Global Information Assurance Certification Paper

Keeping in touch with your critical services running on Windows Operating System securely and within SLA's in small and middle sized Financial Service Providers (FSP) based on open source projects

Dalibor Baskovc
GIAC GSEC Certification
July 05th, 2004
Version 0.1 - option 2 (Case Study)

Summary

Abstract

This paper is a survey of the need in IT sector to be in control of business services running securely within Service Level Agreement (SLA) and the benefits one can have managing this process. It is geared to small and middle-sized business using Windows systems as a core of their transaction systems.

I will concentrate only on the possible way to get in touch with critical services running securely through SLA's. I will describe how important it is having control of information assets performance, their state of security and control over the services running within the company. Most important of all, I will show how all this can be done using open source projects.

I will show how it is possible to follow golden IT objectives:
unified monitoring platform for the network, systems, applications and services,
show the importance of proactive versus reactive IT management,
show the importance of putting the business focus in balance with technology focus.

I will demonstrate a plan how to go through important steps of "defense in depth" strategy that I think is the right way of achieving mature security model within the company.

1. Before

In Financial Service Providers (FSP) business drivers expect from IT to offer information to clients in an accurate and safe way, and within timeframes as agreed in SLA's, including service recovery within desired timeframe in case of disaster.

1.1. Establish FSP's baseline security issues

In FSP business information is the most important business asset, so it needs to be suitably protected to ensure business continuity, minimize business damage and maximize return of investment and business opportunities. Different services offer customers information through different channels. Thinking of the future, applications will start communicating with each other or clients will communicate with an application through Internet. Communication language of the future would definitely be XML through which applications will be offering more and more data to each other (for example through Web service technology).

Normally you start to use all the technical security measures you can, starting from active content monitoring/filtering, authentication, authorization, certificate authority and PKI, firewalls, intrusion detection – host based, intrusion detection- network based, intrusion prevention, security appliances, risk assessment, penetration testing, file and session encryption, VPN& cryptographic communication, vulnerability scanners (network/host based), secure WEB servers, WEB application security, real-time security awareness/incident response, enterprise security administration SSL VPN's, managed security services, trusted operating system. But it is strongly advisable, to make a system that will work for you not only now, but also in the future, so you need to make a strategy[1].

1.2. Security officer role

Depending whether your position as security officer is closer to the management or not, your efforts to develop and deploy information security policies through security measures security as a whole in a company can be successful or not. Closer to the management you are, more possibilities you have[2].

Another important role of security officer would be starting to plan business continuity and disaster recovery plans, first to put it on paper and later on to test it and make it part of normal business process. To be successful with this work, you have to develop them both (BCP and DRP)[3] according to business plans and then make them part of corporate policy.

[1] http://infosecuritymag.techtarget.com/2002/ciso/aug/ciso-roundtable.shtml
[2] http://www.csoonline.com/read/060103/crisis.html
[3] http://www.thebci.org/BCAWG.html

In Widows based system one can constantly keep trying to be on the safe side but without a managed process, which cannot be developed within short period of time, you can never step from reactive to proactive side.

Although "proactively reacting" is a step up from simply fighting IT fires, it doesn't provide your users with any guarantees of performance, and it doesn't provide your organization with any IT stability or security. Once you have monitoring systems in place that can inform you of problems before they occur you can start addressing those problems in advance. At this exact point you can move up to the next level of maturity in IT management-moving to service level agreement.

1.3. Possible solution through SLA's

If you do not have a clear view of what business users expect from IT you must conform to their constant urge for ever changing demands, because businesses nowadays are constantly competing with each other to make a better service to customers than competitor.

One of the possible ways would be to develop a plan of all the services IT must offer to business users. Here, we can help ourselves with different standards, like ITIL (de facto standard by the mid-1990s) or BS15000 consisting of eight sections (Scope, Normative Reference, Definitions, General, Service Design and Management, Relationship Processes, Resolution Processes, Control Processes). These sections form the basis for the assessment of a managed IT service. The standard may be used for a variety of purposes, including outsource tenders, ensuring consistency by all service providers, and benchmarking as the basis for formal certification.

2. During

As a Chief Security Officer I have to be confident all the time that necessary security measures are implemented to lower the risk of information being abused or in any way compromised and at the same time they do not affect performance of running services in any way.

2.1. Achieving security goals

In order to reach confidence of business users in security I asked myself what are actually security goals.

The key to achieving just about any goal is the principle of measurement. Critical components of a successful measurement program[4] are:

Quantitative baseline answers to the questions where we are at the moment - the beginning of a measurement program is understanding where organization is today,

Internal comparison answers to the questions where we came from – a single point of reference internally over time,

External comparison answers to the questions where we can go – comparing organizational reference points with known and valued external references.

[4] http://www.csoonline.com/read/060103/fired.html

I decided to concentrate first on the system of getting all the questions answered about where our organization is today in order to be able latter on to move to internal and external comparison answers.

2.1.1. First step (from chaos to stability)

Most problems arise, because network and system administrators in smaller and middle-sized companies do not have time and are even not keen to document everything they have to do in order to run services implemented within the company. It means that nobody actually knows on which level of maturity IT infrastructure is and how it would be possible to get from reactive to proactive state of IT.

2.1.2. Second step (from stability to seek efficiency)

As soon as I figured out that this is the case in my company also, I stopped trying to follow our usual way of working, using our usual technology and proprietary management programs on active LAN and other systems.

Instead, I started to look for a solution trying to reach following goals:

find unified monitoring platform for the network, systems, applications and services,

find the possibility to follow the network and computer systems independent from proprietary systems and independent whether this is windows system or any other communication device,

find the possibility to follow changes from IP scheme up to applications running on different systems,

find the possibility to make a management perspective from the service level that information technology department has to provide,

find the possibility to come from active to proactive role of security,

follow performance, security and services running on different system and communication devices,

find the possibility to have control over web interface,

find a good price/performance ratio.

2.1.3. Third step (from efficiency to good documentation)

Very important process within security is having implemented a good documentation process. What we had in our company was more in our heads than on the paper, which can be a big vulnerability within the company. So I will try to emphasize different areas that goes with security and can clarify different views from different perspective (IT as well as non IT).

All the systems running would have to cover following documentation processes.

Processes involved

1) Analyze information related to security risks[5] and controls,
2) Identify security vulnerabilities,
3) Recommend infrastructure security controls for supporting operating systems, database systems and networks,
4) Recommend application of security controls,

[5] http://www.computerworld.com/securitytopics/security/story/0,10801,81897,00.html

5) Assess security risks of new technologies (Internet, EDI, Electronic Commerce…),
6) Recommend security controls for new technologies.

Inputs

1) Security policies and standards,
2) Application configuration and parameter settings,
3) Application software documentation,
4) Third party documentation,
5) Network diagrams,
6) Remote access security,
7) Systems Diagram,
8) Processes of security administration, monitoring and review,
9) Process of security change management,
10) Profile management processes,
11) Access authorization process,
12) IT Organization Overview Document,
13) IT Organization Chart,
14) IT Environment Overview Document.

Outputs of documentation process

1) Risk assessment,
2) Infrastructure security controls requirements,
3) Application security controls requirements,
4) New technologies control recommendations,
5) Security administration, monitoring, and review control recommendations.

2.2. Practical Steps

2.2.1. Fourth step (from good documentation to "defense in depth" strategy)

In my company we have to consider our new application which would be completely object oriented, running new internet technologies like Web services in Microsoft environment (using Windows 2003 dedicated servers for MS IIS in NLB, COM+ on MS SQL 2000 as transactional database on cluster servers). It is going to be implemented as a new transactional system. But before we do that, it is imperative to make all the necessary preparations within the system, so that all the system will be running in a secure way.

We are doing regular (once or twice a year) penetration tests by a partner company. This means once penetration test is done, we then try to fix all discovered vulnerabilities. But, by the end of fixing them all, new vulnerabilities come into place. So I decided to give more emphasis on staying secure that just getting secure every now and then.

We considered “defense in depth” strategy, which is becoming modern paradigm nowadays. To get to efficient SLA’s I divided security areas in three main areas that our company would have to focus before switching the newly developed system on:

Communication security,

Systems security,

Application security.

Communication security

On the network layer we considered a single location on the perimeter of our network with the possibility to offer different segmentation with different security policy implementation. From the functional point of view it offers us application-level protection firewall functionality, deployed throughout the network and the ability to protect all network resources, which can be compromised, together with functionality of intrusion detection and prevention. We considered dividing network into logical zones not bounded to physical interfaces, with the possibility to deploy policies between security zones and to interfaces within zones. The same architecture is implemented for disaster recovery centre.

Deployed at the perimeter, deep inspection firewall focuses on preventing application-level attacks aimed at Internet-facing applications, such as Web, e-mail, FTP and DNS.[6]

For high availability purpose we implemented two firewalls in an Active/Active mode.

Domains are organized so that for single sign-on we have dummy domain with only ISA server as a member, next domain is with internal users, then domain that includes internal servers and last would be business transaction servers domain.

Behind the firewall we have put ISA 2004 server, which gives us functionality of proxy server as well as https single sign-on point for traffic going to business transaction servers, from ISA server to business servers again https encrypted.

In this way of LAN segmentation inner and outer users would practically use the same services, within different application authorization, expect for some internal company’s resources (file servers, print servers, mail servers, financial application servers, etc.).

From the security point of view this would also mean, that we can practically even threats coming from inner as well as outer world.

[6] http://mediaproducts.gartner.com/gc/webletter/netscreen/issue2/gartner1.html

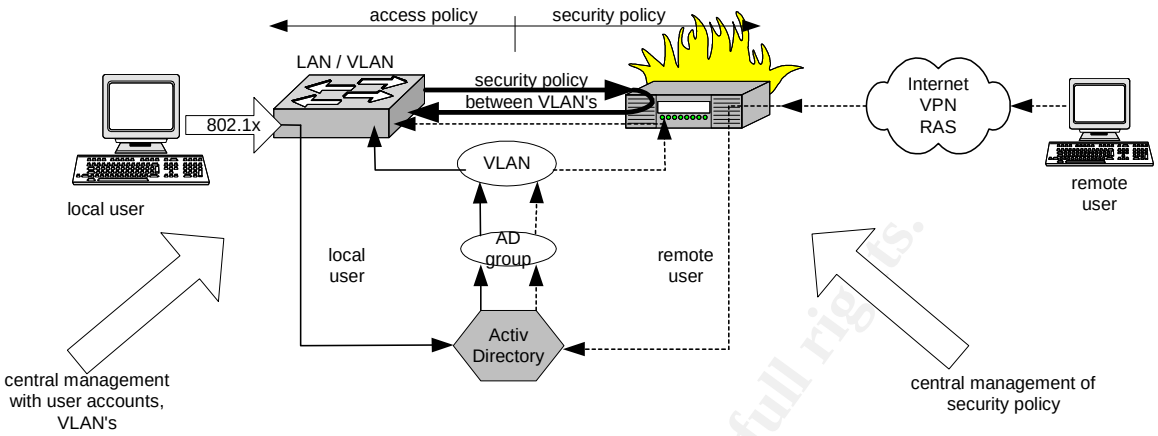


Figure 1: Logical scheme

On the switching level we have implemented the functionality of 802.1x together with AD authentication through level 3 switches. We decided that physical security of our computer centre gives us enough confidence that no unauthorized connection can be made within the centre. For the connection outside, we've implemented 802.1x network switches (layer 3 switches –3Com SuperStack 3 switch 3848).

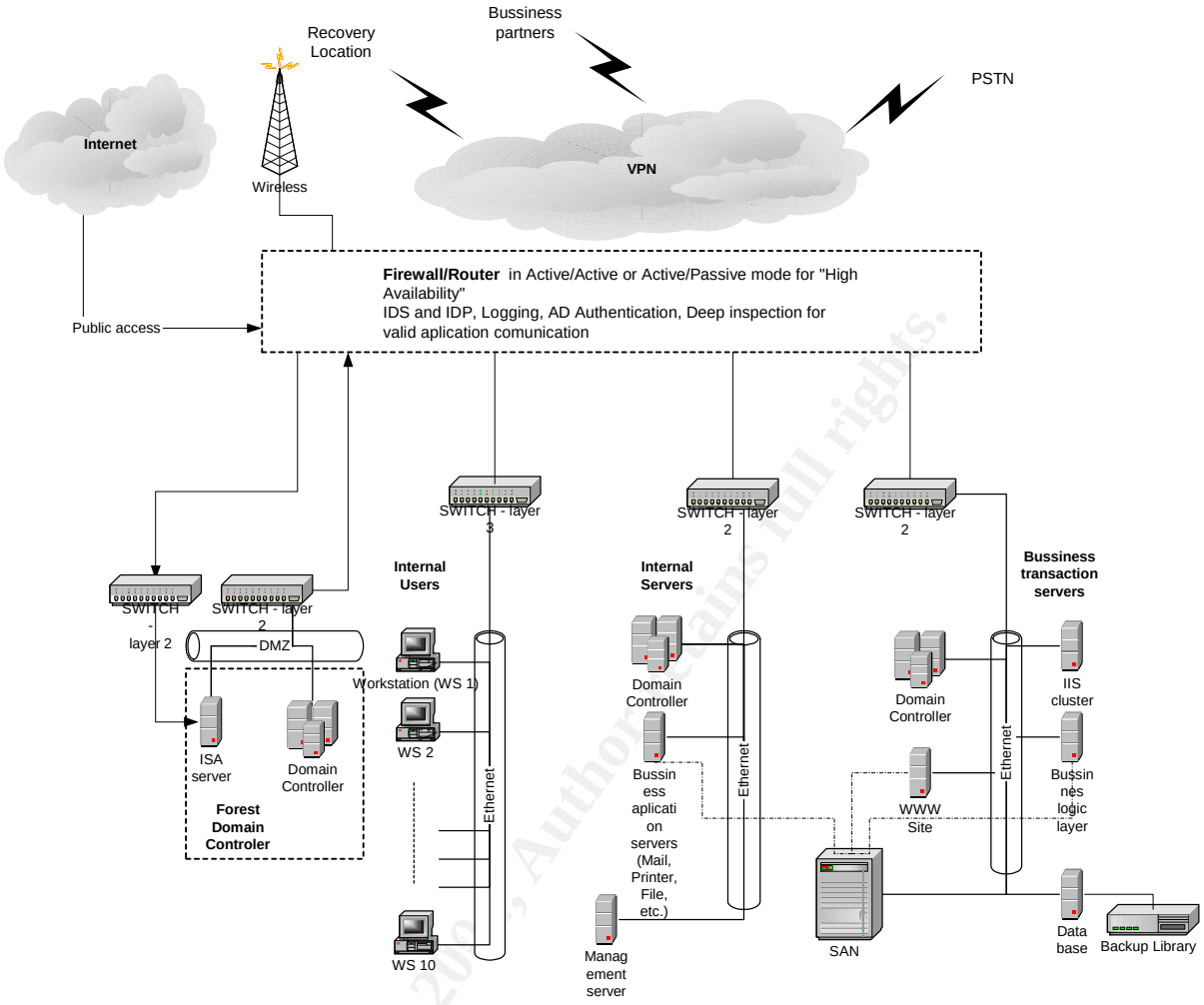


Figure 2: Logical communication scheme

Central security device, giving us the functionality described above was in our case Juniper Netscreen 208 security appliance.

Systems security

For the system part we decided to start a project of upgrading all of our Win2000 and NT 4.0. production servers to Windows 2003, which gives us higher level of security when using it out of the box. This is project that will not be described in this paper.

Application security

Authentication

Users are authenticated by private/public key mechanism and NT challenge/response mechanism. Public key is linked with AD user. Private keys are used to digitally sign transaction, before it is sent over https protocol. Client application is requested to log on to Internet Information Server (IIS) in order to use

business system services, such as XDE server for querying data. Therefore IIS uses NT user authentication for accessing web pages (anonymous access is disabled).

Authorization

Basic AD user properties and certificate revocation lists are used to control if user is allowed to access the system.

Custom security is implemented in database and business objects. Business objects are aware of user and its group to correctly access data. For that purpose we use internal database.

Confidentiality

Confidentiality is achieved by SSL encryption between client application and business system service (between client machine and ISA server). Encryption is decrypted on ISA server, where content scanning is done through ISA plug-ins. After that, SSL session is established between ISA server and IIS, so confidentiality is provided internally also.

Access Control

System uses Windows security for accessing COM+ components and SQL Server. There is no direct data access allowed, instead we use generic system user to access data (also enabling connection polling), which has no interactive sign-on rights. When external application does a request for operation or data, ISA server first authenticates user, then passes the request under special system user impersonation. When request comes to business services, each application uses it's own security modules to filter data or request, based on settings in database as extension of AD security.

Integrity

Information integrity and non-repudiation is assured through digital signatures of HTTP requests.

All data in transactional database must be changed through XML requests.

Digitally signing the XML request with private key also assures non-repudiation, so no one can deny sending request or receiving request. For signing XML requests, *W3C* Recommendation 12 February 2002 XML-Signature Syntax and Processing was implemented.

For the purpose of open PKI infrastructure we are using qualified digital signature.

Why SLA

What does it actually mean? Service level agreement in the simplest way means agreement between you and your users, which service they need within which period of time. What you next have to do is define the asset owners and define on which asset their service is running (which systems, which applications with which data). Next you have to declare what is the maximum allowed downtime and within which period of time asset owner needs service up and running. But before you start planning business contingency you have to do business impact analysis based on risk analysis. Consider doing risk analysis through information assets that business users feel familiar with. If you think well, the most common information asset business users know is services. Often in a FSP companies we can equal IT services with business processes.

For example, the following basic SLA might work within your business:

SLA	Process	Procedure	Owner	System	Application	Data	Availability	Recovery
E-Mail	Managed level of service (processor, memory, etc), Managed availability of service	Monthly reports to management of SLA, Daily backup of system state and each two hours backup of mailboxes	System administrator	Mail server	Exchange 2003	Mailboxes	24 hrs/day	Within 4 hours
Data base	Providing database functional	Transaction backup every two hours,	Database administrator	SQL server	SQL 2000	Company's databases on the server	On line between 7 am. to 6 pm; Offline between 6pm to 7 am next day	Within 15 minutes

2.2.2. Last step (from "defense in depth" strategy to implementation)

To get closer to quantitative baseline answers with a good measurement program I used an open source projects solution to make practical steps gearing to my first goal - making a system that gives me answers about where our organization is today.

When I decided that I would like to cover the performance of the services running in a secure way, I found out following open source projects running. When all put together, they cover all that I need:

Cacti to monitor devices over SNMP information – systems and network devices performance control,

SNORT and ACID network intrusion system – security control,
Nessus and Inprotect to schedule security scans – security control,
Nagios – SLA's control.

SENTINIX

SENTINIX[7] offers all the above solutions as a whole. It is a GNU/Linux distribution designed for monitoring, intrusion detection, vulnerability assessment, statistics/graphing and anti-spam. It's completely free, free to use, free to modify and free to distribute. SENTINIX includes the following software, installed and pre-configured: Nagios, Nagat, Snort, SnortCenter, ACID, Cacti, RRDTool, Nessus, Postfix, MailScanner, SpamAssassin, openMosix, MySQL, Apache, PHP, Perl, Python and lots more. From all of the above I use only what I will show on next pages.
As soon as we have defined services that IT has to offer to business premises, we started to implement a system giving us the information on how these services are running. For that purpose I have chosen an open source solution, because it gave us a good price/performance ratio.

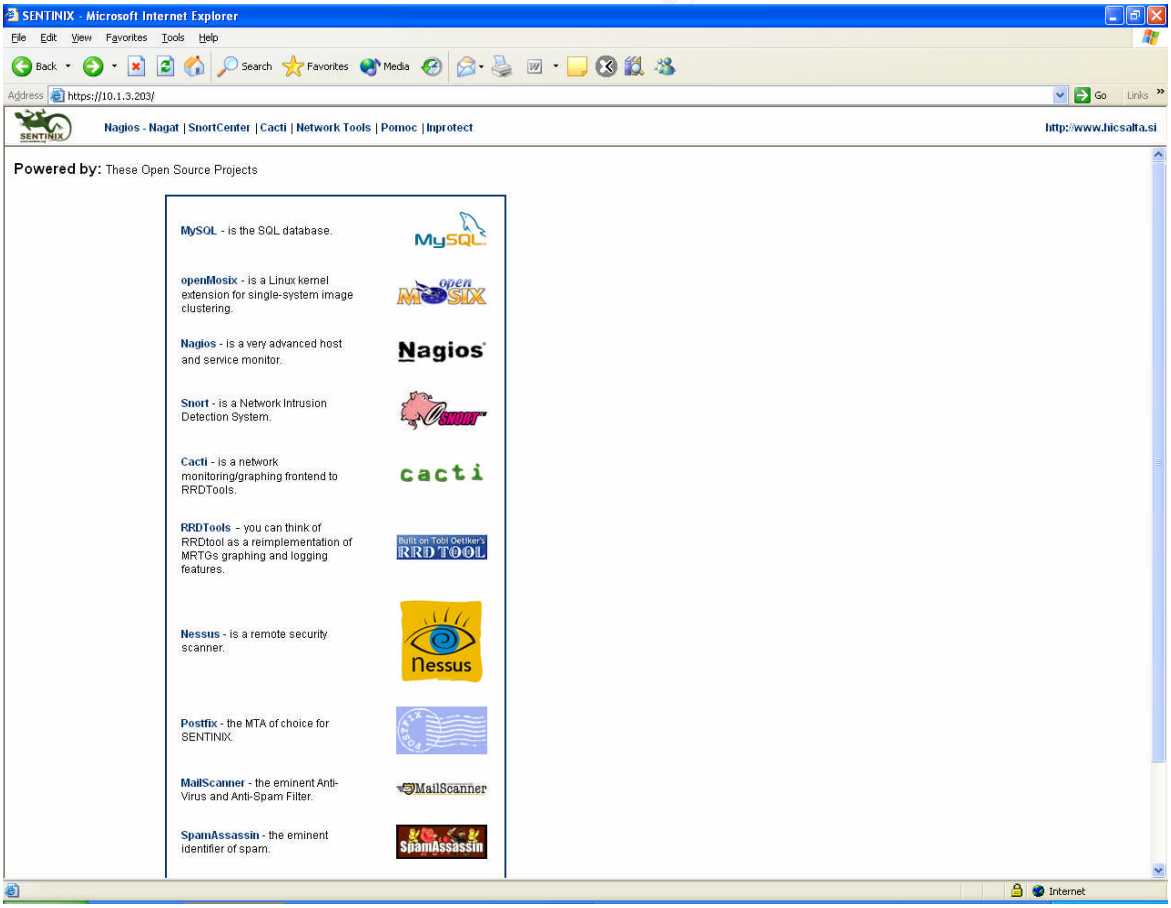


Figure 3: entry point

[7] http://sentinix.org/index.shtml

Systems and network devices performance control

Since I was already using Snort by that time as an intrusion detection system, I started to look for a solution around Snort.

I discovered that using RRDtool together with Perl bindings I can get pretty straight forward solution.

Cacti – performance monitor

Goal is to be a complete frontend to rrdtool, storing all of the necessary information to create graphs and populate them with data in a MySQL database. The front-end is completely PHP driven. Along with being able to maintain Graphs, Data Sources, and Round Robin Archives in a database, Cacti also handle the data gathering. Basic SNMP[8] support is included which allows users to easily create traffic graphs with only a few clicks.

There are many features that make the creation of rrdtool-based graphs faster and easier. Users can create one CDEF and apply it to all of their graphs. Want to start graphing ping times to another host? Simply add a data source; and fill in the necessary fields needed for the "Ping Host" script. That data can then be added to any graph to show the newly gathered data.

Some Cacti's features:

Complete rrdtool front-end; implements data sources, round robin archives, graphing functions, and consolidation functions,

Contains tools necessary to gather data and store it in an .rrd database,

Allows users to build graphs based on the data that is gathered,

Supports multiple ways for displaying the created graphs,

Implements basic SNMP support used to easily create MRTG-like traffic graphs,

User based management allows administrators to create users and assign different levels of permissions,

Everything can be done from a web interface, making the creation of new graphs much more efficient,

Ability to see what command is being passed to rrdtool for the create/graph/update functions while inside Cacti.

[8] http://www.ee.oulu.fi/research/ouspg/protos/testing/c06/snmpv1/

User interface

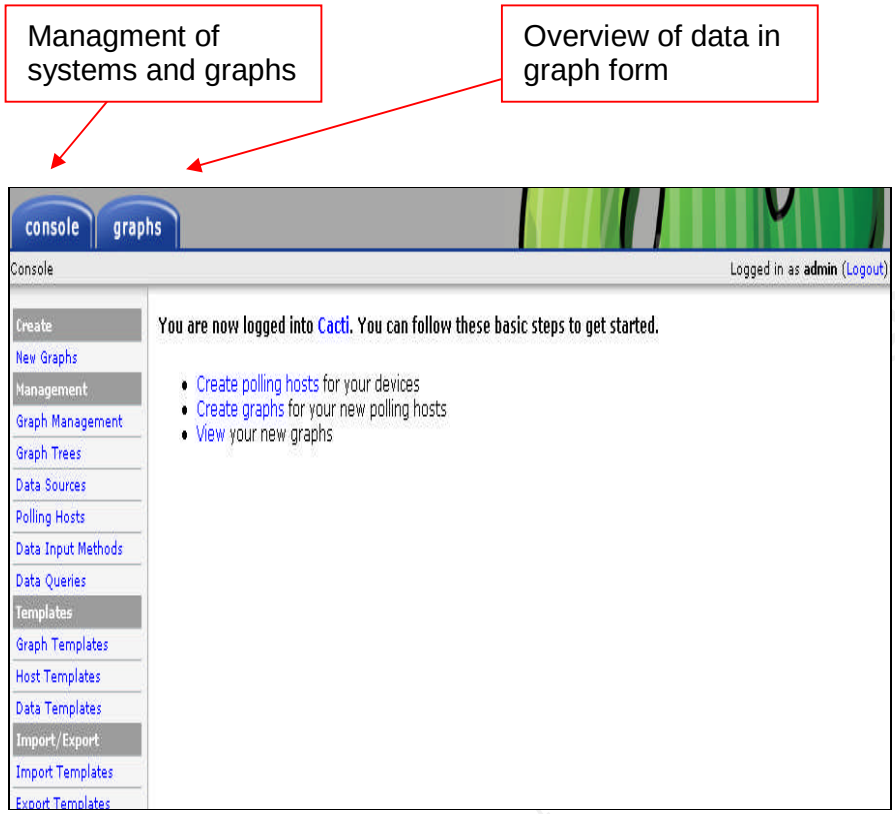


Figure 4: User interface

SNMP testing of the system before including it in the application

Before we include system in Cacti we should try if SNMP service is running properly. We test the service with SNMP Browser within Network Tools:

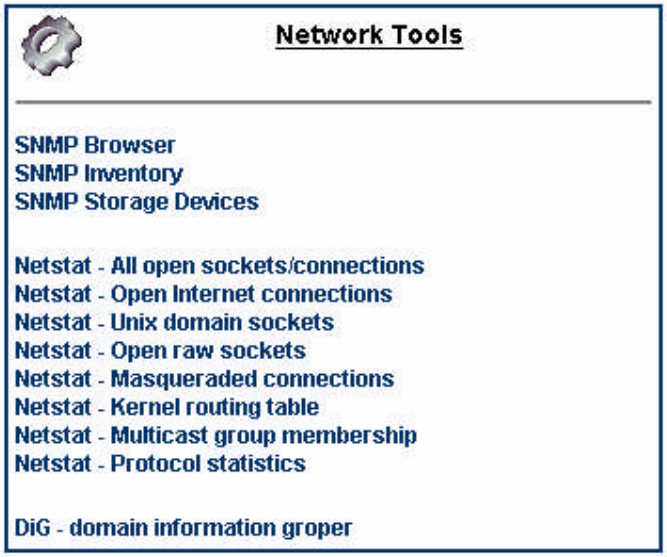


Figure 5: Network tools

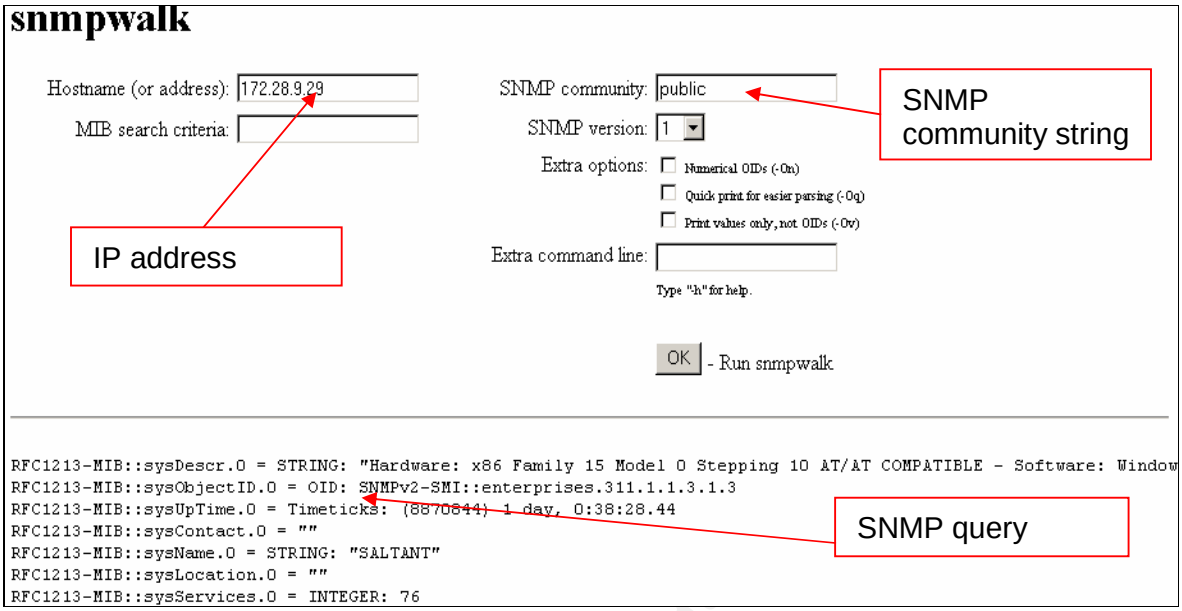
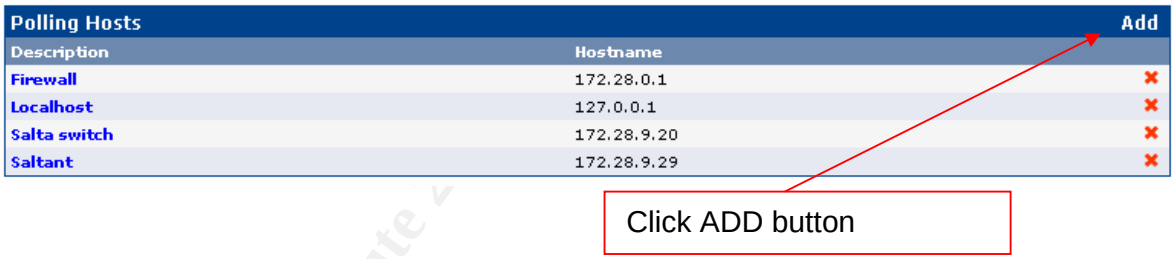
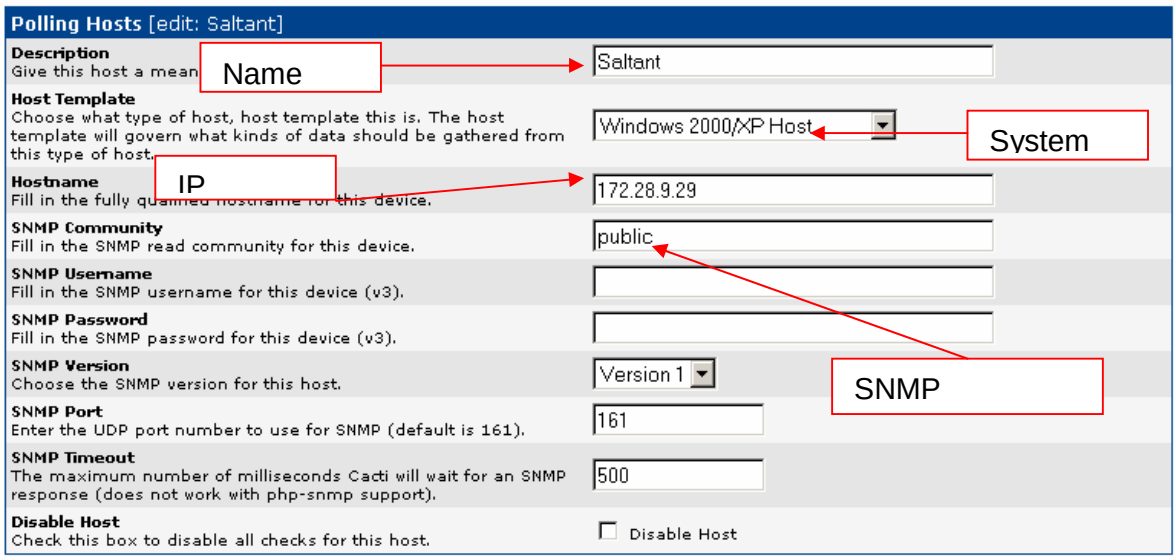


Figure 6: Verifying SNMP query

Create a new system (Polling hosts)

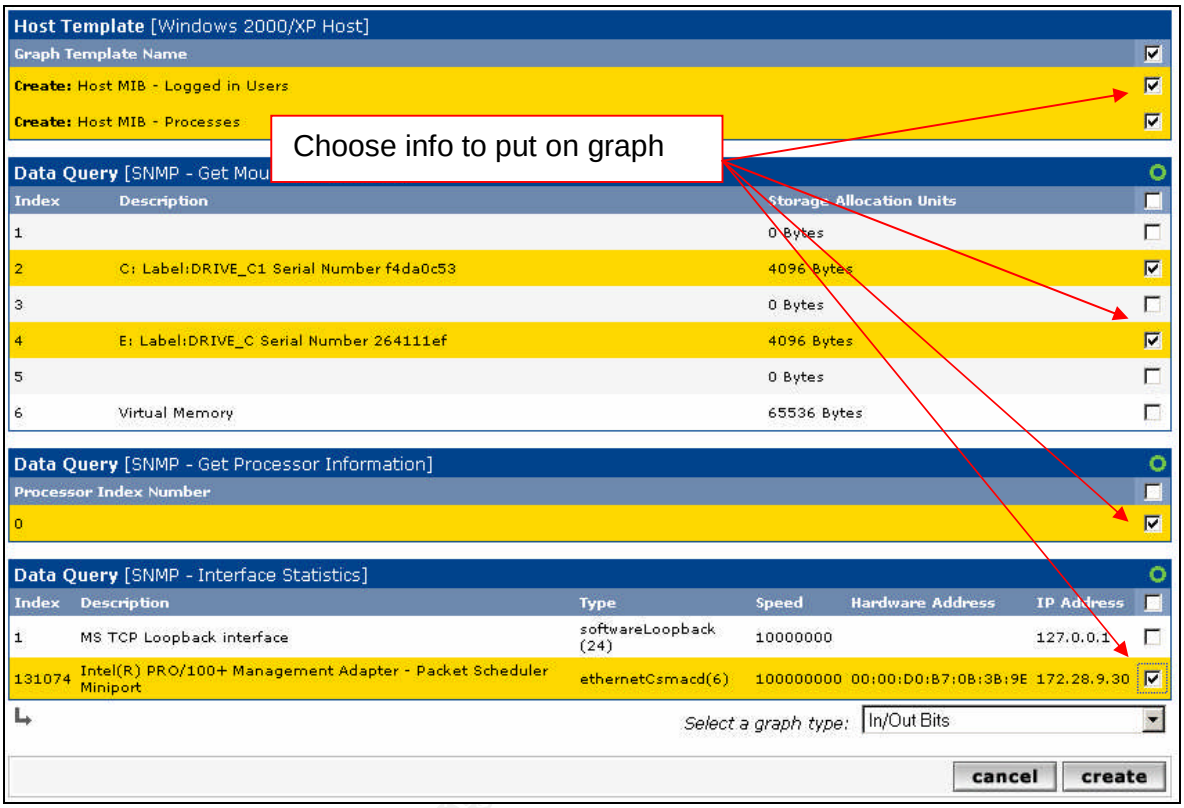
IF SNMP is working properly we add a new system in polling cycle:



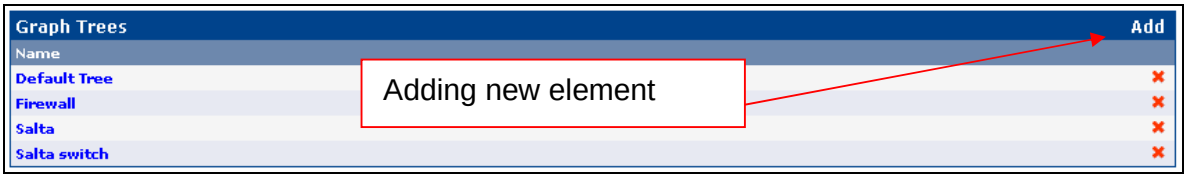


Creating new graphs for the system (New Graphs)

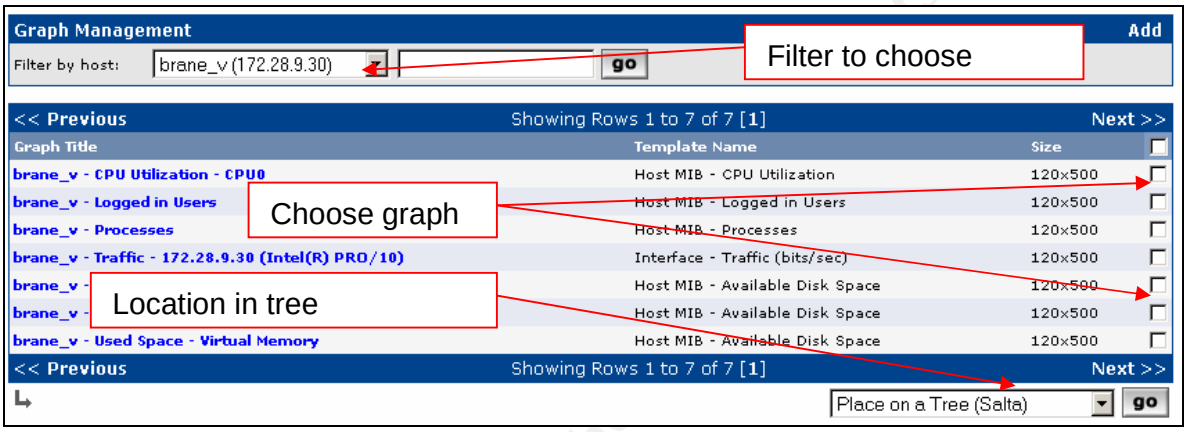
Next we define what we want to see included in our graph presentation, depending which system we want to control:



Adding graphs in tree structure (Graph Trees)



Choose what you want to include in the graphs



Security control

Snort

As I mentioned before, I was using Snort before. There is also a lot written about Snort in SANS literature. We use SNORT as a network intrusion detection system. I wanted to make a step forward in processing the events produced by Snort. For that purpose I decided to use Acid open source solution. There is a lot written in SANS documentation, so I will not concentrate on it within the document.

Acid

The Analysis Console for Intrusion Databases (ACID) is a PHP-based analysis engine to search and process a database of security events generated by various IDS, firewalls, and network monitoring tools. The features currently include: Query-builder and search interface for finding alerts matching on alert meta information (e.g. signature, detection time) as well as the underlying network evidence (e.g. source/destination address, ports, payload or flags).

Packet viewer (decoder) graphically displays layer-3 and layer-4 packet information of logged alerts

Alert management by providing constructs to logically group alerts to create incidents (alert groups), deleting the handled alerts or false positives, exporting to email for collaboration, or archiving of alerts to transfer them between alert databases.

Chart and statistics generation based on time, sensor, signature, protocol, IP address, TCP/UDP ports, or classification

ACID has the ability to analyze a wide variety of events, which are post-processed into its database. Tools exist for the following formats: Snort, tcpdump binary logs,

using logsnorter (www.snort.org/downloads/logsnorter-0.2.tar.gz), ipchains, iptables, ipfw.

Since I was using Snort already before, I decided to use Snort format.

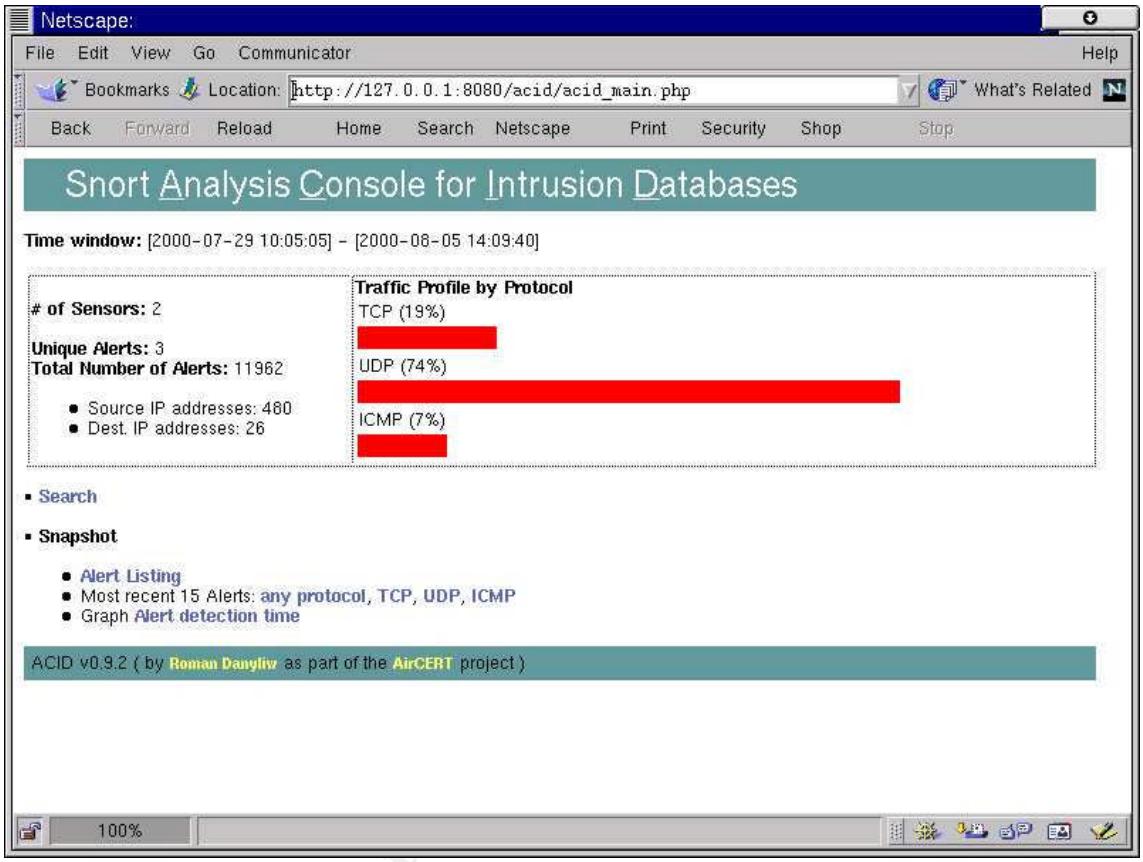


Figure 7: ACID console

Nessus&Inprotect

Next phase was to set up security controls. I decided to use Nessus as a vulnerability scanner.

After installing it, if I had any troubles I always used following URL:

http://www.linuxsecurity.com/feature stories/nessusintro-printer.html

After implementing Nessus as a vulnerability scanner, I decided to use Inprotect[9] as a GUI interface to Nessus, because it offered me the possibility to schedule vulnerability scans on the regular basis and make all this process automatic.

Here are some basic Inprotect features:

Use of open source vulnerability scanners: Nessus and Nmap.

PostNuke and standalone modules.

9 http://www.giac.org/practical/GSEC/Dusty_Hall_GSEC.pdf

Use of open source products and programming languages: Apache, MySQL, PHP, Perl, Linux.

Manual and scheduled security scans.

Free - released under GNU GPL.

Reports generation in PDF format.

SLA's control

Nagios
is a host and service monitor designed to inform you of network problems before your clients, end-users or managers do. It has been designed to run under the Linux operating system, but works fine under most *NIX variants as well. The monitoring daemon runs intermittent checks on hosts and services you specify using external "plugins" which return status information to Nagios. When problems are encountered, the daemon sends notifications out to administrative contacts in a variety of different ways (email, instant message, SMS, etc.). Current status information, historical logs, and reports can all be accessed via a web browser.

Here are some features from Nagios that I used:

Monitoring of network services (SMTP, POP3, HTTP, NNTP, PING, etc.),

Monitoring of host resources (processor load, disk and memory usage, running processes, log files, etc.),

Simple plugin design that allows users to easily develop their own host and service checks,

Ability to define network host hierarchy, allowing detectionand distinction between hosts that are down and those that are unreachable,

Contact notifications when service or host problems occur and get resolved (via email, pager, or other user-defined method),

Optional escalation of host and service notifications to different contact groups,

Ability to define event handlers to be run during service or host events for proactive problem resolution,

Support for implementing redundant and distributed monitoring servers,

External command interface that allows on-the-fly modifications to be made to the monitoring and notification behavior through the use of event handlers, the web interface, and third-party applications,

Retention of host and service status across program restarts,

Scheduled downtime for supressing host and service notifications during periods of planned outages,

Ability to acknowlege problems via the web interface,

Web interface for viewing current network status, notification and problem history, log file, etc.,

Simple authorization scheme that allows you restrict what users can see and do from the their web interface.

There are also some others features I did not use, and have no comment:

Monitoring of environmental factors such as temperature,

For the implementation and use of Nagios I used documentation found on URL: http://www.nagios.org/docs/

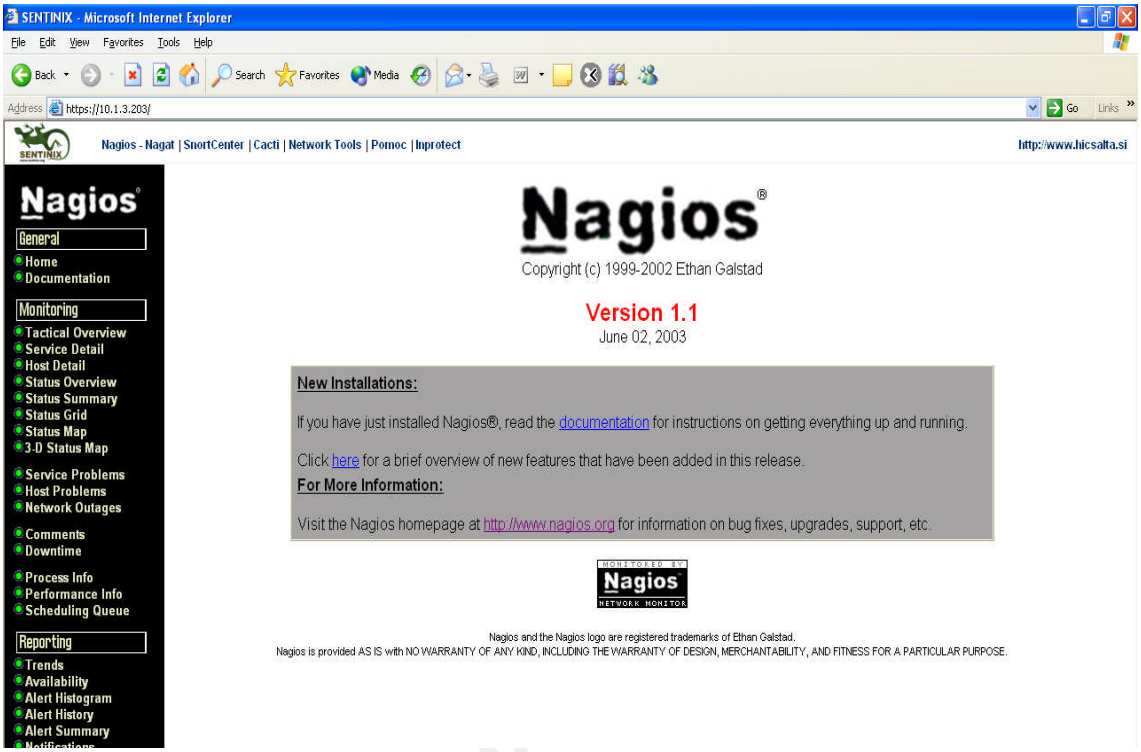


Figure 8: Program entry mask

System definition:

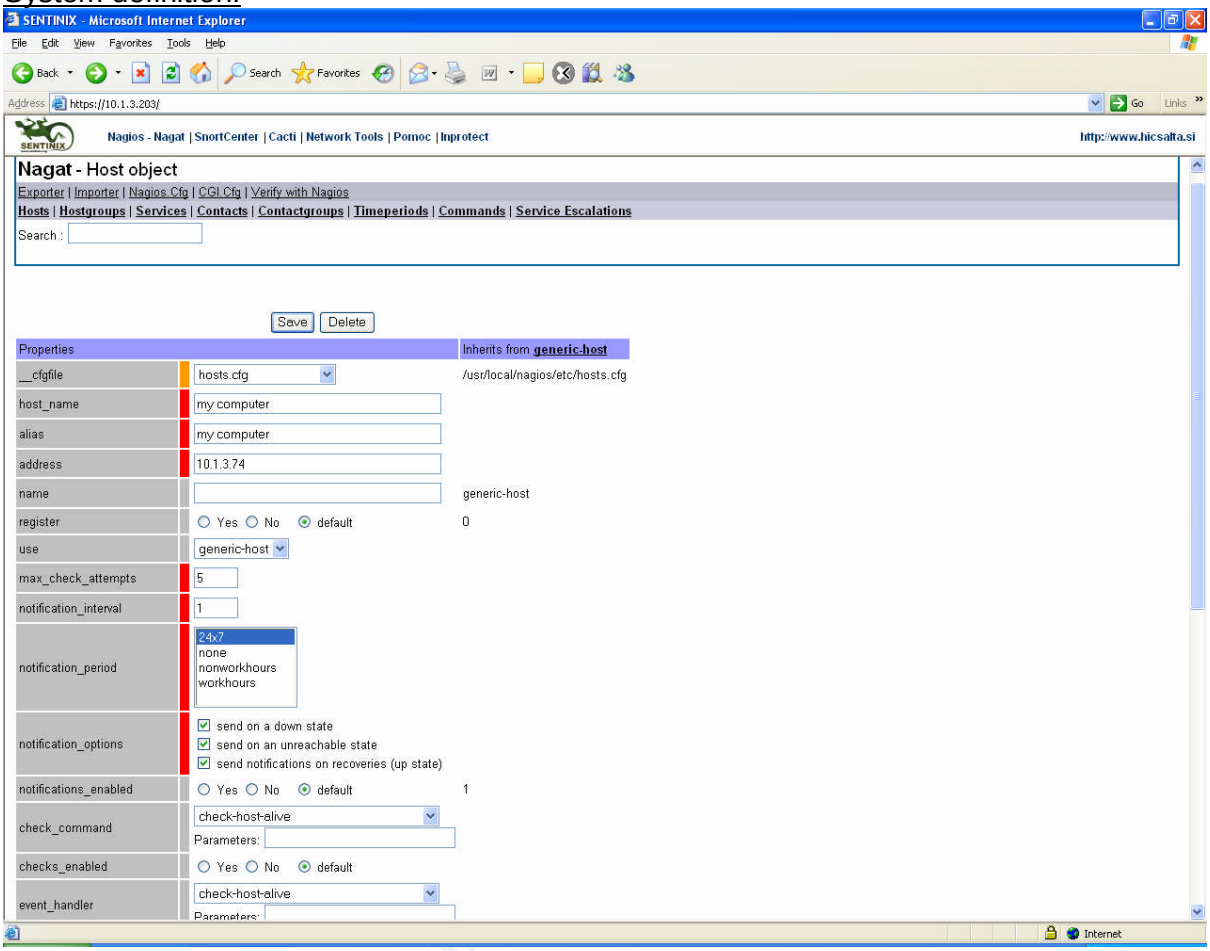


Figure 9: Defining SLA data for a specific system

Status overview screen

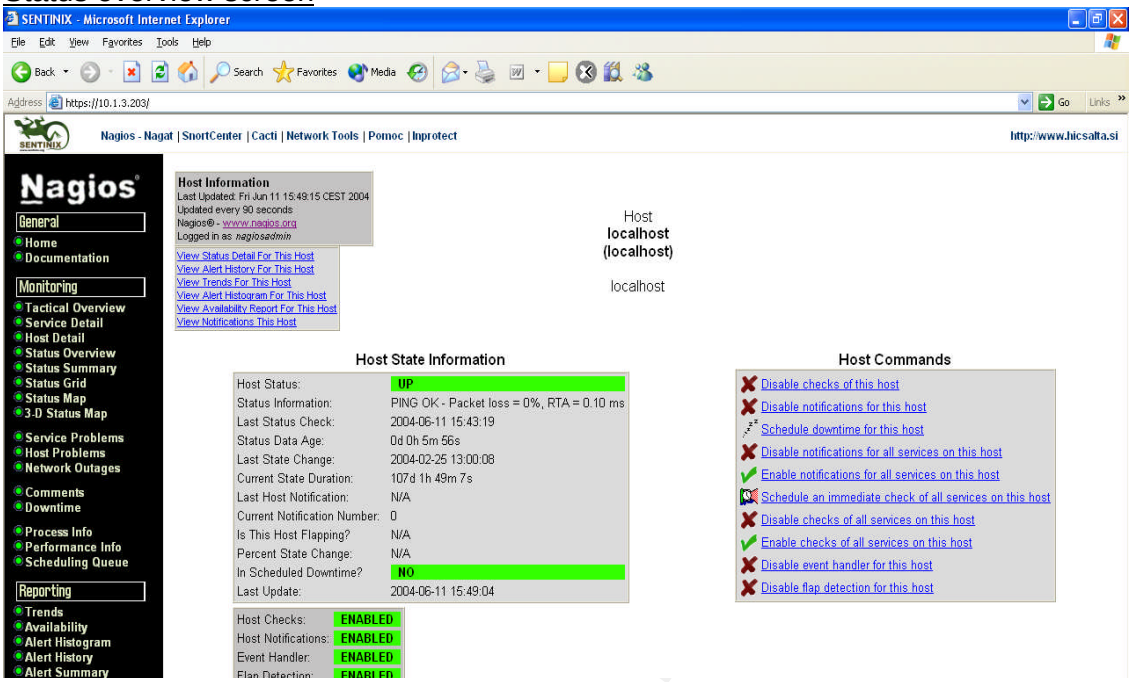


Figure 10: Status information of specific system

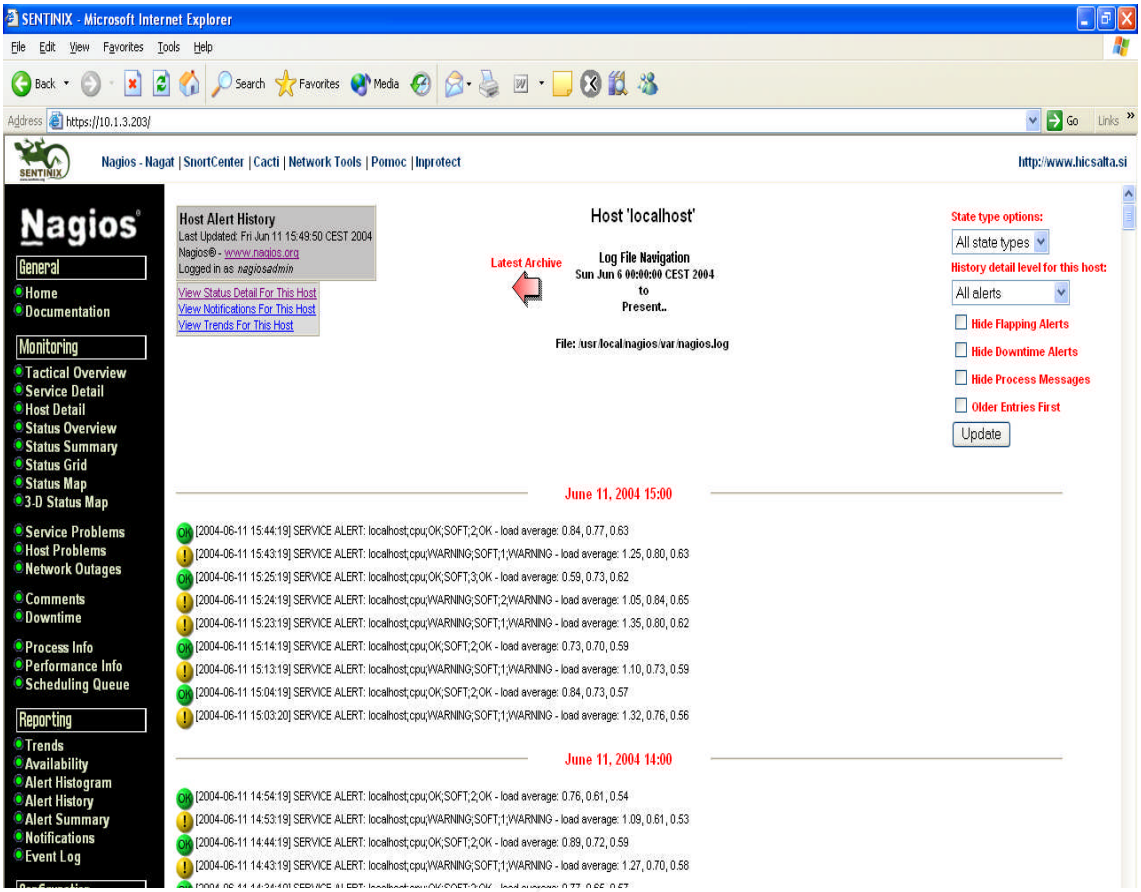


Figure 11: Log information about running service

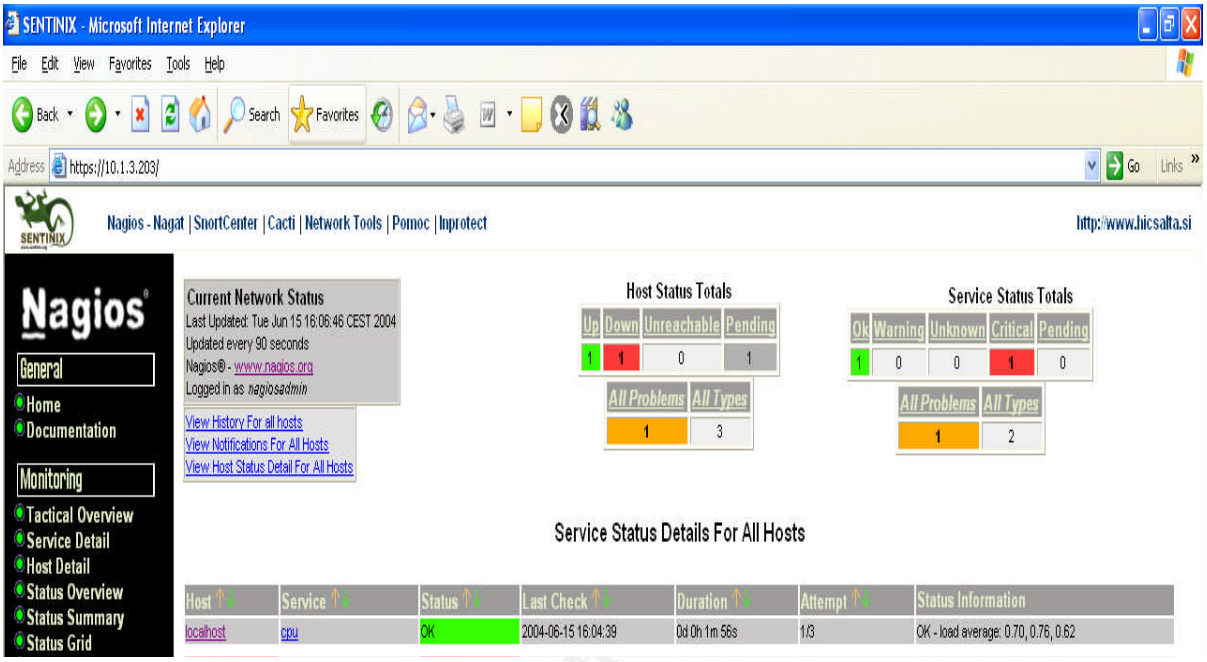


Figure 12: Service status control screen for all SLA's

3. After

After implementing monitoring system over our systems, applications and communications we can finally start with reactive measures like ensuring that the health of all critical systems is monitored constantly. Insurance for security auditing is automated and includes alerts for problem conditions. Next, we can provide management with monthly reports for all critical systems. Although few IT products provide sufficient manageability out of the box, a robust market of third party products exist to provide monitoring, automated management, automated corrective actions, reliable asset migration and reorganization and reporting[10]. These products can add significant manageability to your IT infrastructure, making an optimized or agile environment easier to manage.

Once we have established a system that answers to the question where we are at the moment - the beginning of our measurement program, we can concentrate on the next two goals within our measurement initiative, which is the answer to the question where we came from – a single point of reference internally over time and the last and ultimate goal giving us the possibility to step from reactive to proactive control - answers to the question where we can go – comparing organizational reference points with known and valued external references[11].

10 http://infosecuritymag.techtarget.com/2003/jun/turnover.shtml
11 http://www.netiq.com/offers/cioebook/default.asp

4. References

Appendix A

IT Security; Risking the Corporation, Linda McCarthy

Web Services Security, Mark O'Neill

Information Security Policies Made Easy, 9the Edition, Charles Cresson Wood

Computer business review 2004 – Security Mandates

Appendix B

Sentinix - distribution designed for monitoring, intrusion detection, vulnerability assessment, statistics/graphing and anti-spam: http://sentinix.org/index.shtml

PROTOS Test-Suite: c06-snmpv1:

http://www.ee.oulu.fi/research/ouspg/protos/testing/c06/snmpv1/

Magic Quadrant for Enterprise Firewalls:
http://mediaproducts.gartner.com/gc/webletter/netscreen/issue2/gartner1.html

Managing and securing the Enterprise:
http://www.netiq.com/offers/cioebook/default.asp

Business Continuity Guides: http://www.thebci.org/BCAWG.html

Securing by effective auditing:
http://www.giac.org/practical/GSEC/Dusty_Hall_GSEC.pdf

Saita, Anne. "Turnover at the Top." June 2003:
http://infosecuritymag.techtarget.com/2003/jun/turnover.shtml

Slater, Derek. "Identity Crisis." The CIO Role. June 2003:
http://www.csoonline.com/read/060103/crisis.html

Berinato, Scott. "Bob Moore Knows How Not to Get Fired." June 2003:
http://www.csoonline.com/read/060103/fired.html

Vijayan, Jaikumar. "IT managers see need for risk metrics." June 9, 2003.

http://www.computerworld.com/securitytopics/security/story/0,10801,81897,00.html

Briney, Andrew. "Proving Ground." September 2002.

http://infosecuritymag.techtarget.com/2002/ciso/aug/ciso-roundtable.shtml