Д. А. Частухин, В. В. Широков, М. А. Щиголева

Методика изучения механизмов пространства имен «контрольные группы» Linux в дисциплине «Операционные системы» направления «Информационные системы и технологии» с учётом форматов гибридного обучения

Санкт-Петербургский государственный электротехнический университет «ЛЭТИ» им. В.И. Ульянова (Ленина) г. Санкт-Петербург, Россия

Аннотация. *В работе предлагается методика изучения свойств одного из видов пространств имен «контрольные группы» в дисциплине «Операционные системы» направлений «Информационные системы и технологии».Основное внимание уделяется свойству «видимости» каталогов, обеспечивающему изоляцию ресурсов процессов-потомков от ресурсов процессов-родителей, создающему безопасную среду выполнения процессов. Подготовка учебного материала подразумевает возможность гибридного обучения.*

Ключевые слова: механизмы контейнеризации; изоляция процессов; пространства имен; контрольные группы; видимость каталогов; ограничение ресурсов процессов; гибридное обучение

2020 год в обучении студентов установил необходимость проработки различных форматов обучения по всем действующим программам обучения. За прошедший период были апробированы различные формы проведения учебных занятий – очные, удаленные, дистанционные, онлайн, офлайн, – в различной терминологии; их смешанные формы, с эксклюзивными наименованиями в различных вузах, но подразумевающие сочетание очных форматов с применением электронных форм обучения и дистанционных образовательных технологий. За три года проведения образовательного процесса в этих форматах возникла необходимость пересмотреть методологию организации учебного процесса с конечной целью освоения необходимого учебного материала. Помимо вопросов технической организации качественной связи очной и удалённой группы обучаемых, синхронных и асинхронных форм занятий, видеоконференций и учебных видеозаписей, встают вопросы такой подготовки учебного тематического материала, которая позволит студенту получить необходимые знания, умения, навыки, опыт погружения в предметную среду своей будущей профессиональной деятельности. Необходимо поддерживать вовлеченность и активность обучаемых на протяжении всего учебного курса чередованием теоретического материала, практического, тестирования и оценивания, организации обратной связи по результатам оценки.

Практика обучения в гибридном формате показала необходимость более четкого наполнения занятий по изучаемым темам, необходимость акцента на преемственность материала от темы к теме, чёткое согласование теоретического лекционного материала и практического на лабораторных и практических занятиях по теме изучения, выполнение заданий по всем темам курса обучения, заданий с использованием материала всех тем учебной дисциплины.

Структурирование по темам курса обучения создаёт методологию формирования версий преподавания дисциплины подготовки с возможностью варьирования как напряженности и углубленно-

сти предметной составляющей, так и реализацию в различных форматах компонентов гибридного обучения. Методика организации учебного процесса содержит программу из набора тем обучения с сочетанием теоретической и практической подготовки, схем тестирования, оценочных материалов по освоению полного курса обучения. Методология подхода к преподаванию с учётом форматов гибридного обучения была апробирована для дисциплины «Операционные системы» направления «Информационные системы и технологии». Тематическим разделом дисциплины является «Изучение механизмов пространства имен «контрольные группы» Linux», который может быть самостоятельной темой программы обучения, разделом дисциплины из нескольких тем и обладает выраженным сочетанием теоретического и практического материала, удобными формами тестирования для проверки полученных знаний и навыков.

Пространства имен операционной системы Linux предоставляют механизмы изоляции ресурсов дочерних процессов от ресурсов родительских процессов. Эти механизмы лежат в основе построения одного изнеобходимых функциональных атрибутов проектирования информационных систем, а именно, средств обеспечения безопасности информационных систем, в частности, контейнеров [1].

Поэтому изучение учащимися в дисциплине «Операционные системы» направления подготовки «Информационные системы и технологии» механизмов пространств имен и контейнеризациис одним из внутренних механизмов функционирования - механизмов пространств имен, будет необходимо и актуально как теоретического материала, средства практического применения в профессиональной деятельности и проектирования инструментальных средств решения задач современных информационных технологий. В настоящее время ОС Linux предлагает следующие типы пространств имен, позволяющих осуществлять изоляцию процессов по соответствующим видам ресурсов [2]:

1. IPC – межпроцессное взаимодействие (очереди сообщений);
2. Network – сетевые устройства, сетевые стеки, порты;
3. PID – идентификаторы процессов;
4. UTS – имена хоста и домена;
5. Time – системные часы;
6. User – идентификаторы пользователей и групп;

Mount – точки монтирования файловых систем;

Cgroup – корневые директории контрольных групп.

Перечисленные типы пространств имен можно разделить на три группы с точки зрения сложности освоения учащимися и реализации ими лабораторных работ:

1. Пространства имен IPC, Network, PID, UTS, для которых в функцию создания дочернего процесса достаточно ввести соответствующий флаг (CLONE_NEWIPC, CLONE_NEWNET, CLONE_NEWPID, CLONE_NEWUTC), чтобы получить возможность продемонстрировать работу процессов (родителя и потомка) в изолированных пространствах по соответствующим видам ресурсов.

2. Пространства имен Time, User, для которых недостаточно ввести флаг (CLONE_NEWTIME, CLONE_NEWUSER), но и необходимо произвести предварительную настройку определенных системных файлов. В первом случае это файл /proc/[pid]/timens_offsets, через который необходимо задать смещение системного времени процесса-потомка от системного времени процесса-родителя. Во втором случае это файлы /proc/[pid]/uid_map и /proc/[pid]/gid_map, через которые необходимо задать допустимые диапазоны идентификаторов пользователя и группы, которые могут быть установлены в процессе-потомке.

3. Пространства имен Mount, Cgroup, для которых установка соответствующего флага (CLONE_NEWNS, CLONE_NEWCGROUP) должна продемонстрировать отсутствие «видимости» процессами определенных ветвей файловой системы. В первом случае процесс-родитель «не видит» изменений файловой системы процесса-потомка, которые последний произвел путем вызова функции (команды) mount. Во втором случае процесс-потомок «не видит» ветку дерева файловой системы в

каталоге /sys/fs/cgroup, описывающую контрольную группу процесса-родителя. Такая изоляция не позволяет процессу-потомку воздействовать на ресурсы процесса-родителя.

В данной работе рассматривается методика изучения и проверки возможностей управления ресурсами процессов на основе контрольных групп и изоляции ресурсов с помощью механизма пространства имен Cgroup.Методика может быть использована для проведения практических занятий и выполнения лабораторных работ учащимися при изучении дисциплины «Операционные системы».

На первом этапе изучения механизма контрольных групп учащимся предлагается создать контрольную группу по определенному виду ресурсов. Наиболее наглядным для этой цели является ресурс памяти. Для этой цели необходимо создать каталог в файловой системе /sys/fs/cgroup/memory, например, /sys/fs/cgroup/memory/test_cgroup.

При создании каталога /sys/fs/cgroup/memory/test_cgroup в нем автоматически создается группа файлов, содержащих информацию о контрольной группе. Например, в файле cgroup.procs будут содержаться идентификаторы процессов, принадлежащих этой группе, а в файле memory.limit_in_bytes будет содержаться ограничение на объем памяти, которым могут пользоваться процессы, принадлежащие этой группе. При создании файла memory.limit_in_bytes в нем записаны данные, говорящие об отсутствии ограничений на размер памяти.

Включить процесс в контрольную группу можно, записав его идентификатор (pid) в файл cgroup.procs, а ограничить размер памяти процессам группы можно, записав этот размер в файл memory.limit_in_bytes.

Учащимся может быть предложено задание по управлению контрольными группами, включающее перечисленные выше действия и наблюдение за результатами выполнения этих действий.

Следующим этапом работы является знакомство с механизмом пространства имен контрольной группы. Для этого необходимо предварительно ознакомиться с программными интерфейсами создания дочерних процессов с возможностью создания новой контрольной группы [3]. К таким интерфейсам относится функция clone(), в набор параметров которой могут входить перечисленные выше флаги типов пространств имен, в том числе и CLONE_NEWCGROUP. Другим видом интерфейса является совместное использование функций fork() и unshare(). То есть создаваемый функцией fork() дочерний процесс должен вызвать функцию unshare() с передачей флага типа пространства имен, и тем самым продолжить свое выполнение в новом, изолированном от процесса-родителя пространстве. Как сказано в документации [4], «когда процесс создает новое пространство имен контрольной группы с помощью clone() или unshare() с флагом CLONE_NEWCGROUP, его текущие cgroups каталоги становятся корневым cgroup каталогом нового пространства имен». Вот этот факт и предстоит увидеть учащимся на данном этапе работы.

Можно предложить следующие варианты проверки видимости каталогов cgroups в процессе-потомке:

1. Использование в процессе-потомке вызова функции system("/bin/bash"). В этом случае процесс-потомок вызывает внутри себя командный интерпретатор.

2. Использование в процессе-потомке вызова функции system("cat /proc/self/cgroup").

3. Вызов функции system() не является предпочтительным в программах, которые требуют привилегий, а вызовы clone() и unshared() с флагом CLONE_NEWCGROUP требуют привилегий администратора (CAP_SYS_ADMIN).

Ознакомление с представленным материалом позволит обучающимся лучше понять внутренние механизмы функционирования контейнеров, одним из элементов построения которых являются пространства имен, а Методика организации учебного процессаподдержать методологию формирования версий преподавания дисциплины подготовки с возможностью варьировать углубленность материала предметной составляющей, для различных форматов гибридного обучения.

Список литературы:

1. С Райс Лиз. Безопасность контейнеров. Фундаментальный подход к защите контейнеризированных приложений. – СПб.: Питер, 2021. 224 с.: (Серия «Бестселлеры O’Reilly»).

2. Home DTOS Other Projects Knowledge Base Linux Manpages Community Contribute // Главная DTO Другие проекты База знаний Linux Manpages: информационный портал. Указатели на информацию о различных типах пространств имен: [электронный ресурс] – режим доступа. – URL: https://man7.org/linux/man-pages/man7/namespaces.7.html (дата обращения 16.03.2023).

3. В.В. Широков, М.А. Щиголева. Методические рекомендации по практическому освоению программных интерфейсов пространств имен операционных систем // Современное программирование III Всероссийская научно-практическая конференция. Нижневартовск, 26-27 ноября 2020. С. 295–299.

4. Cgroup_namespaces (7) // Обзор пространств имен cgroup в Linux: Описание. Обзор пространств имен. электронный ресурс] – режим доступа. – URL: https://man7.org/linux/man-pages/man7/cgroup_namespaces.7.html (дата обращения 16.03.2023).

D. A. Chastuhin, V. V. Shirokov, M. A. Schigoleva

Methodology for studying the mechanisms of the namespace "control groups" of Linux OS in the discipline "Operating systems" of the directions "Information systems and technologies" taking into account hybrid learning formats

Saint Petersburg Electrotechnical University, Russia

Abstract. *The paper proposes a methodology for studying the properties of one of the types of namespaces "control groups" in the discipline "Operating systems" of the directions "Information systems and technologies" and "Information security". The main attention is paid to the "visibility" property of directories, which provides isolation of the resources of child processes from the resources of parent processes, creating a safe environment for the execution of processes.The preparation of educational material implies the possibility of hybrid learning.*

Keywords: containerization mechanisms; process isolation; namespaces; control groups; directory visibility; limitation of process resources; hybrid learning