

АНАЛІЗ МОЖЛИВОСТЕЙ ВЗАЄМОДІЇ LINUX ТА WINDOWS В ГЕТЕРОГЕННОМУ МЕРЕЖЕВОМУ СЕРЕДОВИЩІ

У статті розглянуто можливості вирішення проблеми співіснування та ефективної взаємодії різномісних операційних систем в одній локальній мережі та запропоновано напрямки такої взаємодії з використанням поширених програмних продуктів.

In the article there have been considered the possibilities of solving up the problem of coexistence and effective interaction of the different type operation systems within the single local network, and proposed the directions of such interaction with applying the spread software.

Сьогодні існує багато поглядів на тему того, яке програмне забезпечення краще чи гірше. Умовно їх можна розділити на два основні табори: прихильники програмного забезпечення із закритим кодом та шанувальники продуктів напрямку open-source. Для уточнення можна сказати, що до перших відносяться переважно користувачі Windows-систем, а до других – користувачі Linux та Unix-подібних систем. На інтернет-форумах, присвячених перевагам тієї чи іншої системи, часто виникають масштабні словесні суперечки, що подекуди носять особистісний характер. Як правило, в цих суперечках беруть участь некваліфіковані користувачі та «сліпі» фанати тієї чи іншої системи. Тут ще можна згадати про рекламних агентів «під прикриттям», які вдають із себе щасливих користувачів якогось програмного продукту, або навпаки незадоволених і обманутих споживачів, які жодними методами не можуть добитися стабільної роботи від своєї операційної системи. А в цей час справжні професіонали заробляють свій шматок хліба на невігластві і сліпому фанатизмі простих користувачів. Треба зауважити, що і серед професійних користувачів є винятки, які підтверджують правило.

Мало спеціалістів може подивитися на проблему з іншого боку. При цьому варто згадати угоду 2006 року між двома серйозними гравцями на ринку програмного забезпечення – компаніями Microsoft та Novell – про патентне перемир'я та співробітництво з реалізації продукції один одного. Отже, якщо два софтверних гіганти можуть мирно співіснувати, то чому б не спробувати «примирити» такі різні операційні системи, як Windows та Linux в одній мережі. Більше того, варто заставити їх ефективно взаємодіяти на користь кінцевих користувачів, доповнюючи переваги суперника та перекриваючи недоліки один одного.

Для обох груп операційних систем розроблено багато корисного програмного забезпечення, в тому числі й безкоштовного. Для кінцевого споживача при професійному підході, як правило, має значення не тільки ціна програмного продукту, але й зручність розгортання, налаштування, можливості масштабування, звичність інтерфейсу користувача, наявність спеціалізованого програмного забезпечення для вирішення конкретних завдань, стабільність та безпека роботи, рівень технічної підтримки, сумарні витрати на експлуатацію системи протягом усього її життєвого циклу. З такої точки зору Unix-подібні системи зарекомендували себе у таких сферах інформаційних технологій, де критичними являються стабільність і безпека, безвідмовність, продуктивність – веб-технології, бази даних тощо. Продукти Microsoft знайшли свого користувача завдяки неперевершеній маркетинговій діяльності, простоті інтерфейсу користувача та банальному піратству, особливо на пострадянському просторі. Але на сьогоднішній день продукти лінійки Windows NT вже мають високі показники надійності, особливо при застосуванні сторонніх продуктів для безпеки, і втримують користувача чудовим інтерфейсом та тиском на виробників комп'ютерної техніки.

З другого боку не можна сказати, що ті чи інші платформи безкоштовні. Для серйозних комерційних завдань в будь-якому випадку прийдесться платити якщо не за самі програмні продукти, то за кваліфіковані кадри для роботи із цими системами.

Отже, маємо два ідеологічно різних програмних середовища з різними перевагами та недоліками і є задача реалізувати їх ефективне співіснування.

У цьому контексті можна виділити два основних напрямки взаємодії. По-перше, це взаємодія в межах одного комп'ютера, виконання програм, написаних для однієї операційної системи у середовищі іншої. По-друге, взаємодія різних операційних систем у мережевій роботі.

Для реалізації роботи різних операційних систем в межах одного комп'ютера є два основних шляхи – це віртуалізація апаратури в цілому (віртуальні машини з повною або квазіемуляцією апаратної частини) та емуляція API гостьової операційної системи (перехоплення системних викликів та моделювання поведінки гостьової операційної системи засобами хостової, що реально встановлена на комп'ютері, або сторонніми модулями). З точки зору користувача можна виділити ще роботу з аналогічним або кросплатформним прикладним програмним забезпеченням.

Програмні продукти, що в тій чи іншій мірі віртуалізують апаратні ресурси комп'ютера, мають наступні переваги. По-перше, це можливість запуску майже будь-якої операційної системи в середовищі іншої та робота з кількома операційними системами на одному комп'ютері одночасно. По-друге, це можливість емуляції різних апаратних архітектур. Недолік таких систем впливає із переваг – емуляція апаратури

програмним способом потребує додаткових накладних витрат продуктивності систем, особливо для реалізації різнотипної апаратної частини. Крім цього використання віртуальної машини не звільняє від оплати вартості ліцензії на її програмне забезпечення, в тому числі і системне. До таких систем віртуалізації відносяться такі найбільш відомі програмні продукти:

- 1) для гостьової ОС Windows – Microsoft Virtual PC/Virtual Server;
- 2) для Linux – Qemu;
- 3) кросплатформні – віртуальні машини компаній VMware та Parallels, серія xVM компанії Sun Microsystems тощо.

Емуляцію API (інтерфейсу прикладного програмування) використовують програмні продукти з іншим принципом дії. Вони не створюють відокремленого обчислювального середовища, а створюють у процесу, так би мовити, ілюзію того, що він працює у «рідному» операційному оточенні. Це досягається альтернативною реалізацією функцій та інших інструментів деякої гостьової операційної системи у спосіб, звичний для системи-хазяїна. Звідси випливають наступні переваги: висока швидкість роботи, часом навіть з можливістю підвищення продуктивності за рахунок використання більш ефективних алгоритмів та технологій програмування. Основним недоліком таких систем є невисокий ступінь підтримки ними прикладних програм через складність повної реалізації API будь-якої сучасної операційної системи, особливо із закритим кодом, а також необхідність реалізації API для підтримки кожної окремо взятої системи. До найбільш відомих продуктів такого типу відносять наступні програми:

- 1) Cygwin, andLinux – Unix-подібні середовища для Windows-систем;
- 2) Wine – вільна реалізація Win API для Unix-подібних систем (Linux, FreeBSD, Solaris, Mac OS).

Отже, було розглянуто можливості вживання Windows та Linux на одному комп'ютері. В будь-якому випадку очевидне домінування однієї з операційних систем – хостової, або системи-хазяїна, іншому операційному середовищу відводиться другорядна роль. При цьому гостьова система виступає неповноцінною через обмежену сумісність або втрачає велику частину своєї звичайної продуктивності. Але такі можливості є і можуть застосовуватися для підвищення ефективності використання комп'ютерної техніки.

Щодо спільної роботи у комп'ютерних мережах, то рівнів і способів взаємодії розглянутих операційних систем набагато більше. По-перше, це проста передача файлів між різнотипними операційними системами, по-друге, надання доступу до мережесхрещених ресурсів-сховищ інформації, принтерів тощо, по-третє, віддалений термінальний доступ.

З простою передачею файлів у принципі все ясно – протокол FTP підтримується великою кількістю серверних додатків (IIS, iPlanet, vsFTPd, ProFTPd), у тому числі кросплатформних (FileZilla), а клієнські частини включені до складу багатьох файлових менеджерів, навіть в примітивній реалізації до провідника Windows.

Набагато більше проблем виникає, коли необхідно реалізувати постійний доступ до файлових ресурсів мережі. Для досягнення такої мети є тільки один шлях – реалізація «рідних» мережесхрещених протоколів з однієї та/або іншої сторони. Тут задають правила гри власне самі ці протоколи. Для спільної роботи в мережі між комп'ютерами з операційними системами сімейства Windows NT використовується протокол SMB/CIFS, тоді як для Unix-подібних мережесхрещених середовищ характерне використання протоколу NFS.

«Рідний» для Windows протокол SMB/CIFS реалізований у проекті Samba. Цей програмний продукт працює на Linux та інших Unix-подібних системах і дозволяє їх користувачам отримувати доступ до файлових ресурсів Windows-машин та розділяти з ними свої мережесхрещених файлові сховища. Більше того, після не дуже складного налаштування Samba дозволяє організувати контролер домена мережі та файл-сервер, що керується стандартними утилітами NT tools, тобто виконувати функції аналогічні системам Windows Server за значно меншу ціну. Файл-сервер на базі Samba може входити в домен Active Directory, хоча й поки що не може виконувати всі функції контролера цієї файлової служби. Ще одна сильна сторона Samba – це можливість розділення доступу до мережесхрещених принтерів за допомогою служби друку CUPS, характерної для більшості Unix-подібних систем.

Компанія Microsoft також подбала про комфортний перехід користувачів Unix-подібних систем на платформу Windows NT. Для цього випущено програмний продукт Служби Windows для UNIX(SFU), або Interix. Ці служби дозволяють Windows-системам із серій Server та Professional взаємодіяти з Unix-системами по рідному для них протоколу NFS.

Ще одна можливість ефективної взаємодії даних систем полягає у віддаленому термінальному доступі. Ідея, відома ще з часів великих обчислювальних машин, перших мейнфреймів, знайшла свою реалізацію і в сучасних операційних системах. Суть її в тому, що всі обчислення багатьох користувачів виконуються на одному потужному комп'ютері серверного рівня, а клієнту передається тільки зображення, результати роботи, в свою чергу, клієнт передає серверу вхідні дані – відомості про натискання клавіш клавіатури та миші. Для того, щоб говорити про термінальний доступ між різними системами, слід розібратися зі способами його реалізації у «рідних» середовищах.

У Windows механізм термінального доступу реалізовано на основі протоколу RDP (Remote Desktop Protocol, протокол віддаленого робочого стола). Дві складові взаємодії – сервер та клієнт. Термінальний сервер являється складовою частиною всіх сучасних Windows систем, але повнофункціональним він є тільки у системах серверної серії. У персональних версіях цей сервер обмежений одним термінальним сеансом – «віддаленим помічником». Стандартний термінальний клієнт – mstsc.exe – «віддалений робочий стіл», також

можливе використання спеціальних пристроїв – тонких клієнтів. Фактично, для реалізації термінального доступу до Windows-систем, потрібна альтернативна реалізація термінальної програми. Такою реалізацією для Linux та інших Unix-сумісних систем є, наприклад, Rdesktop. Ця програма реалізує майже всі можливості версії протоколу RDP 5, крім перенаправлення USB-портів. З іншого боку, Rdesktop має самостійні функції, нереалізовані у стандартних термінальних клієнтах. Це так званий «безшовний» режим, коли на сторону клієнта передається не все зображення робочого столу, а лише картинка області вікна заданого додатку, при цьому виникає ілюзія роботи з Windows-програмою на Unix-подібній системі.

В реалізації термінального режиму для Linux все не так просто.

В Unix-подібних операційних системах як графічна оболонка застосовується система X Window, у якій є багато різних реалізацій – комерційних, напівкомерційних і т. д. Що стосується Linux, то на даний момент в основному використовуються XFree86 й xorg. Через проблеми з ліцензуванням перша система відсутня в сучасних дистрибутивах Linux, тому все частіше зустрічається xorg.

X Window побудована за принципом клієнт-сервер. Однак принцип взаємовідносин клієнта й сервера поставлений «з ніг на голову». Звичайно під сервером або серверним додатком розуміють яке-небудь програмне забезпечення, що реалізує зберігання інформації або складні обчислення. А клієнт – це програма, що дозволяє здійснювати керування або одержувати доступ до інформації. Вище сказане стосується і механізму термінального доступу для Windows. В X Window сервер – програма для промальовування зображення на екрані, а призначення клієнта – відображати дані на сервері, з використанням функцій бібліотеки Xlib. Таким чином, сервер X Window містить драйвери відеокарти, клавіатури, миші й інші пристрої введення й дозволяє клієнтським програмам «малювати» на екрані монітора. Клієнт і сервер можуть перебувати на різних машинах у мережі. Більше того, дозволяється, щоб на одному сервері «малювали» кілька клієнтів, кожен з яких виконується на різних машинах.

Програму з функціями X-сервера можна розташувати на окремому пристрої або в іншій системі в мережі. Нехай то спеціальний пристрій, так званий X-термінал, чи програма для будь-якої операційної системи (у тому числі Windows), аби тільки вони забезпечували функціональність X-сервера. Потенційно дані на X-сервері здатний відображати будь-який X-клієнт, але звичайно вводять обмеження й вказують конкретні клієнти.

Для промальовування вікон використовується спеціальне програмне забезпечення – віконний менеджер (Window Manager). Від нього залежить, як виглядають вікна і як вони поведуться. Віконний менеджер теж X-клієнт, оскільки для відображення вікон він посилає дані на X-сервер. Деякі програми X-сервера, що працюють під керуванням Windows, мають вбудований віконний менеджер.

Ще одна важлива для X Window програма – менеджер дисплеїв (Display Manager). X-термінал – це пристрій, на якому немає жодної програми. Всі програми (X-клієнти) виконуються на яких-небудь машинах, а на X-терміналі тільки відображаються дані. Після включення X-терміналу підключається по протоколу X Display Manager Connect Protocol (XDMCP) до сервера, де встановлений менеджер дисплеїв, за допомогою якого клієнт здійснює вхід у систему й запускає віконний менеджер. Так ми одержуємо повноцінне робоче місце на машині Linux з відображенням робочого столу на машині Windows. Всі програми (включаючи віконний менеджер) будуть виконуватися на Linux, а дані представити Windows.

Сучасні програми, що реалізують функції X-сервера, дозволяють повністю емулювати X-термінал, але вони можуть обійтися й без емуляції. Якщо немає необхідності відображати весь робочий стіл Linux, конкретні програми можна відтворювати в окремому вікні Windows, запускаючи їх на машині Linux, щоправда, однаково доведеться заводити на ній обліковий запис користувача й виводити приналежні йому дані на X-сервер.

Реалізація X-сервера для Windows.

Роботу X-сервера на машині Windows реалізують кілька різних програм. Серед них варто виділити безкоштовний сервер Gygwin/X (<http://x.cygwin.com>). Cygwin/X – це фактично перенесення X Window System на різні види ОС Windows. Продукт працює на всіх сучасних системах Windows. Cygwin/X містить X-сервер, X-бібліотеку і набір стандартних X-клієнтів: xterm, xhost, xclock й ін.

Для прикладу варто розглянути схему розгортання термінальної роботи у Linux з Windows-системи на базі Gygwin/X. Перший крок – підготовка машини з Linux.

Для того, щоб X-сервери могли взаємодіяти з машиною Linux, на ній треба, по-перше, завести облікові записи користувачів, а по-друге, відповідним чином налаштувати програму менеджера дисплеїв, якщо планується використовувати повноцінний вхід у систему (на зразок X-терміналу). З менеджерами дисплеїв ситуація трохи складніша, оскільки з Linux можуть поставлятися щонайменше три менеджери: xdm, gdm (Gnome) і kdm (KDE). Кожний налаштовується різними способами.

Наступний етап – установка й конфігурація GYGWIN/X.

Для установки Gygwin/X досить відкрити головну сторінку сайту й завантажити програму установки setup.exe. Програму можна встановити прямо з Internet або з каталогу, куди вона була збережена заздалегідь. Після установки Gygwin/X необхідно налаштувати. Конфігураційні параметри передаються Gygwin/X при запуску програми XWin.exe у її командному рядку. Для полегшення запуску поставляється спеціальний файл startxwin.bat.

І, нарешті, робота з CYGWIN/X. Після установки програми на робочому столі створюється ярлик Cygwin, за допомогою якого запускається оболонка bash, так що можна працювати зі знайомим командним рядком Linux. З її допомогою можна запустити X-сервер у режимі X-терміналу, наприклад: X-query

192.168.100.11. Замість цієї IP-адреси треба вказати IP адресу комп'ютера, де перебуває менеджер дисплеїв kdm. На екрані повинне з'явитися запрошення увійти в систему на машині Linux. Після входу активізується віконний менеджер, і тепер можна працювати на машині Linux з відображенням даних в Windows.

Звичайно, такий метод досить ресурсоємний, тому при великій кількості клієнтів для Linux варто встановити потужний комп'ютер. Втім, Gygwin/X припускає й інший режим, коли на машині Windows запускається X-сервер, власний віконний менеджер і програма емуляції терміналу. У цьому випадку для запуску використовується файл startxwin.bat. Завдяки параметру -multiwindow, що передається при запуску XWin.exe, кожен додаток буде відкриватися у своєму вікні.

При використанні startxwin.bat не відбувається підключення X-сервера як X-терміналу, і тому на машині Linux не потрібно налаштовувати менеджер дисплеїв.

Після запуску startxwin.bat з'являється вікно програми емуляції терміналу. Спочатку необхідно дозволити віддаленим програмам відображати дані на X-сервері, що працює в Windows. Для цих цілей можна скористатися програмою xhost: xhost +192.168.100.11.

Потім необхідно зайти на машину з Linux за допомогою програми ssh: ssh user@192.168.100.11 і вказати ім'я користувача машини Linux й її IP-адресу.

Для того, щоб програми, що запускаються на Linux, відображали свої дані у вікні на машині Windows, можна використати змінну оточення DISPLAY: export DISPLAY=192.168.100.10:0.0. Замість цієї IP-адреси, треба використати IP-адресу машини Windows, на якій працює X-сервер. Програми запускаються звичайним способом.

Незважаючи на те що X-сервер від Gygwin вимагає установки майже всього базового середовища запуску, він працює відмінно. І це не дивно, адже він являє собою стандартну реалізацію сервера xorg, зібрану в середовищі gygwin. Отже, при конфігурації сервера застосовуються ті ж прийоми, що й в Linux.

На основі вище сказаного можна зробити очевидний висновок, що взаємодія Windows- і Linux-систем не просто можлива, а необхідна для отримання максимального ефекту від використання обидвох сімейств операційних систем одночасно. Це у великій мірі стосується й інших Unix-подібних операційних систем (наприклад FreeBSD, Solaris тощо). По-перше, така взаємодія може дати можливість підвищення ефективності використання апаратних ресурсів. По-друге, сукупне здешевлення парку програмного забезпечення. По-третє, розумне сумісне застосування різних систем може створити принципово нові ідеї в області інформаційних технологій. Ці три причини є серйозним аргументом для припинення суперечок про те, яка система краща. Натомість краще використовувати переваги кожної з цих операційних систем окремо та у спільній мережевій роботі.

Список використаних джерел

1. LAN. – 2005. – № 8.
2. Родерик В. Смит. Сетевые средства Linux / Родерик В. Смит. – М. : Вильямс, 2003. – С. 672.
3. Харари Ф. Теория графов / Харари Ф. – М. : Мир, 1973. – С. 458.
4. <http://ko-online.com.ua/node/26236>
5. <http://us1.samba.org/samba/docs/man/Samba-HOWTO-Collection/>
6. <http://www.rdesktop.org/>
7. http://www.osp.ru/lan/2002/12/136939/_p1.html
8. <http://stfw.ru/page.php?id=5726>
9. Сети и системы связи. – 2005. – № 11.
10. <http://www.linuxrsp.ru/win-lin-soft/>
11. <http://linux-ve.chat.ru/book/Win-emu.htm>
12. <http://wikipedia.org>