ПОДХОД К ПРОВЕДЕНИЮ АНАЛИЗА СЕТЕВОГО ТРАФИКА НА ТЕРМИНАЛЬНЫХ СЕРВЕРНЫХ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМАХ

APPROACH TO NETWORK TRAFFIC ANALYSIS ON TERMINAL SERVER COMPUTING SYSTEMS

L. Vasin

Summary. The article discusses one of the options for implementing a network traffic control system in terminal systems. Such a system provides access to network packets existing within the terminal system when thin clients are connected to it. This allows analyzing and calculating the network traffic of users working with the terminal system. In addition, this system allows you to organize a system of safe operation on the Internet for such a category of users. The complexity of analyzing user network packets in a traditional terminal system is due to the inability to allocate each user session with an individual network IP address. The purpose of the work is to study the possibility of building a terminal system for collective work in which monitoring and analysis of user network traffic is carried out. A method implemented by such a system based on the use of container isolation technology for user sessions has been proposed. Using a system with such an organization allows each terminal session to be provided with an individual variant that is necessary for the operating system to use. At the same time, a unique network IP address will be allocated, which will make it possible to isolate from the general network traffic a packet with belonging to each terminal session. The article describes in detail the general structural scheme of the organization of the system, shows the order of its operation, examples of network configuration, as well as the rules for processing network traffic. There are various options for optimizing network access from terminal sessions, as well as analyzing network traffic. The proposed organization allows you to provide full analysis and management of network packets. In conclusion, the article presents the main conclusions on the work done.

Keywords: terminal system, data network, operating system network traffic analysis, container, isolation, physical server, work sessions, packet filter, network address.

Васин Леонид Анатольевич
К.т.н., Пензенский государственный университет архитектуры и строительства
leo.vasin@gmail.com

Аннотация. В статье рассматривается один из вариантов реализации системы контроля сетевого трафика в терминальных системах. Такая система обеспечивает доступ к сетевым пакетам, существующих внутри терминальной системы, при работе тонких клиентов, подключенных к ней. Это позволяет осуществить анализ и подсчет сетевого трафика пользователей, работающих с терминальной системой. Кроме этого, такая система позволяет организовать безопасную работы в сети Интернет для различной категории пользователей. Сложность выполнения анализа пользовательских сетевых пакетов в традиционной терминальной системе обусловлена невозможностью выделения каждому пользовательскому сеансу работы индивидуального сетевого IP адреса. Цель работы заключается в исследовании возможности построения терминальной системы для коллективной работы, в которой проводится мониторинг и анализ пользовательского сетевого трафика. Предложен способ реализации системы на основе применения технологии полной виртуализации и контейнерной изоляции сеансов работы пользователей. Использование системы с подобной архитектурой позволяет обеспечить каждый терминальный сеанс индивидуальным вариантом необходимой для работы операционной системы. При этом, каждому из них, будет назначен уникальный сетевой IP адрес, что позволит выделить из общего сетевого трафика пакет с принадлежностью к каждому терминальному сеансу. В статье подробно рассмотрена общая структурная схема организации системы, приведен порядок её работы, примеры сетевой настройки, а также правила обработки сетевого трафика. Предложенная организация позволяет обеспечить полноценный анализ и управление сетевыми пакетами. В заключении статьи представлены основные выводы о проделанной работе.

Ключевые слова: терминальная система, сеть передачи данных, операционная система анализ сетевого трафика, контейнер, изоляция, физический сервер, сеансы работы, пакетный фильтр, сетевой адрес.

Введение

При работе пользователей в информационных системах различного класса необходимо наличие систем анализа и управления сетевым трафиком. Это необходимо при организации работы в пунктах коллективного доступа, а также предприятиях различных форм собственности. Такая система обеспечивает как информационную безопасность, так и защиту от нежелательного контента пользователей, например, в компьютерных классах учебных заведений. Обеспечение беспроводным доступом, так же требует соблюдения законодательных норм по организации доступа к сети Интернет, в процессе которого весь сетевой трафик анализируется и принимается решение на его передачи пользователям или блокировку [1].

Под анализом сетевого трафика понимается совокупность программно-аппаратных компонентов, участвующих в процессе обработки, классификации, контроля

в реальном режиме времени [2]. Он может проводится над всеми категориями сетевых пакетов, существующих в системе, например, локальными, которые циркулируют внутри информационной системы, а также межсетевыми — взаимодействующим с другими, внешними компьютерными сетями. Процесс анализа заключается в мониторинге и изучении содержимого сетевого пакета, с служебными заголовками. На их данных доступна информация по используемым сетевым адресам, а также типам применяемых сетевых протоколов.

Для идентификации сетевых пакетов по сеансам работы используют сетевой адрес IP, который является идентификационным для используемого оборудования, например, персональные вычислительные машины различного назначения и другие мобильные решения. Каждый сетевой интерфейс имеет свой IP адрес, который назначается системным администратором в ручном или автоматическом режиме [3]. При работе пользователя с различными сетевыми приложениями, использую протоколы прикладного уровня модели OSI, происходит генерация трафика с формированием сетевых пакетов. Они содержат пользовательские данные, исходящие и входящие IP адреса, номера портов приложений и другая служебная информация.

Существующие информационные технологии позволяют провести процесс сбора данных и его анализа по необходимым критериям, например, используемые протоколы транспортного и прикладного уровней, значения IP адресов и другая информация. Для этого необходимо организовать захват сетевого трафика в прозрачном режиме с сетевого интерфейс маршрутизатора, через который осуществляется сетевой доступ с пользовательского оборудования в другие сети.

В работе рассматривается подход к организации управления сетевым трафиком в терминальной информационной системе. Она организована по принципу выполнения всех приложений на центральной вычислительной машине, к которой подключены пользовательские терминалы получившие названия «тонкие клиенты» или «терминальная станция» [4]. Это класс аппаратных устройств или пользовательского программного обеспечения, реализующей клиент-серверную или терминальную модель организации вычислительного процесса, при которой основная часть вычислений выполняется на сервере. Основное достоинство реализации такой информационной системы в соотношении вычислительной эффективности и низкой стоимости клиентского оборудования, которое имеет минимальные аппаратные и программные характеристики. Минимальное использование собственных аппаратных ресурсов — есть основное преимуществом тонкого клиента.

Существуют несколько подходов, принятых к организации терминального сервера: используемых в операционных системах семейства UNIX и в операционных системах Windows. В первом используется технология X Winow System, во втором — использование протокола RDP (Remoute Desktop Protocol). Оба обеспечивают выполнение базовых функций графической среды визуализации и организуют взаимодействие с устройствами ввода-вывода.

X-Windows System относится к клиент-серверной технологии, где сервером выступает программный сервис, например, Xorg, функционирующий на терминале пользователя, а клиентом — графическое приложение, функционирующее на сервере. X-сервер обеспечивает отображение графической информации удаленным программам пользователей, называемыми X-клиентами [5].

Протокол RDP предназначен для использования ресурсов высокопроизводительного сервера терминалов другими менее производительными рабочими станциями. Он является протоколом прикладного уровня, базирующегося на TCP и обеспечивает организацию графического вывода на пользовательский терминал, а также обеспечивает передачу данных с устройств ввода — вывода на терминальный сервер. В рамках графического вывода передается точная копия экрана и команды на отрисовку графических объектов. При вводе передаются скан-коды клавиатуры и координаты манипулятора ввода [6].

Таким образом, оба варианта позволяют организовать удаленный доступ к центральной вычислительной машине (терминальному серверу) и обеспечить запуск и выполнение графических приложений.

При использовании информационных систем с терминальной организацией, невозможно идентифицировать сетевой трафик пользовательских устройств, подключенных к терминальному серверу. Причина этого — невозможность идентифицировать сетевой трафик пользовательских терминалов или тонких клиентов по причине использования в сеансе работы IP адреса терминального сервера, на котором выполняются сетевые приложения.

Для осуществления сетевой идентификации работы пользовательских приложений, а так же для анализа такого сетевого трафика необходимо обеспечить выполнения условия выделения уникального IP адреса для пользовательского сеанса работы. Для этого на терминальном сервере необходимо использовать технологии виртуализации в которой выполняется пользовательский сеанс и назначается индивидуальный уникальный IP адрес, который будет идентифицировать сетевые пакеты.

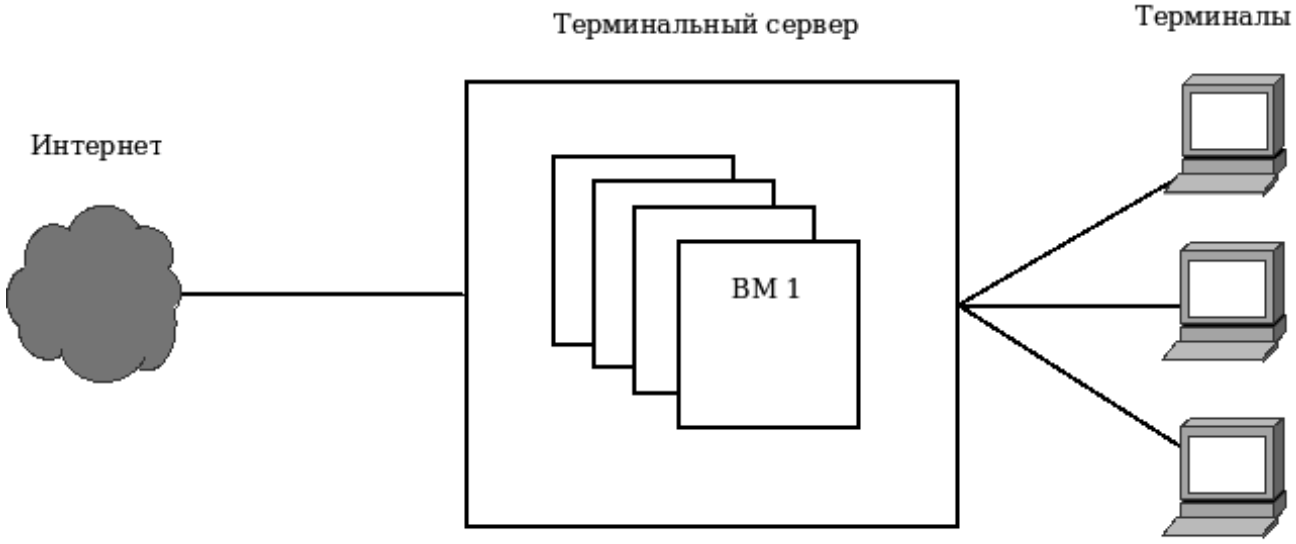

Рис. 1. Структурная организация терминальной системы

Для повышения эффективности использования аппаратных ресурсов сервера на базе ОС Linux рекомендуется использовать технологию LXC (Linux-container). Это обусловлено возможностью применения собственных файловых систем контейнеров и один общий экземпляр базового ядра ОС. Технология контейнеров, например, Docker позволяет создать набор необходимого программного обеспечения с изолированной процессорной средой и собственной файловой системой, в которой находится прикладное и системного программное обеспечение. Такая организация позволяет повысить общий КПД использования терминальной системы с пользовательскими сеансами, выполненными в виде изолированного контейнера, при этом достигается незначительная потеря производительности [7].

Если необходимо использовать отличные от базовой версии ОС, например, Windows, в таком случае необходимо использовать систему полной виртуализации, например, KVM, VmWare, VirtualBox [8].

Целью исследования является разработка структурной организации процесса анализа сетевого трафика в составе информационной системы, использующей терминальный сервер. Такая система, ориентированная на пользователей, работающих с персональными информационными средствами в терминальном режиме, относящихся к классу «Тонкие клиенты». Она должна обеспечивать анализ сетевого трафика, принадлежащего пользователю, работающего с терминальной системой. В ходе этого процесса необходимо обеспечить проведение оценки объемов использованного трафика, его типе, а также мониторинг используемых внешних сетевых ресурсов.

Основная часть

Для обеспечения построения терминальной системы с возможностью полного анализа сетевого трафика показана её разработка с использованием изолированных контейнеров. Для анализа сетевых пакетов на терминальном сервере развертывается система контейнеров (виртуальных машин), в которых функционирует необходимая операционная система и набор сетевых приложений, с которым работает пользователь.

Структурная организация терминальной системы показана на рисунке 1. Она состоит физического сервера под управлением ОС, на котором развернута система визуальных машин. Может быть использованы системы визуализации следующих типов: LXC, KVM или OpenVZ. Терминальный сервер обеспечивает информационное взаимодействие с пользовательскими терминалами, которые реализуются аппаратно или программным путем. В качестве аппаратных могут быть применяться промышленные реализации, например, HP, Compaq, Depo, Aquarius. Программный вариант может быть реализован с помощью терминального программного обеспечения, например, X-client по протоколу X11, RDP клиент, TightVNC клиент для подключения к VNC службе, клиенты SSH и Telnet, который входит в состав используемых ОС. Терминальный сервер содержит два сетевых интерфейса и обеспечивает доступ пользователей во внешние компьютерные сети. Терминалы (тонкие клиенты) подключены к терминальному серверу по локальной сети и используют сетевой протокол Ethernet Base-T. Загрузка клиентского программного обеспечения на аппаратные терминалы может происходит, используя протокол PXE.

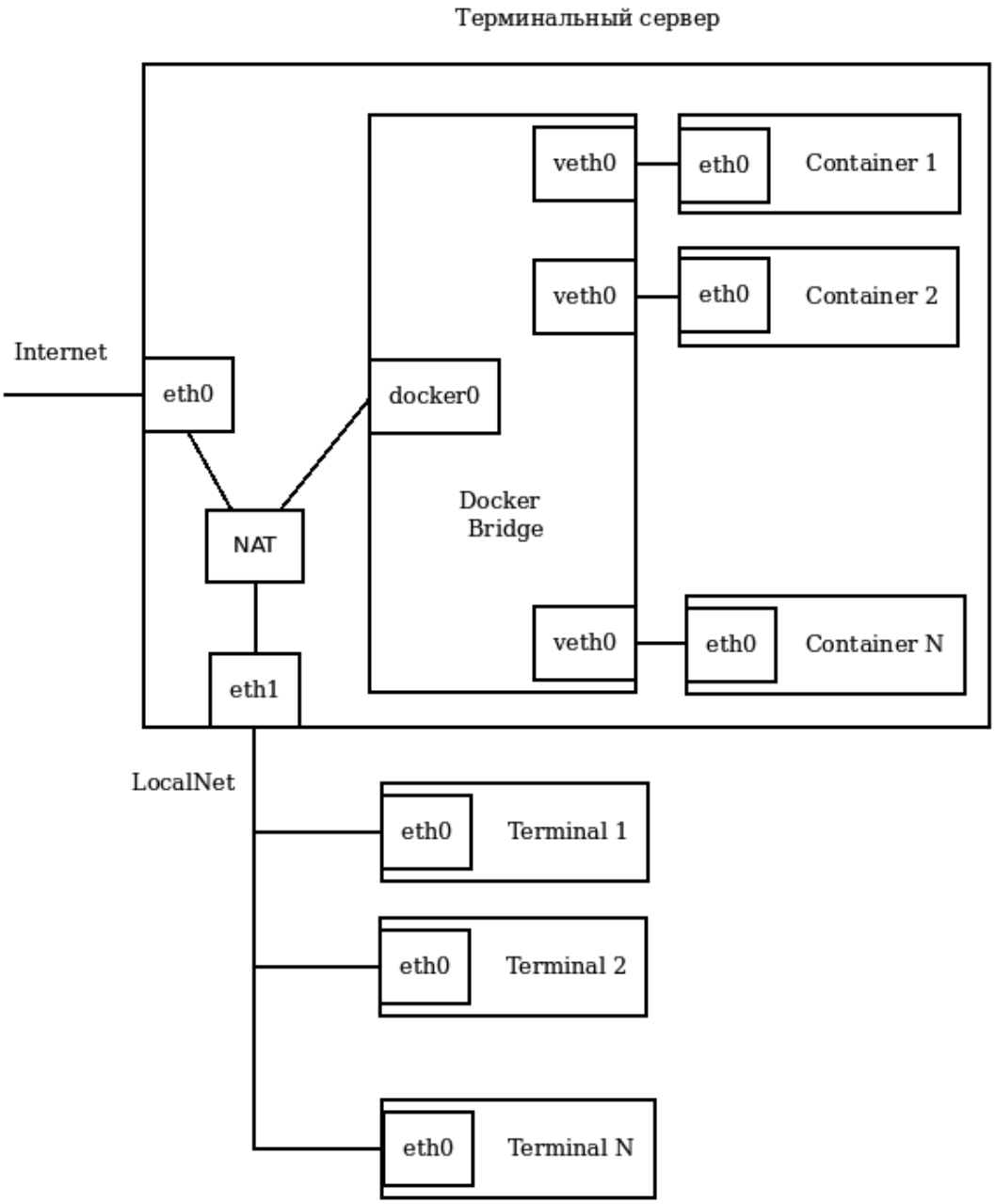

Рис. 2. Структурная организация сетевой подсистемы изолированных контейнеров.

В программную часть входит базовая ОС, система виртуализации и образы, из которых формируются изолированные контейнеры или виртуальные машины под управлением гипервизора. Выбор необходимой технологии обусловлен системными требованиями к программному обеспечению используемое в работе пользователя. Существует возможность запуска графических приложений в изолированном контейнере [9]. Если необходима ОС Windows или другая, не совместимая с ОС Linux, в этом случае применяется система KVM, которая поддерживает аппаратный тип виртуализации на базе Intel VT или AMD SVM. В случаи использования пользователями ОС однотипной с базовой ОС, то может быть использована технология визуализация на уровне операционной системы, например, программное обеспечение Docker, которая позволяет управлять системой изолированных контейнеров, внутри которых функционируют пользовательские терминальные сеансы.

В качестве базовой ОС на терминальном сервере используется Linux CentOS7, в состав которой входит необходимое программное обеспечения для управления

виртуальными машинами и контейнерами, а также для организации сетевой загрузки тонких клиентов и дальнейшей работы пользователей с необходимыми приложениями.

Сетевая подсистема обеспечивает взаимодействие сетевыми пакетами между сетевыми приложениями контейнеров или виртуальных машин и внешней и локальной сетями. Для этого, в процессе её администрирования назначаются уникальные сетевые IP адреса каждому контейнеру или виртуальной машине, а также настраивают передачу сетевого потока внутрь терминальных сеансов работы. Структурная организация сетевой подсистемы, для варианта использования системы изолированных контейнеров Docker, показана на рисунке 2.

Она состоит из моста (bridge), реализованного в docker, сетевых интерфейсов физического сервера и сетевых интерфейсов в контейнерах. Мост с именем docker необходим для организации соединения на канальном уровне нескольких сегментов Ethernet без необходимости применения протокола IP, на уровне MAC адресов, где передача выполняется на уровне 2 модели OSI. Такая организация сети позволяет создать сетевой сегмент по протоколу Ethernet, в который входят все сетевые адаптеры изолированных контейнеров. Эта схема напоминает «виртуальный» коммутатор, который управляет процессом коммутации пакетов Ethernet [10]. Использование моста позволяет организовать прозрачное прохождение пакетов более высокого уровня для контейнеров с минимальными временными задержками и без ограничения типа пропускаемого сетевого трафика.

В состав сетевого моста docker входят виртуальные адаптеры: docker0, veth0, veth1, …, vethN. Интерфейс docker0 используется для взаимодействия с внешними сетями через сетевой пакетный фильтр используя NAT и физический интерфейс eth0, а также обеспечивает подключение пользовательских терминалов через сетевой интерфейс eth1 по локальной сети. Каждому сетевому интерфейсу назначается свой сетевой адрес IP в пределах одной сети, например, docker0–172.16.1.1. Каждый контейнер имеет сетевой интерфейс eth0 с адресом 172.16.1.2, контейнер 2 — eth0 с адресом 172.16.1.3, контейнер N — eth0 с адресом из диапазона 172.16.1.0/24. Сетевые адреса контейнеров и интерфейса docker0 входят в одну сеть.

Сетевой пакетный фильтр iptables управляет сетевым трафиком по заданным правилам, которые обеспечивает преобразование сетевых адресов для передачи трафика к виртуальным сетевым интерфейсам контейнеров eth0 через интерфейс Docker Bridge, с присоединенными виртуальными интерфейсами veth. При этом, Docker использует маскарадинг (NAT MASQUERADE) для исходящего сетевого трафика, согласно правилам маршрутизации default gateway для интерфейсов контейнеров eth0.

При такой организации работы сети на терминальном сервере производится настройка NAT таким образом, чтобы обеспечить каждый терминальный сеанс персональным контейнером. Это возможно при разрешении прохождения только терминального трафика с номерами портов (VNC:5900, RDP:3389, SSH:22), по конкретному IP адресу сетевого интерфейса eth0, соответствующего контейнера. В этом случае дополнительно блокируется доступ с терминалов во внешние компьютерные сети.

При применении системы полной виртуализации, например, KVM, организация сетевой среды выполняется на основе сетевого моста (Linux bridge) и сетевого пакетного фильтра. В сетевой мост br0, в который входят виртуальные сетевые интерфейсы vnet0, vnet1, vnet2,, vnetN, а также один из физических интерфейсов терминального сервера eth1. Интерфейс vnet0 обеспечивает прохождение сетевого трафика виртуальных машин во внешние сети, используя сетевой пакетный фильтр с NAT MASQUERADE и правила маршрутизации default gateway, применительно к сетевым интерфейсам eth0 виртуальных машин и br0 сетевого моста. Через интерфейс eth1 обеспечивается подключение терминальных устройств через локальную сеть. Включение этого интерфейса в состав сетевого моста br0 позволяет организовать передачу сетевых пакетов от терминальных устройств до виртуальных машин в пределах одного сегмента Ethernet. Структурная организация сетевой подсистемы, для варианта использования системы виртуальных машин KVM, показана на рисунке 3.

Для взаимодействия с внешними сетями и для управления коммуникациями между виртуальными интерфейсами на терминальном сервере используется сетевой пакетный фильтр Linux iptables. В процессе взаимодействия используются его таблицы nat и filter. Они осдержат правила, позволяющие разрешить прохождение трафика к IP адресам и портам контейнеров или виртуальных машин, а также дает возможность взаимодействовать с IP адресами внешних хостов.

Показанная организация сетевой системы терминального сервера позволяет надежно выделить сетевые пакеты терминальных сеансов работы по их собственным IP адресам и тем самым проводить их анализ. Этот процесс заключается в захвате трафика с сетевого моста виртуальных интерфейсов veth, изолированных контейнеров или vnet виртуальных машин для его накопления с дальнейшей обработкой программными анализатора-

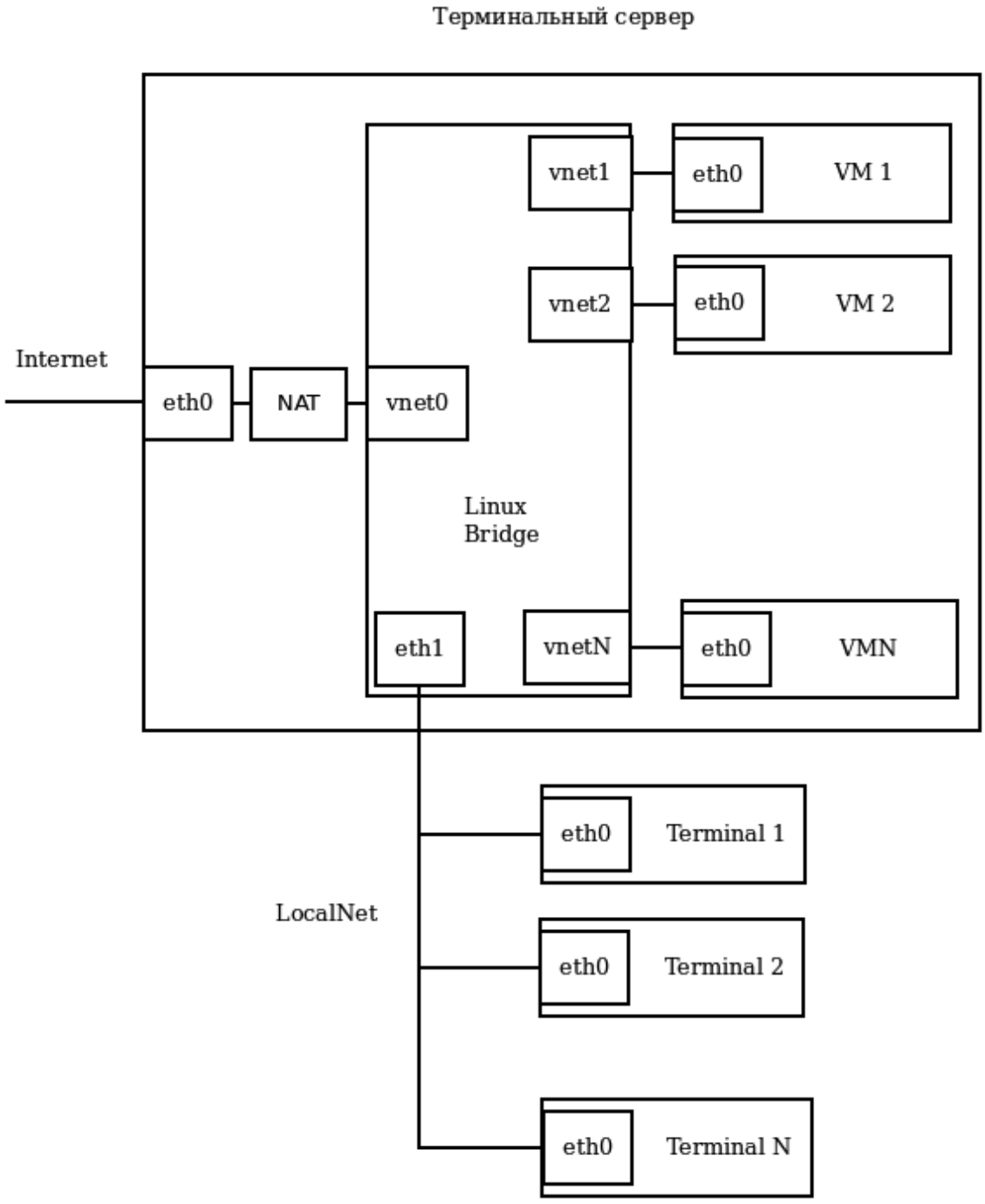

Рис. 3. Структурная организация сетевой подсистемы при использовании полной виртуализации

ми. Целью этого является исследования содержимого сетевых пакетов и расчет объема использованного исходящего трафика.

Для захвата трафика может быть применен механизм журналирования пакетного фильтра iptables, реализованный с помощью действий LOG или ULOG. При их использовании iptables передает в системный журнал или внешнюю базу данных сетевую активность по заданным критериям. В их качестве могут выступать: IP адрес, тип протокола, название сетевых интерфейсов.

В рамках выполнения исследования была развернута терминальная система, состоящая из сервера и несколькими терминалами. Её назначение — реализация в качестве пункта коллективного доступа пользователей в сеть Интернет. В аппаратных платформах использовались x86 совместимые терминальный сервер

и тонкие клиенты. В качестве операционных систем использовалась версия Linux CentOS. На терминальном сервере развернута система виртуальных машин, использующих систему виртуализации Linux-Vserver, используя изоляцию контейнеров на уровень ядра, к которым подключаются тонкие клиенты. В среде контейнеров функционируют пользовательские терминальные сеансы, использующие ОС Linux. Тонкие клиенты загружают собственные копии минимальной версии ОС Linux, используют сетевой вариант загрузки (PXE). В неё входит различные программные клиенты для удаленного доступа:

- классический X-server для доступа к X-terminal, обеспечивающий запуск графических приложений
- клиент VNC
- клиент SSH

Именно такой набор программного обеспечения позволяет обеспечить работу с графическими и тестовыми приложениями в среде терминального сервера.

В системе для анализа сетевого трафика используется программные анализаторы wireshark и tcpdump. Учет трафика производится на основе данных пакетного фильтра iptables, записанных в базу данных под управление СУБД MySQL. В рамках выполнения работы разработано программное обеспечение, позволяющее вычисление использованного сетевого трафика по IP адресам сетевых контейнеров.

Выводы

В работе показан способ построения терминальной системы на основе систем виртуализации. Такая архитектура позволяет проводить анализ сетевого трафика различного класса для пользовательских сеансов работы с использованием внешних аппаратных или программных терминалов. Применение виртуальных машин или изолированных контейнеров с уникальными сетевыми IP адресами, делает возможным программным способом идентифицировать сетевые пакеты и организовать их последующую обработку и анализ.

ЛИТЕРАТУРА

1. Рожков, С. А. Терминальные системы для предприятий / С. А. Рожков // Снежинск и наука 2006: сб. науч. тр. междунар. науч.-практ. конф. -Снежинск: СГФТА, 2006.— С. 184–186.
2. А.И.Гетьман, Е. Ф. Евстропов, Ю. В. Маркин. Анализ сетевого трафика в режиме реального времени: обзор прикладных задач, подходов и решений. Препринт ИСП РАН, том 28, 2015. 1–52 с
3. Таненбаум Э. Компьютерные сети: пер. с англ. СПб.: Питер, 2003. 992 с. (Сер. «Классика computer science»)
4. Спиряев О. «Терминальные решения на базе тонких клиентов» [Электронный ресурс] — Режим доступа: https://www.bytemag.ru/articles/detail.php?ID=8637 (дата обращения: 06.08.19).
5. X.Org Wiki [Электронный ресурс]. -http://www.x.org/wiki/ (дата обращения: 06.08.19).
6. Спецификация Microsoft на основные функции RDP [Электронный ресурс]. http://msdn.microsoft.com/en-us/library/cc240445(PROT.10).aspx. (дата обращения: 06.08.19). Понимая Docker. [Электронный ресурс]. https://habrahabr.ru/post/253877 (дата обращения: 15.11.2017).
7. Виртуальный Linux. [Электронный ресурс]. https://www.ibm.com/developerworks/ru/library/l-linuxvirt/ (дата обращения: 06.08.19).
8. Docker: запуск графических приложений в контейнерах. [Электронный ресурс]. https://habr.com/en/post/240509/ (дата обращения: 06.08.19).
9. Linux Bridge [Электронный ресурс]. http://xgu.ru/wiki/Linux_Bridge (дата обращения: 06.08.19).
10. Подсчёт трафика в Linux посредством ulog [Электронный ресурс]. https://www.opennet.ru/base/net/ulog_traf.txt.html (дата обращения: 06.08.19).

Журнал «Современная наука: актуальные проблемы теории и практики»