть фактически обход загрузки подсистемы разграничения доступа в ОС[1]). После этого сформулируем дополнительные рекомендации по устранению таких возможностей.

[1] В качестве примера подсистемы разграничения доступа в ОС Linux использовался ПАК СЗИ НСД «Аккорд-X», разработанный ОКБ САПР. Таким образом, статья призвана в первую очередь указать на те моменты в конфигурировании ОС Linux, которые требуется дополнительно учитывать при использовании этого комплекса. Однако материал статьи полностью применим и к аналогичным продуктам.

Модификация параметров загрузки ОС Linux

Как уже было сказано выше, штатные загрузчики Linux (в том числе и grub) часто позволяют в режиме реального времени модифицировать параметры загрузки без модификации настроек/кода загрузчика, записанного в загрузочный сектор диска (MBR) или раздела (BR).

С точки зрения безопасности в первую очередь интересны такие изменения:
- загрузка в single user mode (в однопользовательском режиме) вместо штатной загрузки ОС;
- передача различных параметров ядру ОС, таких как, например, отключение различных средств защиты и т. д.

В первом случае пользователь получает привилегии суперпользователя ОС (root), этот режим работы используется в административных целях, но, как правило, не запрещен в большинстве дистрибутивов Linux «из коробки».

Практически во всех ОС Linux для загрузки в однопользовательском режиме (SystemV runlevel 1 или S, BSD boot-loader options) достаточно в параметрах загрузки указать в строке с параметрами, передающимися в ядро ОС, single или 1 (рис. 1):

Рис. 1. Пример передачи параметра «single» в строке параметров ядра ОС

Собственно, то, чем опасно оставлять возможность загрузки в single user mode, объяснять подробно не нужно — в однопользовательском режиме можно получить доступ ко всем данным ОС (а подсистему разграничения доступа, например, можно просто отключить).

В качестве примера второго случая (передача прочих параметров ядру ОС) можно привести отключение такой подсистемы безопасности, как SELinux, ставшей де-факто стандартом в ОС Linux, например, способом, приведенным на рис. 2:

Рис. 2. Пример отключения SELinux с помощью передачи соответствующего параметра ядру ОС

Смена текущего загрузочного диска

В загрузчике не столь страшна модификация параметров или выбор какой-то старой версии ядра из уже доступных. Действительно неприятной особенностью тут является возможность смены загрузочного носителя информации (эта возможность исключается АМДЗ до передачи управления загрузчику).

Для реализации смены текущего загрузочного диска достаточно воспользоваться интерактивной командной строкой загрузки, в которой с помощью определенных команд можно изменить текущий загрузочный раздел (root, см. рис. 3—4).

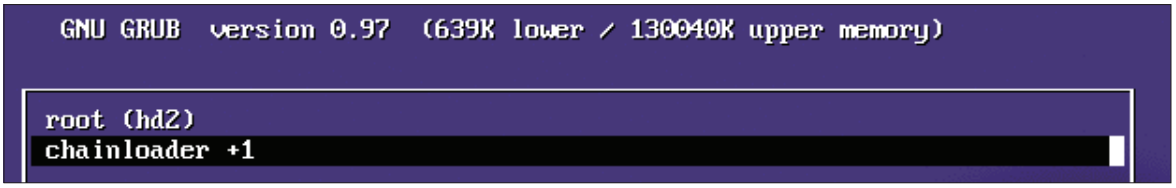

Рис. 3. Демонстрация «проброса» управления загрузчику на внешнем устройстве (в качестве примера загрузка будет осуществляться с загрузочного USB-устройства, данным способом также осуществляется загрузка Windows/DOS или OS/2 с обычных носителей)

Рис. 4. Демонстрация «прямой» загрузки ядра и образа initramfs с внешнего устройства (в качестве примера ядро и образ initramfs будут загружены с того же устройства, что и на рис. 3)

Отметим также, что в ветке grub2 существует возможность загрузки не со стороннего *реального* физического носителя информации, а из загрузочного образа диска с дистрибутивом Linux, расположенного в каком-либо разделе ФС (рис. 5).

Однако такая загрузка live-образа корректна не для всех дистрибутивов Linux. Работоспособность подтверждена только в дистрибутивах, основанных на Ubuntu, в Gentoo Linux, SystemRescueCD, а также в маленьких дистрибутивах, которые после распаковки initrd не используют файлы из образа (то есть полностью помещаются в память).

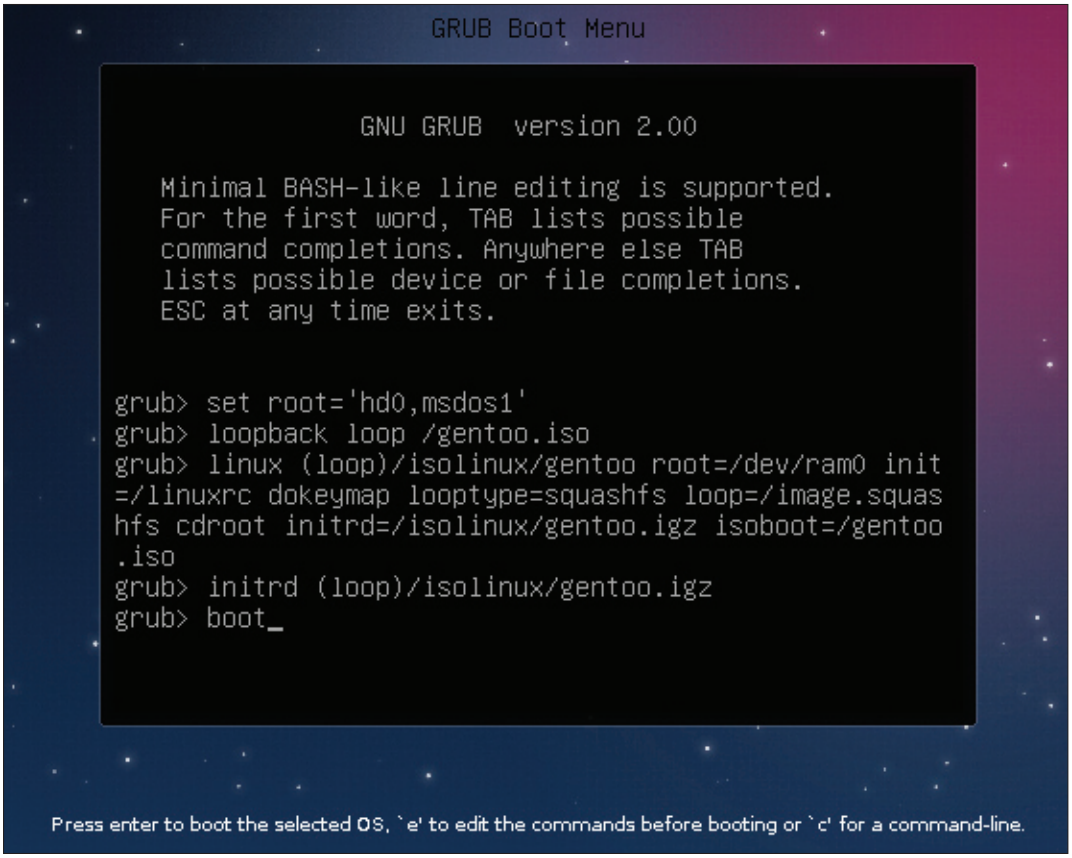

Рис. 5. Демонстрация загрузки ОС из образа загрузочного диска, находящегося в одном из разделов диска

Как настроить загрузчик для обеспечения доверенной загрузки ОС

Какие меры можно предпринять для обеспечения доверенной загрузки ОС после доверенной загрузки загрузчика ОС (в контексте ОС Linux, для которых такие загрузчики, как grub, являются стандартными)? Рассмотрим по порядку следующие возможности загрузчика grub, описанные выше:

1. grub позволяет на лету изменять параметры уже существующих записей (в том числе вход в single user mode);

2. grub позволяет входить в интерактивную командную строку загрузки (с возможностью изменения как загрузочного раздела, так и местоположений ядра ОС и initrd, а также всех параметров ядра).

Поскольку оставлять возможность работы в однопользовательском режиме по тем или иным причинам все равно приходится, лучше позаботиться о защите single user mode отдельно. Следует настроить ОС так, чтобы при загрузке в single user mode, как минимум, запрашивался пароль суперпользователя (так как, вообще говоря, изменить runlevel можно не только во время загрузки ОС, но и позже). Для этого необходимо дополнить /etc/inittab следующей строкой: «~:S:wait:/sbin/sulogin» (см. «Руководство администратора» для конкретного дистрибутива Linux).

В дополнение к этому для защиты от изменения параметров существующих записей (п. 1) можно выставить минимальный тайм-аут ожидания в меню загрузчика (timeout, GRUB_TIMEOUT). Но даже тайм-аут со значением 0 (секунд) позволяет войти в меню загрузчика пользователю, который точно знает, что делает, — обычным нажатием любой клавиши в момент загрузки grub. При этом в grub также имеются, как минимум, следующие параметры для тайм-аутов: GRUB_HIDDEN_TIMEOUT, GRUB_HIDDEN_TIMEOUT_QUIET (или просто hiddenmenu). Но и они не спасают ситуацию.

Выход — заблокировать существующие варианты загрузки с помощью пароля grub (см. рис. 6–7). При этом для разных точек входа можно задавать собственные пароли для блокировки, а в некоторых версиях grub для нестатичных точек входа (например, возникающих при обновлении ядра Linux) — блокировать в том числе и новые записи (lock для статических, lockalternative=true для динамических).

```
(08:10 PM) ~:8|160Kb # grub-md5-crypt
Password:
Retype password:
$1$typOA1$UAbDD4MfTj/IORbo.COdg0
(08:10 PM) ~:8|160Kb #
```

Рис. 6. Создание пароля grub

```
default 0
timeout 1
splashimage=(hd0,0)/boot/grub/splash.xpm.gz

# vim:ft=conf:
password --md5 $1$typOA1$UAbDD4MfTj/IORbo.COdg0

title live-usb startup
root (hd2)
chainloader +1
```

Рис. 7. Демонстрация настройки блокировки для изменения существующих записей grub (/boot/grub/grub.cnf)

Разобравшись с первым пунктом, мы фактически разобрались и со вторым. Войти в интерактивную командную строку загрузки и изменить процесс загрузки также невозможно без ввода заданного пароля.

Дополнительно можно описать еще несколько рекомендаций по настройке такой критически важной вещи, как загрузчик ОС.

Поскольку загрузчики, в том числе и grub, имеют в своем составе собственные реализации разнообразных модулей для поддержки оборудования (USB-стек и т. п.), то правила, действующие

в ОС, работать на этой стадии не будут. Поэтому желательно по мере возможности ограничить доступные модули-расширения для grub (например, модули поддерживаемых файловых систем, CD/DVD-дисков, USB-устройств) только теми, которые реально используются на отдельно взятом СВТ. Желательно удалить модули для поддержки, например, отличных от раздела /boot файловых систем или устройств, с которых потенциально существует возможность загрузить стороннее ядро и/или образ начальной загрузки, например uhci и fat. Кроме того, желательно не оставлять в разделе /boot лишних, сохранившихся с былых времен, ядер и образов начальной загрузки Linux. Причем простого переименования старых версий ядер или складирования их в какой-либо каталог в разделе /boot недостаточно (например, grub2 позволяет выводить список файлов ФС выбранного корневого раздела прямо в интерактивную консоль).

При настройке загрузчика основной целью является ограничение любого рядового пользователя в выборе. Процесс загрузки ОС должен быть строго одинаков, не должно быть способа, при котором изменить процесс загрузки может кто-либо кроме администратора, ответственного за работу СВТ и всех средств защиты информации.

Выводы

По умолчанию в ОС Linux используются довольно необычные загрузчики, которые, с одной стороны, позволяют гибко конфигурировать загрузку системы, передавать ядру ОС нужные только в данный момент параметры, загружаться по сети и т. д. С другой же стороны, такие загрузчики неявно дают возможность сделать все эти изменения на лету, не изменяя код самого загрузчика, не оставляя никаких следов. При этом по умолчанию для всего этого не требуется каких-то эксклюзивных прав.

Загрузчики сами по себе защитить нельзя, все механизмы защиты могут осуществляться только штатно самим же загрузчиком (о чем и рассказано в данной статье). Мы не можем шифровать код загрузчика — все конфигурационные файлы должны располагаться в открытом разделе (причем только в поддерживаемой загрузчиком ФС).

Однако мы должны обеспечивать контроль целостности кода и конфигурационных данных загрузчика средствами, например, АМДЗ. Дальнейший доступ к конфигурации загрузчиков можно контролировать с помощью подсистемы разграничения доступа. Однако для этого необходимо, чтобы загрузчик корректным образом обеспечил загрузку ядра и уже в дальнейшем подсистемы разграничения доступа. Таким образом, загрузчик применительно к ОС Linux является еще одним связующим звеном между АМДЗ и подсистемой разграничения доступа в ОС Linux (то есть промежуточным элементом связки «АМДЗ — подсистема разграничения доступа»).

Нужно понимать, что производитель СЗИ НСД может только описать набор требований к среде функционирования своего СЗИ, необходимых для того, чтобы СЗИ действительно эффективно защищало данные. Реализовать же некоторые требования должен сам пользователь (администратор безопасности), для этого он должен разбираться в том, как функционирует его СВТ, — уметь локализовать и закрыть узкое место в системе защиты. Данная статья является попыткой указать пользователям СЗИ НСД нужное направление — направление всестороннего обеспечения защиты информации, с самого низкого уровня.

СПИСОК ЛИТЕРАТУРЫ:

1. *Bryan Ford, Erich Stefan Boleyn, Free Software Foundation Inc*. Multiboot Specification version 0.6.96. URL: http://www.gnu.org/software/grub/manual/multiboot/multiboot.html (дата обращения 01.07.2013).