Chapter 7
Working with Guardant Net in LAN

This Chapter contains tips on how to work with Guardant Net protection in local networks. You will learn about local networks in which protected products can be run, how they interact with Guardant Net dongle and how the license licenses are allocated. You will learn to configure Guardant Net servers and find out about the importance of network monitors. Finally, you will be advised on the issues of making network communication fault-free. This information is not related to the protection issue directly. It is rather intended for network administrators. **We recommend that you take the most important tips from below and include them into your software Manual.**

Important
Please pay special attention to the mechanism of allocating network licenses and to the tips on increasing the reliability of network communication.

Guardant Net Concept

Let us first investigate the basic principles of Guardant protection. This knowledge will help you understand why you should work with Guardant Net in the way prescribed in this chapter.

What is Guardant Net?

Guardant Net is a Guardant Stealth dongle specially adapted for local networks. It ensures not only protection but also licensing of network software*. The idea of licensing may be formulated as that of exercising control over the number of copies of network products that run simultaneously on a network. The objective of licensing is to prevent running of more copies than allowed. The network license limit (i.e., the maximum permitted number of workstations for the network product) is stored in the memory of Guardant Net in the field of counter #2.

For full protection and licensing of your network product, one Guardant Net dongle is enough for the entire local network. It can be installed either on a workstation or on a server.

What is Guardant Net Server?

Protected network products have no ability to communicate with the network dongle directly. No network protocol can provide for it. A special utility called Guardant Net server provides a link between the client (i.e., protected application) and Guardant Net dongle. It is utility providing for transmission of queries from the client directly to the dongle and backwards in accordance with the network protocol.

This brings about the main rule of loading Guardant Net server.

Important

Guardant Net server must be loaded on the same workstation or server to which Guardant Net dongle is attached. Otherwise, the server (as well as the Guardant Net client) will be unable to detect the dongle and serve as a link between this dongle and the client.

How does Guardant Net Work?

When started, Guardant Net server reads network license limits and other parameters of all dongles attached to the computer and stores them. To start working with the dongle the protected application should be logged in the server. Logging in is carried out by Login operation. During its execution, the server verifies if the dongle with required parameters is attached to the computer, and decrements its license counter by 1. Otherwise, it returns an error code to the client stating that 'The dongle is not found'. After login is completed successfully, the application can execute any permitted operations with the dongle. When the application terminates it logs out using Logout operation. During its execution, the license counter of the corresponding dongle is restored (the value is incremented by 1).

Thus, correction of the license counter is essential for Guardant Net protection. If the client executes Login operation when the network counter of the dongle has already been exhausted (shows 0), the server will return a corresponding error message, and the application copy will not be started. This is how licensing of network software is implemented in Guardant Net protection.

Important

Network license counters are corrected in the server's memory, but not the memory of Guardant Net dongles. This ensures the safety of the network license counter during hardware failures in the network, workstation 'hang-ups', etc.

How are Licenses Allocated?

The important result of logging in Guardant Net server is the capture of one of the licenses.

Theoretically, the licenses of the network dongle can be allocated between two objects: protected applications and workstations on which these applications run. On the face of it, allocation to applications seems more reasonable, since it is their running that causes the license counter to be decremented. However this way of allocating licenses has the following serious disadvantages. If network licenses were allocated to applications, then: in case an application hung up its network license would remain captured (as a matter of fact, the license would be lost) until Guardant Net server is reloaded; running of several copies of the same application on the same workstation would lead to overuse of licenses.

By the way, this is quite a common situation. This may happen if a user (either accidentally or due to lack of experience) runs several application copies on his computer.

That is why in Guardant Net protection the network licenses are allocated to workstations but not to protected applications. It means that:

- The network license is captured (license counter decremented by 1) only when the first copy of protected application is run. If new copies of the same application (or other applications bound to the same dongle) are run on this workstation, the network license counter will not decrement.
- The network license counter of the dongle recovers (i.e., is incremented by 1) only after the last copy of the protected application that has been started on this workstation terminates. In this case, it does not matter in which order the copies have been started.
- If the running protected application has hanged up, the license will remain captured by this workstation. You will still be able to run the application on this workstation, however this will not result in the license counter changing.
- If after capturing the license no application from this workstation communicates with Guardant Net server for 24 hours, then the license will be returned to the dongle after the timeout.

Important

In Guardant 4.1 (released 07.06.02) and later versions the license control mechanism has been improved.

In order to release the hung-up licenses the Guardant Net is regularly polled, at least every five minutes, by the Client (Win16, Win32). Communication with the application, which has not sent its data during the three intervals between the polls, terminates, if a new client lacks a license.

Guardant Net Features

- Guardant Net protection is very easy-to-use. All the end-users will have to do is install the protected product, set up configuration files of the dongle's client and server (GNCLIENT.INI and NNKSRV32.INI) and run Guardant Net application server; as soon as they are done with this they can proceed with their work straight away.
- Even if your customers have several different NetBIOS interfaces loaded on their workstations, this will not affect either the user of the protected application or the protection itself. The client will 'agree' with Guardant Net server which NetBIOS interface should be used for communication. All this will be done automatically and require no additional settings of Guardant Net software.
- Guardant Net software provides for highly reliable network communication. If several NetBIOS interfaces are loaded on the workstation and one of them fails, Guardant Net will simply switch to another available NetBIOS interface (and in this case there is no need in reloading Guardant Net server).
- Guardant Net protection ensures reliable functioning on network bridges (i.e., computers with several network adapters). Guardant Net software will be automatically adjusted to the environments.
- Supports complex networks, composed of several segments.
- Guardant Net dongles support all protection capabilities provided by Guardant Stealth plus special capabilities such as protection and licensing of network software.

Supported Networks and Protocols

Guardant Net supports TCP/IP and NetBIOS network protocols (or their emulators). TCP/IP protocol can be used by Win16 and Win32 applications only, while NetBIOS can be used by DOS, Win16 and Win32 applications.

At least one of these protocols should be configured in LAN, otherwise the dongle's server will return an error: 'Protocol not found'. In other words, there may be cases when the client (i.e. the protected application) and the Guardant Net server do not 'see' each other, because neither NetBIOS nor TCP/IP protocol, which are present on the client computer, has been loaded on the computer with the dongle's server.

Guardant Net network dongles can work in any local networks with NetBIOS and TCP/IP interfaces. However, one should note that Guardant Net server is a Win32 application, so it must be loaded on the server or workstation that is running under Windows.

Supporting Several Adapters and Network Interfaces

Latest operating systems are capable of supporting several protocols simultaneously, for each of the network adapters installed on the computer. Each 'network interface – adapter' pair is supplied with a unique number and is called LANA (LAN Adapter). All Guardant Net network software products are designed with the capability to concurrently work with several network interfaces and network adapters.

Specifics of Using the NetBIOS Protocol

Working in Complex Networks

By default, the Client can 'see' Guardant Net Server only if it is located within the same network segment. However, sometimes it may be needed to run the protected application in a complex network composed of several segments. Here are some ideas:

1. If network configuration permits, Guardant Net Server can be loaded on a computer that can be accessed from several network segments at a time (for example, on Windows NT Server connected to several network segments simultaneously).
2. Permission of packet exchange between the segments. This requires installation or reconfiguration of the router and/or the switch. This can be done, for example, for one of the NetBIOS interfaces available.
3. Installation of one Guardant Net Server in each segment.
4. Use of Guardant Stealth (i.e., local dongles) on some of the workstations that belong to other segments.

Compatibility with Network Interfaces in Various Operating Systems

The most widespread versions of NetBIOS interfaces are the following:

Microsoft NetBEUI. Non-routed protocol offered by Microsoft for smaller networks. In Windows 95 it is installed by default.

Microsoft NetBIOS over TCP/IP. In Windows 95/98/Me it is installed by default during the installation of TCP/IP. Routing is possible.

Microsoft NetBIOS emulator over IPX. It is most often used in mixed networks to link applications, which run under MS Windows and NOVELL NetWare. Routing is possible.

NOVELL NetBIOS emulator over IPX. It is most often used in mixed networks to link applications, which run under NOVELL NetWare and MS Windows. Routing is possible.

IBM NetBIOS in OS/2. Basic protocol for OS/2.

LANtastic NetBIOS. Basic protocol for ARTISOFT LANtastic.

By now the following have been tested:

	NetBEUI	NB on IPX	NB on TCP/IP	NetBIOS
MS LAN Manager	+	N/A	N/A	N/A
MS Windows 3.11	+	+	+	N/A
MS Windows 95	+	+	+	N/A
MS Windows 95 OSR2	+	+	+	N/A
MS Windows 98	+	+	+	N/A
MS Windows 2000	+	+	+	N/A
MS Windows XP	+	+	+	N/A
MS Windows NT 3.5x	+	+	+	N/A
MS Windows NT 4.00	+	+	+	N/A
IBM OS/2 4.00	N/A	N/A	N/A	?
NOVELL NetWare 3.1x	N/A	+	N/A	N/A
NOVELL NetWare 4.1x	N/A	+	N/A	N/A
ARTISOFT LANtastic	N/A	N/A	N/A	?

+– protocol has been tested.

? –protocol has not been tested yet.

N/A – NetBIOS interface is not available in this particular operating system.

It should be kept in mind that NetBIOS interfaces are not generally compatible with each other. It means, for example, that packets exchange between NetBEUI and IPX-based NetBIOS emulator is impossible. The only exceptions are IPX-based NetBIOS emulators from Microsoft and NOVELL, which are highly compatible.

Network Throughput

As it has been mentioned above, operating systems, such as Windows 95/98/Me/NT/2000/XP are capable of supporting several protocols simultaneously for each of the network adapters installed on the computer. This is a very useful capability, yet an inefficient configuration can considerably reduce the network throughput.

In our case, to minimize Guardant Net Server response time during Login operation (logging the protected application in Guardant Net Server) it is a good idea to use one of the NetBIOS interfaces already installed on the computer running Guardant Net Server, as a default protocol for each of the Client workstations. Otherwise, the Client will attempt to establish a link with the Server by trying all available NetBIOS interfaces one after another until it finds the required one.

Configuring the Guardant Net Server and Client

General Information

To run the protected application in local network it is enough to install one Guardant Net dongle on any workstation or server. Operations with Guardant Net dongle via the network are supported by the Client (Guardant Net API and/or automatic protection 'vaccine') and the Server (Guardant Net server) components of Guardant Net software.

To link the Client with the Server of Guardant Net software you must set up the Client's (GNCLIENT.INI) and the Server's (NNKSRV32.INI) configuration files; depending on the current protocols you must specify the Server's NetBIOS-name, its IP (or host name), set timeouts for sending and receiving of data, etc.

Guardant Net Client software does not require Guardant drivers to be installed since it does not communicate directly with the dongle. Instructions for installation of Guardant drivers required for Guardant Net Server functioning are the same as for the local usage of Guardant Stealth (see 'Guardant Drivers').

Configuring Guardant Net Server

Configurable parameters of Guardant Net servers are accumulated in NNKSRV32.INI file, which must be located in the same directory as the corresponding server. If this file is not found all parameters of the server will be assigned default values.

Configurable parameters of the server are grouped into the [NCBs], [CACHE], [TIMEOUT], [SYSTEM], [SERVICE], [PROTOCOLS] and [SERVER] sections.

[NCBs] Section

[NCBs] section accumulates parameters, which allow you to configure the server to work with greater or smaller number of clients.

By default, Guardant Net server has configuration that is enough to serve about 10 clients at a time, no additional configuration is required for this. Intervention may be needed when the peak value of NCB parameter displayed in the status window of Guardant Net server is getting close to its maximum, or when corresponding messages are displayed by the server.

TotalNCB=xx

This parameter specifies the maximum number of NCBs that Guardant Net server can create when working with clients (or, in other words, it is the maximum number of network packets which the server can receive/transmit). A server can 'spend' up to 2 NCBs per client at a time, therefore TotalNCB value indirectly reflects the maximum number of clients the server can theoretically poll simultaneously. Valid values of the parameter range between 1 and 256, the default value is 50.

NCBInLANA=xx

When using NetBIOS protocol LANA parameter is very important. The number of LANAs on a workstation depends on the number of NetBIOS interfaces installed on this workstation, as well as the number of installed network adapters (in fact, the number of LANAs on a workstation is to be equal to the result of multiplying these two values). After being loaded, the server waits for queries from new clients via all available LANAs, yet actual communication with each client is carried out via one LANA only.

NCBInLANA parameter specifies the number of NCBs that are allocated by the server for waiting a query from a new client on each LANA. To put it otherwise this parameter shows how many new clients can theoretically be logged in by the server on each LANA at the same time. Allowed values of the parameter range between 1 and 9, the default is 3.

For proper functioning of the server, the value of TotalNCB parameter should exceed by two the value of NCBInLANA parameter multiplied by the number of LANAs used on a particular workstation. Thus, the following condition is to be met:

TotalNCB > NCBInLANA*LANAs + 2

[CACHE] Section

[CACHE] section accumulates parameters, which specify the configuration of Guardant Net server's cache. The cache is used to reduce the response time of Guardant Net server during execution of the most frequent operation, i.e., reading from the dongle's memory. The cache is most effective for a Guardant Net server interfacing with a big number of clients, and it dramatically increases the stability of the server during peak overloads.

CacheMode=On|Off

This parameter allows you either to enable (On) or disable (Off) the cache of Guardant Net server. If the cache is off, other parameters of [CACHE] section are ignored. By default, the cache is enabled (On).

CacheCnt=xx

This parameter specifies the maximum number of reads during which the information can be taken from the server's cache but not from the dongle. Allowed values of the parameter range between 1 and 16, the default value is 10. As soon as the counter reaches the specified value, the next read operation is done directly from the dongle, while the cache contents will be updated. The cache contents are also updated during each write into the dongle's memory.

CacheTime=xx

This parameter specifies the interval in seconds during which the read operations will be executed from Guardant Net server's cache, when possible. Allowed values of the parameter range between 1 and 60, the default value is 30. Upon the expiry of this interval, the next reading will be done from the dongle, while the cache contents will be updated.

Thus actual reading from the dongle's memory will be executed in case either of the conditions specified by the above two parameters are met: either the number of reads from the cache reaches its maximum value or the specified time interval expires. This scheme has been implemented in order to prevent any attempts to activate the server without the dongle.

[TIMEOUT] Section

[TIMEOUT] section contains parameters that set the duration of timeout for dongls locking, as well as timeouts for sending and receiving of data (in seconds):

LockTimeout=xx

You can lock and unlock a dongle using LockBeg and LockEnd operations. If, for some reason, the dongle remains locked for a long time, it will be automatically unlocked after the timeout period expires. Timeout values can be specified in the range between 1 and 600, the default is 60.

TO_SEND=xx

Timeout for sending of data by the client to the dongle server. Timeout duration can range between 1 and 120 seconds, the default duration is 10 seconds.

TO_RECEIVE=xx

Timeout for receiving of client's data by the dongle's server. Timeout duration can range between 1 and 120 seconds, the default duration is 10 seconds.

If the line is slow or the server is overloaded, it is recommended that you set higher values for TO_SEND and TO_RECEIVE parameters, in order to prevent the cut off of the client upon the expiry of the timeout.

[SYSTEM] Section

[SYSTEM] section contains parameters, which specify the behavior of the server as Windows application.

StartMinimized=On|Off

Enabling of this parameter (On) allows Guardant Net server to be loaded with the main window minimized. By default, this parameter is disabled (Off).

MoveToTSA=On|Off

Enabling of this parameter (On) allows the server to place its icon to TSA (Taskbar Status Area) during loading. When the window is minimized, the server removes its icon from the main Taskbar. You can invoke the main window of the server by double-clicking on its icon in TSA. This parameter can be used in 32-bit server only. By default, this parameter is enabled (On).

QuietExit=On|Off

This parameter allows you to activate (On) or deactivate (Off) the mode of shutting down the server without confirmation. If the parameter is activated (On) and none of network licenses of any dongle appears captured at the moment of exit, the server terminates its running without confirmation. Otherwise, the server displays a corresponding warning message, and termination of the application should be confirmed by the user.

[SERVICE] Section

[SERVICE] section accumulates parameters that specify the features of running 32-bit Guardant Net server as Windows NT/2000/XP Service.

ServiceMode=On|Off

Enabling (On) of this parameter gives an opportunity to run Guardant Net server as Windows NT/2000/XP Service. In case the parameter is disabled, the remaining parameters of this section are ignored. The default value is 'Off'.

ServiceInstTimeout=xx

When Guardant Net server is loaded, it polls all network dongles attached to the computer. Since the Service is started during the loading of the OS, this process may coincide in time with Guardant drivers initialisation process. If, during starting of the Service Guardant drivers are not yet loaded, then the dongles will be unavailable, and the Service will fail to start.

'ServiceInstTimeout' parameter specifies the time in seconds during which Guardant Net Service will be waiting for Guardant drivers to be loaded. Timeout values range between 1 and 600 seconds, the default value is 100.

[PROTOCOLS] Section

The [Protocols] Section contains parameters that define current network protocols and their priority

TCP_IP=x

NETBIOS=x

Possible values for the parameter:

0-protocol is not used

1-protocol is used as the primary protocol

2-protocol is used as the secondary protocol

DOS applications cannot use TCP/IP protocol. All they need from the NNKSRV32.INI file is only the information about NetBIOS: server name (NB_NAME) and timeouts (TO_SEND, TO_RECEIVE).

If the .INI file is not found, all parameters of the server will be assigned default values.

[SERVER] Section

The Server Section contains parameters that are used to specify NETBIOS name of the server and the TCP/IP address of the port. When license management system is being used, the dongle description counter is also stored in this section.

NB_NAME=NVSK_SRVR

NVSK_SRVR is a default server name

TCP_PORT=3182

3182 is a default number of TCP/IP port.

Dongles=x

x is the number of dongle descriptions

[KEY_xx] Section

When license management system is enabled, the sections of [KEY_xx] type, where xx is a section number, are added to the server configuration file by means of NSKUTIL program (or manually in any text editor). These sections contain descriptions (such as license table data and dongle search parameters) of dongles that can be used for any applications.

Dongle search parameters are arranged in the window according to their priority, in the descending order. Thus, the dongle ID has the highest priority while the bit mask, the lowest priority. The value of the higher-priority parameter is higher than the aggregate value of all lower-priority parameters.

When the Guardant Net server is launched, it reads information from the attached dongles and selects out of descriptions contained in the INI-file the one which fits each dongle most of all. The most fitting description is the one in which the aggregate priority of the search parameters, which fit a particular dongle, is higher that the aggregate priority of all other descriptions. If there are several descriptions sections in the INI-file with the same aggregate priorities, then the first description section will be used.

Public Code=xx

Public code of a dongle.

ID=xx

ID number of a dongle. This parameter is assigned the highest priority. If the ID of a dongle is specified, the description will be allocated to this particular dongle only.

VendorName=xx

Name of the vendor. Data from the license table.

ProgramName=xx

Name of the protected software package. Data from the license table.

ProgramNumber=xx

Program number. An additional parameter for the search of a fitting description for the dongle.

Version=xx

Version. An additional parameter for the search of a fitting description for the dongle.

Mask=xx

A bit mask. An additional parameter for the search of a fitting description for the dongle.

SerialNumber=xx

A serial number. An additional parameter for the search of a fitting description for the dongle.

Module0=xx

Name of the first module of the software package. Data from the license table.

ModuleN=xx

Name of the n-module of the software package. Data from the license table.

Example:

You are required to specify text descriptions for the vendor's multi-module applications. The vendor releases Version 1 of the application (OLD_ PROGRAM) under its original name (OLD_NAME). While the two new applications Version 2 are released under the new name (NEW_NAME) such as NEW_PROGRAM A and NEW_PROGRAM B.

Below is a fragment of the configuration file, beginning from the [SERVER] section.

```
[SERVER]
; Default NETBIOS-name of the dongle server
NB_NAME=NVSK_SRVR
; Default TCP/IP port
TCP_PORT=3182
; A number of sections with descriptions of dongles
Dongles=3

[KEY_00]
; Public code of the vendor
PublicCode=PBLCODE
; Previous name of the vendor
VendorName=OLD_NAME
; Name of the 'old' program
ProgramName=OLD_PROGRAM
; Program version
Version=1
; Name of modules of the 'old' program
Module0=OLD_MODULE 1
Module1=OLD_MODULE 2
ModuleN=OLD_MODULE N

[KEY_01]
; Public code of the vendor
PublicCode=PBLCODE
; Program number of the NEW_PROGRAM A program
ProgramNumber=0
; Program version
Version=2
; New name of the vendor
VendorName=NEW_NAME
; Program name
ProgramName=NEW_PROGRAM A
; Program modules' name
```

```
Module0=MODULE_A 1
Module1=MODULE_A 2
ModuleN=MODULE_A N

[KEY_02]
; Public code of the vendor
PublicCode=PBLCODE
; Program number of the NEW_PROGRAM B program
ProgramNumber=1
; Program version
Version=2
; New name of the vendor
VendorName=NEW_NAME
; Program name
ProgramName=NEW_PROGRAM B
; Program modules' name
Module0=MODULE_B 1
Module1=MODULE_B 2
ModuleN=MODULE_B N
```

Configuring Guardant Net Client

Configurable parameters of Guardant Net client are accumulated in [PROTOCOLS], [TIMEOUT] and [SERVER] sections of GNCLIENT.INI file. This file should be located in the same directory as the copy of the protected application. Configuration file for the JAVA network client can also be stored in the Windows root directory (for example C:\WINDOWS).

If the GNCLIENT.INI file is not available, all parameters of the Guardant Net server are assigned default values. In this case, the client will search for a server with the default name (NVSK_SRVR) and only via NETBIOS protocol.

[PROTOCOLS] Section

The [Protocols] Section contains parameters that define current network protocols and their priority

TCP_IP=x

NETBIOS=x

Possible values of the parameter:

0-protocol is not used

1-protocol is used as the primary protocol

2-protocol is used as the secondary protocol

DOS applications cannot use TCP/IP protocol. All they need from the NNKSRV32.INI file is only the information about NetBIOS: server name (NB_NAME) and timeouts (TO_SEND, TO_RECEIVE).

[TIMEOUT] Section

[TIMEOUT] section contains parameters that set the duration for the sending and receiving of data (in seconds):

TO_SEND=xx

Timeout for the sending of data by the client to the dongle server. Time-out duration can range between 1 and 120 seconds, the default duration is 10 seconds.

TO_RECEIVE=xx

Timeout for the receiving of client's data by the dongle's server. Timeout duration can range between 1 and 120 seconds, the default duration is 10 seconds.

If the line is slow or the server is overloaded, you should set higher values for TO_SEND and TO_RECEIVE parameters, in order to prevent the cut off of the client upon the expiry of the timeout.

[SERVER] Section

The Server Section contains parameters which are used to specify NETBIOS name of the server and TCP/IP address of the port.

TCP_PORT=3182

3182-default address of the TCP/IP port.

IP_NAME=127.0.0.1

If the network uses dynamic IP addresses (DHCP server), you should specify the host name of the computer in which the dongle's server is installed, instead of the IP address. 127.0.0.1 is the default IP address of the dongle server.

NB_NAME =NVSK_SRVR

NVSK_SRVR is the default name of the dongle server.

Guardant Net Server

Guardant software includes a 32-bit Guardant Net server (NNKSRV32.EXE utility). Guardant Net server enables communication between the protected network application and Guardant Net dongle in LANs where TCP/IP and NetBIOS protocol are supported. One server is capable of servicing queries addressed to several Guardant Net dongles.

Loading the Server

Guardant Net server should be loaded on the same computer to which the dongle is attached. Within the LAN several Guardant Net servers can be run. They must be run on different computers and have unique names. You cannot run two servers (server and service, two services) on the same workstation.

Important

For running of 32-bit Guardant Net server (NNKSRV32.EXE) the presence of external vaccine file NOVEX32.DLL is required.

Guardant Net server can run not only as an ordinary window application, but also as Windows NT/2000/XP service.

After loading is completed, the main window of Guardant Net server will be displayed.

Monitor Function

Guardant Net Server combines the functions of both a server and a monitor. The server window is split into two parts.

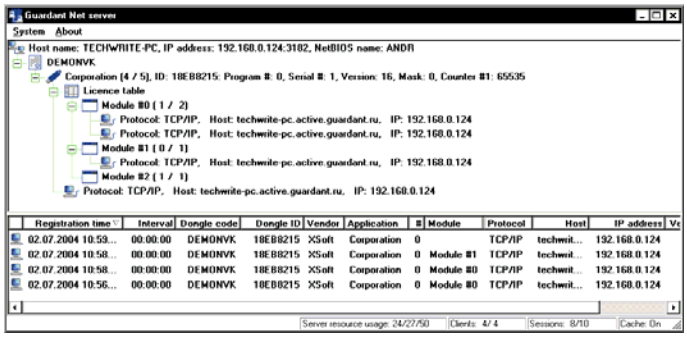

Figure 2. Guardant Net Server main window.

The upper part of the window displays, in a tree-like structure, information about the computer on which a dongle/dongles and a server are installed, as well as basic data about clients.

At the bottom of the tree the details about the computer on which the Guardant Net server is run are displayed, such as the computer name (host name), IP-address, NetBIOS-name.

The first nesting level contains the Public code of the dongle.

The second nesting level displays information about the dongle, particularly the name of the application protected by the dongle, the license counter (current/maximum), dongle's ID, application number, serial number, program version, value of counter 1.

The third nesting level displays basic information about the client, such as network protocol, name and IP-address of the computer on which the client is run. If the license management system is enabled, the license table icon is displayed on this level as well.

The level below the license table contains information about protected application modules, particularly the module name and the number (current/maximum) of licenses in the module.

The next level contains basic information about the clients that use resources of any application module: network protocol, name and IP-address of the computer on which the client runs.

Important

The status of a dongle registered on the server is indicated by means of special marks next to the dongle icon:

- 'lock' means that the dongle is locked by the LockBeg operation (one of the application copies executes several read/write operations one after another).
- 'x' means that the dongle is not available (disconnected).

Absence of marks means that the dongle is available (i.e. physically connected to the computer port) and is not locked.

At the bottom of the server window you can find a table with a detailed information about the clients:

- Login time
- Time elapsed since the last communication with the client
- Public code and ID of the dongle, which serves the client
- Program vendor
- Program name and number
- Module name and number (if the license table is used)
- Network protocol used for the connection
- Name and IP-address of the computer on which the client is run
- Platform for which the application has been designed

Data about clients can be sorted by any parameter, in ascending or descending order.

If a query is not received from the client within 15 minutes (i.e. the client hangs up), the client is highlighted in grey but is not removed from the clients list. Connection with the hung-up client terminates only when a new user needs this particular license.

The server status line displays statistics about its functioning:

- Utilization of server resources – current, peak and maximum number of NCBs used by the server during the sessions of communication with clients.
- Clients – current and peak number of clients currently served by the server.
- Sessions – current and peak number of sessions opened by the clients (communication sessions).
- Cache – the state of server cache.

Important
'Peak value' means the maximum value of the parameter actually achieved at some point of time.

Registration of a New Dongle

A new dongle can be added to the dongles already registered on the server. However, mere attachment of the dongle to the computer is not enough; the dongle needs also to be registered. To register a new dongle, **System | Refresh** menu command can be used. The dongle is deemed registered successfully if the information about this dongle has appeared in the list of dongles of this server. From this moment onwards programs bound to this dongle can be run.

If you disconnect the registered dongle from the workstation and then re-read the dongles using **System | Refresh** menu command, the dongle will be marked with an '**x**' symbol, which means that the dongle is registered but is physically not attached (i.e., the dongle is unavailable). From this moment onwards running of any programs bound to this dongle will become impossible.

To restore the dongle registration follow the above-described steps. If registration is completed successfully, the 'x' symbol will disappear.

Important
You cannot cancel the registration of the dongle, which has already been registered.

License Management System

Guardant software versions 4.5 and up allow Guardant Net server to manage licenses in multi-module software packages, separately in each module.

A two-level license control scheme is used in the Guardant Net server:

1. The total number of workstations, on which the protected applications are run, is limited to the actual license limit available in a dongle (value of counter #2)

2. The number of workstations on which a certain module of the program is used, is limited to a resource of this module (value of the appropriate byte in the license table)

An actual license limit should not necessarily be equal to the total number of licenses in all modules.

Example:

The protected program package consists of four modules: Accounting, Wages, Personnel, Office. The actual license limit in a dongle is 15. The number of licenses in each module is indicated in the table:

Module	Number of licenses
Accounting	10
Wages	10
Personnel	7
Office	5

Thus, different modules can be run on 15 workstations simultaneously, but the number of computers, on which any of these module runs, cannot exceeded the license limit of this module (i.e. not more than 10 Accounting licenses, 7 Personnel licenses, 5 Office licenses, etc.)

If several modules, for example Accounting, Wages and Personnel modules, are run on the same computer, the actual license counter in the dongle is decremented by 1; likewise the number of license counters in each of these modules will be decremented by 1 too.

To be able to utilize the license management system you should do the following:

- create the 'License table' field in the dongle's memory and define the number of modules, license limits, as well as additional parameters in this table
- enable /MN=xx option during the automatic protection, where xx is the number of the module in the license table (use 'Enable the license management system' parameter of the autoprotection wizard)
- when working with an API-based protection you should use nnkLoginLMS function, instead of nnkLogin, to register the application on the server.

Important

When you are updating the entire memory of a dongle using the Guardant API, you should apply nnkProtectLMS function, instead of nnkProtect.

License table format

The address of the license table is identified in the dongle memory by the value indicated in kmTableLMS field (29 SAM).

The size of the table heading is two bytes.

The first byte contains the information on the size of a cell (1 byte if the high bit is 0, and 2 bytes if the high bit is 1) and on the number of modules in the license table.

The maximum license limit in a module depends on the size of a cell. If the size of a cell is 1 byte, the number of licenses in each module can be limited to a maximum of 254; if the size of a cell is 2 bytes, the number of licenses in a module can be limited to a maximum of 65534.

Example:

00000011 – a high bit of the first byte of the table is set to 0, so the size of each cell is one byte and the number of modules in the table is three.

10000010 – a high bit of the first byte of the table is set to 1, so the size of a cell is two bytes and the number of modules in the table is two.

The second byte of the table is reserved.

The cells of the table go below the heading.

1) General structure of the license table with single-byte cells

Offset	Description
0	Number of modules in the table
1	Reserved
2	Number of licenses for the 1st module
3	Number of licenses for the 2nd module
4	Number of licenses for the 3rd module
5	Number of licenses for the 4th module
...	.....
	An extra byte for word-alignment (ONLY if the number of modules is odd)

2) General structure of the license table with the two-byte cells

Offset	Description
0	Number of modules in the table
1	Reserved
2	Number of licenses for the 1st module (low byte)
3	Number of licenses for the 1st module (high byte)
4	Number of licenses for the 2nd module (low byte)
5	Number of licenses for the 2nd module (high byte)
6	Number of licenses for the 3rd module (low byte)
7	Number of licenses for the 3rd module (high byte)
8	Number of licenses for the 4th module (low byte)
9	Number of licenses for the 4th module (high byte)
...	.....

Running the Server As Windows NT/2000/XP Service

Guardant Net server can run not only as an ordinary window application, but also as Windows NT/2000/XP Service.

The advantages of this Service is that it is started by the operating system when the latter is loaded, and to start the Service no logging in computer is required, while the user gets access to Windows special Service control facilities.

Starting Guardant Net Service

To permit the running of Guardant Net server as a Service you should specify 'On' value in 'ServiceMode' parameter of NNKSRV32.INI file (for more details on configuring the server see below). After that you will be able to run the server both as an ordinary application and a Service. Otherwise, NNKSRV32.EXE utility can be run only as an ordinary window application.

To launch a Service you should run Guardant Net server with /I option:
nnksrv32.exe /I

This operation needs to be executed only once. As soon as Guardant Net server is successfully launched, the protected applications will get access to Guardant Net dongles. The Service will be launched automatically each time when Windows NT/2000/XP is started.

Important
Guardant Net Service can run only in Windows NT/2000/XP.

Working with Guardant Net Service

- Guardant Net Service cannot be launched if 'ServiceMode=On' parameter is not specified in NNKSRV32.INI file.
- Guardant Net Service has no interface window. You should control network licenses allocation and the status of Guardant Net dongles with the help of Guardant Net network monitors.
- You can temporarily suspend Guardant Net Service. To do this, you should select **Control Panel|Services** (for Windows NT) or **Control Panel|Administrative Tools|Services** (for Windows 2000/XP) and right-click on 'Guardant Net Service' item. In the popped out menu you should choose 'Stop' item. The Service will remain installed in the system, but will no longer process queries sent to Guardant Net dongles. To resume the running you should start the Service from 'Control Panel' or use NNKSRV32.EXE /I command.

- Specifying of 'Off' value in 'ServiceMode' parameter does not lead to automatic removal of Guardant Net Service from the system. Meanwhile it will become impossible to start the Service during loading of Windows. To prevent this, you should change 'ServiceMode' value and start the Service from the 'Control Panel' or by NNKSRV32.EXE /I command, or delete it from the system as described below.
- If no Guardant Net dongle is attached to the computer, you will not be able to launch the Service when loading Windows. To resume its running you should attach the dongle and start the Service from 'Control Panel' (see above) or by NNKSRV32.EXE /I command.

Removing Guardant Net Service From the System

To disable Guardant Net Service you should run NNKSRV32.EXE with /R option:

```
nnksrv32.exe /R
```

Guardant Net Network Monitors

The purpose of network monitors is to receive information about Guardant Net dongles via network. They can be started from any workstation, and the same rules apply to them as to Guardant Net clients.

There are three network monitors, which are identical in functions:

NNKMON.EXE	DOS application;
NNKMONW.EXE	16-bit Windows application;
NNKMON32.EXE	32-bit Windows application.

When any of these utilities is started, it will connect to Guardant Net server and display the information received from the server about current parameters and the status of all Guardant Net dongles registered on the server (this information is identical to the information displayed by the server in its window). This will allow the network administrator to receive information about the status of Guardant Net server and about the allocation of license resources to Guardant Net dongles at any time and from any workstation.

How to Increase Reliability of Network Communication

Improving Protection Strength against Attacks

End-users may try to run more copies of the protected application in the network than permitted. After running a maximally permitted number of copies, they may force reloading of Guardant Net server and get an opportunity to run just as many application copies again.

It is quite easy to protect against this. Both Guardant Net server and Guardant Net client have reliable built-in facilities to obstruct such attempts on the part of the user. Your task is to activate these facilities. The protected network application should periodically verify the dongle's presence. Use timer-based dongle verification option during the automatic protection of the application and/or periodically poll the dongle with the API functions from different locations of the application.

The thing is that after being reloaded Guardant Net server will not process queries of those applications started before reloading. Thus, the first polling of the dongle carried out by an 'old' application copy after the server has been reloaded will return the following error: nse_ServerReloaded. In some time, all copies of the application started before the reloading of the server will stop running.

Increasing Performance of Guardant Net Server

1. It is not recommended that you call the dongle too often. The point is that the minimum response time of Guardant Net is about 150 - 200 milliseconds. Thus, the server can exchange with Guardant Net no more than 5 to 6 times per second. Moreover, during the execution of Transform operation the time of exchange may even increase several times (Transform operation is quite slow). Therefore, for example, five applications, each calling the dongle once in a second, can easily overload Guardant Net server, because in this case little will depend on its speed. The server will start 'freezing' for a long time and lose network packets. Therefore, you should remember that the optimal interval between the polls should be random and range between 5 and 30 minutes. It is not recommended that you carry out many tests at one time, because in this case the possibility of peak overloads increases dramatically. If you follow these recommendations, the server will be able to poll up to 100 protected applications, which are running simultaneously. This number seems big enough, however it should be kept in mind that there has to be only one server in the network and that several dongles can be registered on it (each with its own network license limit). Network administrators should be warned against the risk of overloading the server.
2. Usage of cache enables to considerably increase Guardant Net server's speed. Accordingly, the server will be capable of working with more clients (protected applications) at a time. However, if the above recommendations are not followed, even cache will not help to prevent overloading of the server.

3. It is not recommended that you enable the automatic start function for the protected network application, because in this case the risk of overloading is also very strong. For example, imagine how a new day begins in a big bank, where hundred of terminals are turned on at once, and all of them start to send their queries to the dongle almost simultaneously.
4. To avoid overloading it is not recommended that complicated checks on the dongle be carried out (especially, where several Transform operations are used one after another) when loading the protected application. A simple check of the dongle's presence would be enough, while more complicated tests should be better postponed until a later stage, making them incidental and timed to certain events. This will make the hacker's life more difficult.
5. Do not assign too high values to configurable parameters in INI file of Guardant Net server. This will not lead to the effect you expect (increase of performance, stable functioning during peak loads, etc.). Instead, the server will start to consume system resources (RAM and CPU usage) excessively. The default values of parameters appear to be optimal for the networks with little and medium number of workstations; there is sense in increasing them only when serious matters arise (for example, when the server has to work in large-scale networks with many dongles) and this should also go along with corresponding changes in the NetBIOS protocol configuration. By the way, if there is a shortage of resources specified by the configurable parameters, the server will inform of this by displaying a corresponding message on the screen.

How to Avoid Problems of Sharing Data in the Network

1. When polling the dongle try not only binding it to its Private codes, but also do carry out a deeper check, involving general-purpose fields (serial number, version, etc.). This will guarantee that the protected application will only use the network licenses of the dongle to which it is bound. It is of importance when several dongles with your Private codes are registered on one server.
2. When hardware algorithms are used which depend on the decrementing value of their executions counter (with the set nsaf_GP and nsaf_GP_dec flags of the hardware algorithm), there is a risk of this protected application copy receiving wrong responses from such algorithm, because only one counter is used in it for all copies of the protected application. Therefore, to avoid possible conflicts do not use algorithms with this set of properties to protect the network applications.

Chapter 8
Guardant API

From this Chapter onwards study of Guardant protection facilities begins as such. This Chapter addresses the principal method of protection – protection with API functions. You will learn which operations are supported by Guardant dongles, and will get comprehensive advice about API functions that these operations are executed with.

Important
Please pay special attention to Transform and EnCode/DeCode operations and, particularly, to the way they are executed and to the difference between them.

Operations

As you already know Guardant API functions allow various actions to be performed with the dongle, including but not limited to writing to the memory, implementing hardware locks, running hardware algorithms, etc. These actions are called operations*.

To execute an operation a corresponding API function has to be called. When calling most of API functions Private access codes are used as mandatory parameters.

Operations with Guardant Stealth, Guardant Fidus and Guardant Net dongles can be divided into 2 main groups: built-in* operations and service* operations.

Built-in Operations

The main distinctive feature of built-in operations is that all of them are executed by the dongle.

Built-in operations are further subdivided into primary and secondary.

Primary built-in operations allow you to execute most important operations with the dongle, particularly:

- Search for the dongle that matches specified search conditions
- Initialize the dongle's memory
- Read from the dongle's memory
- Write data to the dongle's memory
- Implement or release memory read/write locks
- Convert information using the dongle's hardware algorithms.