Технология представления результатов сетевого мониторинга в Интернете. Системная служба

Терентьев Александр Макарович, кандидат технических наук, PhD
Центральный экономико-математический институт Российской академии наук (г.Москва)

Аннотация. *В статье приведена история построения и развития сетевого мониторинга в ЦЭМИ РАН. Сформулирована проблема передачи информации от обычного Windows-приложения на web-сервер. Подробно показан пример построения системной службы сервера на языке высокого уровня PowerBASIC.*

Ключевые слова: *сетевой мониторинг, информационная безопасность, UDP, web-сервер, системная служба*

Идея круглосуточного низкоуровневого сетевого мониторинга с использованием постоянно работающей наблюдающей станции (НС) в качестве одного из основных средств информационной безопасности была высказана автором достаточно давно [1], сразу после реализации пилотного проекта [2] программных средств.

Возможность дампировать на жёсткий диск сетевые пакеты и затем спокойно разобраться побайтно с их составом сразу же очертила сферы многочисленных возможных приложений сетевого мониторинга [3], фактически превращая его в средство круглосуточного аудита локальной вычислительной сети (ЛВС). После получения гранта РФФИ 04-07-90260, программные средства были доведены до уровня промышленной разработки [4-6]. Это позволило провести многочисленные разноплановые эксперименты – выявить «узкие места» в ЛВС ЦЭМИ РАН и выделить её часть в специфицированный сегмент сети [7], провести экспресс-мониторинг в сети иного предприятия с выявлением нарушителя корпоративных правил [8], провести теоретические исследования структуры ARP-пакетов и сформированных с их девиацией запросов [9], исследовать характеристики протоколов обращающихся по сеансам POP3 и FTP пользователей [10], выявить точное подмножество протокола TelNet при обращении к коммутаторам Cisco Catalyst [11-12], ввести суммарный биллинг ЛВС ЦЭМИ РАН за год [13], уточнить детали протокола выдачи антивирусных обновлений пакета Doctor Web пользователям [14] и многие другие. Без преувеличения можно сказать, что развитие системы сетевого мониторинга в ЦЭМИ РАН помогло решить целый ряд крупных задач, в том числе дефрагментацию ЛВС [7], автоматическое исключение зараженных ПК из ЛВС с помощью управления коммутатором Cisco Catalyst [15] и корпоративный вариант технологии снабжения пользователей научных учреждений антивирусными средствами [16-18].

Финальный вариант по упомянутому выше гранту РФФИ был описан в [6] и [15]. Особенности мониторинга в условиях, когда центральный хаб заменён на современный коммутатор Cisco Catalyst, приведены в [19]. Некоторые проблемные следствия этого изменения показаны в [14]. Дополнительные действия в связи с решением проблем освещены в [22].

В то время представлялось, что дальнейшее развитие сетевого мониторинга не имеет перспективных направлений. Тем более, что изменилась структура ЛВС с выделением из ЛВС «внутренней» части, в которой функция отсекания заражённых ПК неприменима вследствие организационных особенностей ЦЭМИ РАН. Однако, это не исключало значения ряда функций основной НС, находящейся по-прежнему в локальной сети с истинными IP-адресами. Прежде всего, это биллинговые функции подсчёта трафика «Из Интернета» и «В Интернет», как показано в [13]. Эти данные остаются адекватными, поскольку весь трафик «внутренней ЛВС» учитывается как исходящий от роутера внутренней ЛВС, имеющего внешний реальный IP-адрес, известный для основной НС. После того, как была налажена агрегация данных НС в мониторной програме (МП) соседнего Windows-компьютера [6], соединённого с НС serial-кабелем, естественно, появилось желание представить эти агрегированные данные в Интернете. Первый вариант связи МП↔Сервер с помощью UDP описан в [21]. В этот момент серверную часть исполняло обычное Windows-приложение.

Скоро было замечено, что такое решение неоптимально для сервера: для начала работы обычного приложения на сервере необходимо выполнить процедуру входа в систему пользователя (Logon), что не всегда отвечает требованиям безопасности сервера. При такой эксплуатации сервер становится общедоступным для физически находящегося рядом с ним персонала и случайных посетителей, что недопустимо. Поэтому, опираясь на уже сделанную работу, встала задача обеспечить взаимосвязь между компьютером с МП и сервером на новом, закрытом от пользователей и случайных людей, уровне.

Для исполнения этого, Windows-приложение на сервере было заменено на 3 программы:

– системную службу TAM LAN Server Service, исполняющую UDP-связь с МП и периодически сбрасывающую актуальную информацию на диск в виде текстового файла определённого формата;

– программу управления вышеописанной системной службой TAM LAN Server Control, являющуюся обычным Windows-приложением и работающую только после исполнения входа в систему (Logon) администратора;

– CGI-программу формирования выходной формы по состоянию текстового файла на диске, оставшуюся от прежнего варианта связи МП ↔ Сервер.

Преимущество такой схемы состоит в том, что системная служба начинает работу сразу же при загрузке сервера, до выполнения процедуры Logon [20]. Программа управления системной службой может быть не загружена, что не мешает работе системной службы и исполнению передачи оперативных данных на сервер. Таким образом, функционирование сервера уже не зависит от места его установки: при размещении сервера в специально выделенном месте процедура Logon, как правило, не выполняется.

Системная служба выполнена на хорошо известном мировым профессионалам-программистам языке PowerBASIC [23], позволяющим создавать эффективные приложения как консольного типа (компилятор PowerBASIC Console Compiler, автор использует версию 5.05), так и

полноценные GUI-приложения (компилятор PowerBASIC for Windows 9.05). Являясь мощной альтернативой языку среднего уровня C++, PowerBASIC предлагает язык высокого уровня, специфицированный для создания Windows-приложений благодаря встроенной поддержке множества обращений к различным библиотекам Windows. Собственно говоря, все модули сетевого мониторинга с начала его создания (1999г) были написаны на различных диалектах этого языка, при необходимости дополняясь процедурами на Макроассемблере [5]. Так, модули HC были написаны на PowerBASIC for DOS. Все версии компиляторов являются платными (порядка $100 за каждую), лицензирование осуществляется вендором исключительно персонально для каждого пользователя.

Общая структура системной службы как консольного приложения показана на рис. 1. Отметим, что на приведённом рисунке показан именно «скелет» программы; в целях экономии места многие тривиальные функции типа протоколирования работы (подпрограмма AddLog) и занесения/чтения ключей реестра не отражены. Опущены также многочисленные проверки корректности исполнения вызовов различных подпрограмм Windows, которые в реальной программе, разумеется, присутствуют.

Из рис.1 видно, что запуск и исполнение системной службы реализуется в виде обращения ко входу StartServiceCtrlDispatcher (соответствующее определение которого, как и многие другие, находятся в сопровождающей компиляторы библиотеке Win32API.INC). Одним из параметров передаётся адрес участка программы ServiceMain, собственно, и реализующий службу.

Большинство задающих операций исполняется с помощью обращений ко входящему в состав Windows так называемому Service Control Manager (SCM).

Windows вызывает для исполнения службы функцию ServiceMain, которая, в свою очередь, управляет необходимыми вызовами для инициализации службы, стартует службу и затем ожидает завершения службы. Параметры для этой функции могут передавать эквивалент командной строки. При задании максимума в миллисекундах с помощью g_ServiceStatus.dwWaitHint следует принять во внимание, что необходимо выполнить вызов SetServiceStatus за данный промежуток, иначе SCM посчитает инициализацию службы неудавшейся.

В тексте программы сохранены все обращения к SCM для организации паузы исполнения, продолжения работы, останова.

```
'===== TAMOBSS = Серверная часть LAN Audit как системная служба. А.Терентьев
#COMPILE EXE
#INCLUDE "Win32API.inc"
%UPort   = 150353       ' Номер используемого UDP-порта
GLOBAL g_fRunning       AS LONG      ' служба зарущена и работает
GLOBAL g_hEvent         AS DWORD     ' Event- handler работающей службы
GLOBAL g_hService       AS DWORD     ' service status handle
GLOBAL g_sServiceName   AS STRING    ' Имя службы
GLOBAL g_ServiceStatus AS SERVICE_STATUS  ' Статус службы (service status)
.........
'===================================== Вспомогательные функции
FUNCTION SetRegValue(lpKey AS LONG, BYVAL cMainkey AS STRING,_   ' Установка клю-
      ча в реестре
           BYVAL sKey AS STRING, BYVAL Setting AS STRING) AS LONG
.........
END FUNCTION
SUB Signal (Setting AS STRING)
  LOCAL lpKey, fOk AS LONG, cMainKey,sKey AS STRING
  lpKey=%HKEY_LOCAL_MACHINE
  cMainKey="SYSTEM\CurrentControlSet\Services\TAMObService"
  sKey="Signal": fOk=SetRegValue(lpKey,cMainKey,sKey,Setting): END SUB
.........' Прочие функции не указываются вследствие недостатка места
' ServiceProc - тред, параллельно с основным процессом. Делает основную работу.
' ====================================================================================
FUNCTION ServiceProc (BYVAL unused AS LONG) AS LONG
LOCAL ix AS LONG
.........
 DO WHILE g_fRunning
   SELECT CASE iPause
    CASE 0      ' ----- Нормальное продолжение программы
      GOTO WCGo
    CASE 1      ' ----- Pause Pending, Подан сигнал Pause
      g_ServiceStatus.dwCurrentState = %SERVICE_PAUSED 'Tel the SCM pause
      g_ServiceStatus.dwWaitHint     = 5000
      SetServiceStatus g_hService, g_ServiceStatus
      iPause = 2 ' Set "pause" internal flag
      SetEvent g_hEvent
    CASE 2      ' ----- Pause, Отработан сигнал Pause
      GOTO WCEnd
    CASE 3      ' ----- Continued
      g_ServiceStatus.dwCurrentState = %SERVICE_RUNNING
```

```
      g_ServiceStatus.dwWaitHint     = 5000
      SetServiceStatus g_hService, g_ServiceStatus
      iPause = 0 ' Set "pause" internal flag
      SetEvent g_hEvent: GOTO WCGo
   END SELECT
WCGo:
   SELECT CASE iCond
    CASE 0 ' ------------------------------ Начальные установки
      HOST ADDR TO ip' Get this machines IP address ` Принять свой IP
      ERRCLEAR: UDP OPEN PORT %UPort AS %hUdp TIMEOUT %UDPTime
.....
      iCond=1: EXIT SELECT
    CASE 1 ' ------------------------------ Слушание порта, ожидание сигнала
      ERRCLEAR                    ' Начало слушания UDP/IP port
      UDP RECV #%hUdp, FROM ipAddr, ipPort, Buffer ` IP-адрес заполняется при
      приёме
      IF MID$(Buffer,1,3)<>"===" THEN ` Проверка корректности обращения к серве-
      ру
        EXIT SELECT
      END IF
      Reply = "***" + $AppTitle  ' Подготовка ответа "Ок, принято"
      UDP SEND #%hUdp, AT ipAddr, ipPort,Reply
      tCond = TIMER: iCond = 2: tFile=0
    CASE 2 ' ------------------------------ Исполняемые действия
      ERRCLEAR: UDP RECV #%hUdp, FROM ipAddr, ipPort, Buffer ` Приём очередного
      пакета от МП
      IF MID$(Buffer,1,3) <> "=*=" THEN ' ---- Некорректен заголовок пакета
         iCond=1: EXIT SELECT
      END IF: tCond=TIMER
      sIdBuf=MID$(Buffer,5,55) ' Выделение нужного участка буфера, от тек.даты
      до CHR$(22) вкл.
      CurTime: Reply = CDat3+","+$VM
      UDP SEND #%hUdp, AT ipAddr, ipPort, "*=* "+Reply ' Подтверждение handshake
      IF MID$(Buffer,4,1) <> " " THEN ' Утеряна связь с НС
         sMaBuf="": sAdBuf="": sMCBuf="": iASReady=0
      ELSE
         iASReady=1
      END IF
     ` Здесь - Обработка информации из Buffer
.....
      IF TIMER-tFile>5 OR TIMER-tFile<0 THEN
         ERRCLEAR: OPEN $DatFile FOR OUTPUT ACCESS WRITE AS #3
.....
      END IF
    END SELECT
WCEnd:      ' Здесь ждём несколько секунд, но только 1 секунду за раз.
            ' Это позволяет службе ответить на "shutdown"-запрос
    ix = 2: DO WHILE g_fRunning AND ix: SLEEP 1000: DECR ix: LOOP
 LOOP: CLOSE #%hUdp: AddLog 2,"ServiceProc Ended": SETEOF#1.....
END FUNCTION
SUB ServiceCtrlHandler (BYVAL dControlCode AS DWORD)
    g_ServiceStatus.dwCheckpoint = 0
    SELECT CASE dControlCode
    CASE %SERVICE_CONTROL_STOP
.....
        g_ServiceStatus.dwCurrentState = %SERVICE_STOP_PENDING ' Сообщить SCM
      что будем ост. службу
        g_ServiceStatus.dwWaitHint     = 5000
        SetServiceStatus g_hService, g_ServiceStatus
        g_fRunning = 0      ' Сброс "running"-флага
        SetEvent g_hEvent  ' Завершение
    CASE  %SERVICE_CONTROL_SHUTDOWN
..... ` Аналогично предыдущему
    CASE %SERVICE_CONTROL_PAUSE
```

```
        g_ServiceStatus.dwCurrentState = %SERVICE_PAUSE_PENDING
        iPause = 0: SetEvent g_hEvent
    CASE %SERVICE_CONTROL_CONTINUE
.....
    END SELECT: END SUB
SUB ServiceMain (BYVAL dwArgc AS DWORD, BYVAL lpszArgv AS DWORD)
    LOCAL hThread AS DWORD, LOCAL nJunk   AS LONG
    g_hEvent = CreateEvent(BYVAL 0, 1, 0, BYVAL 0)
         ' Create a service termination event
    g_hService = RegisterServiceCtrlHandler(BYVAL STRPTR(g_sServiceName),
      CODEPTR(ServiceCtrlHandler)) ' Регистрация службы. - Имя и начало
      CtrlHandler.
    g_ServiceStatus.dwServiceType  = %SERVICE_WIN32_OWN_PROCESS' Define the type
      of service.
    g_ServiceStatus.dwWaitHint = 2000
        ' Максимум мс ожидания SCM перед SetServiceStatus
    g_ServiceStatus.dwControlsAccepted = 7
    g_ServiceStatus.dwCurrentState = %SERVICE_START_PENDING  ' Определить теку-
      щий статус службы
    SetServiceStatus g_hService, g_ServiceStatus
    INCR g_ServiceStatus.dwCheckpoint
' Здесь команды инициализации службы (если есть)
' Помнить, что вызов SetServiceStatus необходим для поддержания актуальности
      SCM. Увеличивать g_ServiceStatus.dwCheckpoint каждый раз.
    g_fRunning = -1 ' Предохраняет от несвоевременного закрытия треда
    THREAD CREATE ServiceProc(0) TO hThread  ' Запуск параллельного треда
    ' Задать тип control events, которые Ваша служба способна обрабатывать.
    g_ServiceStatus.dwControlsAccepted = %SERVICE_ACCEPT_STOP OR %SER-
      VICE_ACCEPT_SHUTDOWN
    g_ServiceStatus.dwCurrentState = %SERVICE_RUNNING ' Сообщить the SCM, что
      служба запущена.
    g_ServiceStatus.dwCheckpoint = 0
    SetServiceStatus g_hService, g_ServiceStatus
    AddLog 2,"ServiceMain: SCM Notified that Service is Running"
    WaitForSingleObject g_hEvent, %INFINITE
    DO: THREAD CLOSE hThread TO nJunk: SLEEP 50: LOOP UNTIL ISTRUE nJunk
    g_ServiceStatus.dwCurrentState = %SERVICE_STOPPED ' Сообщить SCM что служба
      завершена.
    SetServiceStatus g_hService, g_ServiceStatus
END SUB
FUNCTION PBMAIN () AS LONG    '  Основная программа – только запускает службу
    DIM ste(0 TO 1) AS LOCAL SERVICE_TABLE_ENTRY
    DIM sArr(1 TO 14)
    g_sServiceName = $SERVICENAME
    ste(0).lpServiceName = STRPTR(g_sServiceName)
    ste(0).lpServiceProc = CODEPTR(ServiceMain)
    StartServiceCtrlDispatcher BYVAL VARPTR(ste(0))  ' Старт и исполнение службы
    AddLog 3,"X "+$AppTitle+" is ended"  ' Строки, завершающие службу
    SETEOF#1: CLOSE#1
END FUNCTION
```

Рис. 1. Скелет программы системной службы TAMObService

Программистам рекомендуем обратить внимание на процедуру Signal, которая сигнализирует о получении очередной порции данных управляющей программе службы, если она вызвана, для синхронизации их показа на экране. Для связи используется ветка реестра, специально предусмотренная для служб SYSTEM\CurrentControlSet\Services\TAMObService\Setting. Занесение значения «Signal» отслеживается программой управления службой.

Оператор "UDP OPEN PORT" означает открытие соответствующего порта как серверного (ведущего) с последующим прослушиванием. При обнаружении (UDP RECV) сигнала от какого-либо ПК по этому порту определяется его IP и посылается ответный пакет (UDP SEND). Таким образом, после успешного handshake устанавливается постоянная связь по UDP между сервером и МП. В качестве номера порта выбрана дата рождения автора, хотя это может быть и другой номер из числа разрешённых.

Получение информации в МП от НС идёт ежесекундно. Соединение МП↔Сервер срабатывает каждые 5 секунд. Системная служба обновляет рабочий файл на сер-

вере синхронно с получением очередной информации по UDP.

Актуальный на момент первоначального написания текст программы зарегистрирован в ФИПС. Эксплуатация программы проверена в средах Windows 2000 Server и Windows Server 2003 R2.

Следует отметить, однако, что автор не претендует на открытие собственной техники программирования системных служб: часть приведённых программных решений заимствована из многолетних обсуждений на международном форуме программистов по языку PowerBASIC. Конечно, эти решения адаптированы автором под требования конкретной реализации и использованные версии компиляторов.

Данная статья является первой в цикле описания технологии передачи информации НС на сервер через МП.

Большинство работ автора, приведённых в списке Литературы, доступны для чтения в разделе «Литература» Антивирусного сайта ЦЭМИ РАН [24].

Литература:

1. Терентьев А.М. Информационная безопасность в крупных локальных сетях. // «Концепции», N1(9)-2002, с.25-30. Свидетельство Роскомпечати 014305.

2. Терентьев А.М. Методы и средства наблюдения загрузки локальных вычислительных сетей на примере ЦЭМИ РАН. / Препринт #WP/2001/110 - М., ЦЭМИ РАН, 2001, - 74с. ISBN 5-8211-0141-7.

3. Терентьев А.М. Задачи полноценного аудита корпоративных сетей // «Концепции», N1(11)-2003, с.94-95. Свидетельство Роскомпечати 014305.

4. Терентьев А.М. Построение и развитие системы сетевого мониторинга. / Развитие и использование средств сетевого мониторинга. Вып.1 Сб. статей под ред. А.М.Терентьева. – М.: ЦЭМИ РАН, 2004, с.5-23. ISBN 5-8211-0317-7.

5. Терентьев А.М. Ускорение форматных преобразований в системах реального времени, реализованных на языке PowerBASIC для i386+. / Развитие и использование средств сетевого мониторинга. Вып.1 Сб. статей под ред. А.М.Терентьева. – М.: ЦЭМИ РАН, 2004, с.24-36. ISBN 5-8211-0317-7.

6. Терентьев А.М. Мониторная программа как средство интеграции данных наблюдающей станции в локальной сети. / Развитие и использование средств сетевого мониторинга. Вып.2. Сб. статей под ред. М.Д.Ильменского. – М.: ЦЭМИ РАН, 2005, с.6-13. ISBN 5-8211-0365-7.

7. Вегнер В.А., Ляпичева Н.Г., Львова А.С., Терентьев А.М. Разработка и реализация типового проекта выделенного сегмента ЛВС на примере ПК административно-финансовой группы ЦЭМИ РАН. / Развитие и использование средств сетевого мониторинга. Вып.1. Сб. статей под ред. А.М.Терентьева. – М.: ЦЭМИ РАН, 2004, с.88-101. ISBN 5-8211-0317-7.

8. Терентьев А.М. Опыт сетевого экспресс-мониторинга с помощью переносной наблюдающей станции. / Развитие и использование средств сетевого мониторинга. Вып.1. Сб. статей под ред. А.М.Терентьева. – М.: ЦЭМИ РАН, 2004, с.41-46. ISBN 5-8211-0317-7.

9. Терентьев А.М. Об одной побочной возможности использования ARP-пакетов. / Развитие и использование средств сетевого мониторинга. Вып.1. Сб. статей под ред. А.М.Терентьева. – М.: ЦЭМИ РАН, 2004, с.37-40. ISBN 5-8211-0317-7.

10. Ляпичева Н.Г., Терентьев А.М. Исследование сетевых сервисов на примере клиентского почтового протокола POP3. / Развитие и использование средств сетевого мониторинга. Вып.1. Сб. статей под ред. А.М.Терентьева. – М.: ЦЭМИ РАН, 2004, с.60-74. ISBN 5-8211-0317-7.

11. Терентьев А.М. Возможность полуавтоматического управления сетевыми коммутаторами Cisco Catalyst. / Развитие и использование средств сетевого мониторинга. Вып.2. Сб. статей под ред. М.Д.Ильменского. – М.: ЦЭМИ РАН, 2005, с.14-27. ISBN 5-8211-0365-7.

12. Терентьев А.М. Консоль управления сетевыми коммутаторами Cisco. / Развитие технологий и инструментальных средств информационной безопасности. Вып.1. Сб. статей под ред. А.М.Терентьева. – М.: ЦЭМИ РАН, 2010, с.6-20. ISBN 978-5-8211-0543-1.

13. Ляпичева Н.Г. Выбор Интернет-провайдера на основе измерения трафика / Развитие и использование средств сетевого мониторинга и аудита. Вып. 3. Сб. статей под ред. А.М.Терентьева. – М.: ЦЭМИ РАН, 2006, с.36-49. ISBN 5-8211-0409-2 (978-5-8211-0409-0).

14. Ляпичева Н.Г., Акиншин А.А., Терентьев А.М., Григорьев П.В. Коррекция ошибок HTTP-соединения в локальной сети ЦЭМИ РАН. / Развитие технологий и инструментальных средств информационной безопасности. Вып. 2. Сборник статей под ред. А.М.Терентьева. – М.: ЦЭМИ РАН, 2012. – С.49-58.

15. Терентьев А.М. Автоматическая изоляция некорректных объектов КВС по информации сетевого мониторинга. / Развитие технологий и инструментальных средств информационной безопасности. Вып. 3. Сборник статей под ред. А.М.Терентьева. – М.: ЦЭМИ РАН, 2012. – С.6-15. ISBN 5-8211-0409-2 (978-5-8211-0409-0).

16. Терентьев А.М. Корпоративный вариант реализации антивирусных пакетов Doctor Web в научных учреждениях: предпосылки. // «Национальные интересы: приоритеты и безопасность». – М.:Издательский дом «Финансы и кредит», N17(206), 2013, с.41-48, ISSN 2073-2872.

17. Терентьев А.М. Корпоративный вариант реализации антивирусных пакетов Doctor Web в научных учреждениях: реализация. // «Национальные интересы: приоритеты и безопасность». – М.:Издательский дом «Финансы и кредит», N19(208), 2013, с.40-45, ISSN 2073-2872.

18. Терентьев А.М. Корпоративный вариант реализации антивирусных пакетов Doctor Web в научных учреждениях: результаты. // «Национальные интересы: приоритеты и безопасность». – М.:Издательский дом «Финансы и кредит», N20(209), 2013, с.41-46, ISSN 2073-2872.

19. Терентьев А.М., Ляпичева Н.Г., Кочетова Н.А. Мониторинг корпоративной сети ЦЭМИ РАН в условиях использования коммутатора Cisco Catalyst. / Развитие и использование средств сетевого мониторинга и аудита. Вып.1. Сб. статей под ред. А.М.Терентьева. – М.: ЦЭМИ РАН, 2004, с.60-74. ISBN 5-8211-0317-7.

20. Microsoft Windows 2000 Server. Справочник администратора. 2-е изд. / Пер. с англ., - М.: Издательство «СП ЭКОМ», 2003. – 1360 с.: ил. ISBN 5-9570-0002-7.

21. Терентьев А.М., Львова А.С. Адекватное отображение на технологическом www-сервере событий реального времени. / Развитие и использование средств сетевого мониторинга и аудита. Вып.3. Сб. статей под ред. А.М.Терентьева. – М.: ЦЭМИ РАН, 2006, с.50-59. ISBN 5-8211-0409-2.

22. Терентьев А.М. Поддержание доступности HTTP-соединения с помощью периодического пингования. // XXIV Международная научная конференция «Современные концепции научных исследований» – М.: «Евразийское научное объединение», N2(24), 2017. Т.1, с.37-39. ISSN 2411-1899.

23. [Интернет-ресурс] Международный сайт языка программирования PowerBASIC: http://www.powerbasic.com.

24. [Интернет-ресурс] Антивирусный сайт ЦЭМИ РАН: http://av.cemi.rssi.ru