Monitorització d'un Centre de Processament de Dades d'una administració pública, mitjançant *Nagios* i *Centreon*

Projecte en administració de xarxes i sistemes operatius en entorns de programari lliure

Josep Pagès i Lozano
Juny de 2011

Consultor: Miguel Martín Mateo

UOC
Universitat Oberta de Catalunya
www.uoc.edu

This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Unported License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/3.0/ or send a letter to Creative Commons, 444 Castro Street, Suite 900, Mountain View, California, 94041, USA.

1 Introducció i resum del projecte.

El present projecte té com a àmbit d'actuació el centre de processament central (CPD) de dades d'una administració pública. Actualment, aquest és gestionat per personal propi de l'administració, enquadrat dins del seu Departament d'Informàtica, i es troba majoritàriament ubicat en dues seus allunyades uns pocs quilòmetres.

Els serveis que ara mateix està oferint el Departament d'Informàtica a aquesta administració són molt nombrosos, d'abastos diversos i basats en plataformes de maquinari i programari ben variats. Des de fa uns anys, s'ha iniciat una virtualització massiva del CPD que n'ha facilitat la gestió i ha facilitat l'expansió quasi explosiva del nombre de màquines servidores, tot això amb el mateix personal per a gestionar-les. El nombre de màquines servidores actuals, entre físiques i virtuals, és superior a les 200, oferint tota mena de serveis: bases de dades, webs dinàmiques, servidors d'aplicacions i un llarg etcètera.

És la responsabilitat del Departament d'Informàtica de garantir la màxima disponibilitat i continuïtat de tots aquests serveis. Una condició absolutament necessària per a assegurar que siguin bones és assabentar-se de diversos esdeveniments que poden acabar tenint impacte sobre els serveis abans que aquest impacte tingui lloc. Uns exemples podrien ser saber quan un disc dur d'un servidor és força ple o molt ple, quan tenim pics de consum de CPU, quan un servidor web no respon peticions HTTP, quan un servidor DNS no respon correctament les preguntes DNS, etc. En resum, és imprescindible fer una monitorització proactiva de la infraestructura física i de servidors, com a mínim d'un conjunt de paràmetres mesurables obvis de l'estil dels que he esmentat. Això minimitza i pot evitar les incidències amb impacte sobre la disponibilitat i qualitat dels serveis.

Com s'ha dit, el nombre de servidors s'ha multiplicat enormement, i amb data de juny de 2010 no hi havia cap sistema de monitorització automatitzat de serveis i servidors, cosa que generava un nombre d'incidències si bé no gaire gros si de prou impacte real i polític per fer necessari corregir aquesta mancança.

Sorgeix doncs la necessitat d'una monitorització programada i activa, amb capacitat per notificar els responsables per diferents vies. Aquest projecte es centrarà en explicar la solució o solucions que s'han triat, i perquè, per satisfer aquesta necessitat.

Donat que els equips de xarxa – firewall corporatiu, switchos, etc – ja s'estan monitoritzant amb una eina propietària de Cisco, l'abast d'aquest treball no els inclourà en la seva fase inicial objecte de la present memòria.

Índex de continguts

1Introducció i resum del projecte....3
2Estudi de viabilitat....6
2.1Necessitats i requisits del client....6
Requisits del client....6
Objectius a satisfer....6
Recursos econòmics i recursos humans....7
Cost de manteniment....7
2.2Anàlisi de la situació actual....7
Situació actual del CPD....8
Infraestructura física....9
Servidors de virtualització i emmagatzemament compartit....9
Estructura i topologia de la xarxa....10
2.3Definició de requisits del sistema....11
2.4Solucions possibles i elecció....12
OpenNMS (http://www.opennms.org/)....13
Zabbix (www.zabbix.com)....14
Nagios....14
SNMP....15
Centreon....16
Cacti....16
3Anàlisi del sistema....17
3.1Definició del sistema....17
3.2Requisits exactes del projecte....17
3.3Definició d'interfícies d'usuari....17
3.4Especificació pla de proves....17
4Disseny del sistema....19
4.1Arquitectura....19
Servidor....19
Sistema operatiu i distribució Linux....19
Configuració de xarxa....19
Nagios....19
Instal.lació....19
Documentació prèvia i prerequisits....19
Instal.lació de Nagios....20
Configuració....21
A /usr/local/nagios/etc:....21
A /usr/local/nagios/etc/keys:....25
A /usr/local/nagios/etc/objects/....25
Monitorització de Windows mitjançant NSClient++....34
Instal.lació de NSClient++ en un servidor Windows 2000/2003/2003 R2 de 32 bit....35
Creació del fitxer de configuració de Nagios “nom_servidor_windows.cfg” (a l'exemple “servidormailboxesexchange” per a nova_maquina en el servidor de monitorització....39
Monitorització de Linux: per SSH....41
Incorporar una nova màquina per ser monitoritzada per ssh amb nagios....41
Tot seguit, la instal.lació dels plugins....42
Testejar i posar a punt la connexió des de monitor --> nom_servidor....42
Creació del fitxer de configuració de Nagios “nom_servidor_linux.cfg” per a nova_maquina en el servidor de monitorització....42
Centreon....44
Instal.lació sobre Red Hat 5.x Enterprise....44

Instal.lació sobre una Debian....47
Preparatius i prerequisits: http://en.doc.centreon.com/Setup:Prerequisite/Debian/Ubuntu 48
Instal.lació Nagios: http://en.doc.centreon.com/Setup:CompileNagios....48
Instal.lació plugins Nagios: http://en.doc.centreon.com/Setup:CompileNagiosPlugins....50
Instal.lació NDOUtils : http://en.doc.centreon.com/Setup:ndoutils....51
Instal.lació Centreon: http://en.doc.centreon.com/Setup:Centreon2....52
Configuració Centreon – Interfície Web....53
Tallafocs....58
Procés de desplegament i implantació....59
4.2Requisits d'implantació....64
Seguretat....64
5Desenvolupament....65
5.1Planificació de les activitats d'integració del sistema....65
Setembre de 2010 – juny 2011:....65
Maig de 2011 – Juny de 2011....65
5.2Llicència....65
5.3Entorn de desenvolupament....65
5.4Documentació....65
6Implantació....66
6.1Formació....66
6.2Implantació del sistema, proves i nivell de servei....66
6.3Acceptació del sistema....66
6.4Manteniment....66
Actualització del sistema operatiu (automàtic)....66
Actualitzacions del programari....67
Incorporació de nous servidors i serveis....67
Còpies de seguretat de les dades. 67
Backups de Nagios. 68
Backups per a Nagios-Centreon....69
7Conclusions i assoliments....70
8Annex A. Breu mostra de dues alternatives....72
8.1OpenNMS (http://www.opennms.org/)....72
8.2Zabbix (www.zabbix.com)....81

2 Estudi de viabilitat

2.1 Necessitats i requisits del client

Requisits del client

El meu client és el Departament d'Informàtica de l'Administració X, que té una clara necessitat de poder visualitzar, en un moment donat, l'estat del CPD d'una manera general i global, però amb prou precisió per saber si hi ha alguna afectació dels serveis crítics o si pot haver-n'hi una en un futur proper si no es pren una acció preventiva.

Objectius a satisfer

Els objectius que es pretenen amb aquest projecte es poden esquematitzar en els següents punts:

- Reduir el nombre d'incidències i afectacions sobre els serveis que es proporcionen als usuaris de l'Administració X per part del seu Departament d'Informàtica.
- Notificar de potencials problemes i esdeveniments amb probable impacte sobre els serveis abans que s'hagi produït una afectació negativa sobre qualsevol servei, i poder notificar selectivament els responsables de diversos serveis de maneres diferents i amb requeriments diferents.
- Ser capaços de monitoritzar i tenir una visió global de l'estat de tota la infraestructura de servidors i serveis en un punt central.

Durant el transcurs del projecte, el Departament d'Informàtica i l'Administració X han implantat un procés de certificació de la ISO20000 per a alguns dels serveis que s'ofereixen. Aquesta certificació exigeix poder presentar tota una sèrie d'evidències, especialment de disponibilitat i capacitat, que han obligat a estudiar la implantació de noves solucions basades en *Nagios* que puguin satisfer aquests nous requeriments. Això ho resumiríem amb un nou requisit ja era previsible però ara és necessari:

- Ser capaços de generar informes sobre la disponibilitat i la capacitat dels servidors i equips de la infraestructura, i generar gràfics de tendències de consum i capacitat.

Exemples d'esdeveniments que haurien de poder-se detectar abans que hi hagués una notificació d'incidència per part dels usuaris podrien ser:

- Discs durs al límit de la capacitat o plens: logs, especialment → denegació de servei.
- Serveis (daemons) que no funcionen correctament, responen lentament, o simplement no responen i que per tant tenen el servei limitat o el deneguen.
- Recursos de les màquines saturats: CPUs al 100%, excessiu I/O

Així doncs, el que es vol és la monitorització en general del CPD del Departament d'Informàtica de l'Administració X, i com a mínim del maquinari i servidors més crítics el mal funcionament dels quals tingui un impacte immediat o elevat tant sobre la imatge de l'Administració X com sobre els serveis que es donen als usuaris, i que pugui afectar la productivitat tant de l'administració com general dels seus usuaris/clients.

Aquesta monitorització ha de permetre la identificació preventiva i anticipada de situacions que puguin representar una amenaça o un problema futur per als serveis d'informació que s'estan oferint, i sobretot ha d'emetre notificacions i avisos tant d'aquestes possibles situacions perilloses com de les interrupcions reals de servei que puguin esdevenir-se.
Bé que molt interessant, una monitorització passiva es deixa per més endavant. La monitorització passiva és la que permet que diferents equips envïin activament informacions al monitor sense que aquest els n'hagi demanat explícitament. Permetre i tractar *traps* SNMP seria un exemple de monitorització passiva.

Recursos econòmics i recursos humans

Dins del context actual de restriccions econòmiques generalitzades, és molt valorable que el cost directe del projecte sigui molt baix. En aquest sentit, una solució amb programari de codi obert té un cost directe de llicències de zero. Donat que la implantació es farà per part del que està fent aquest projecte, el cost de recursos humans de la implantació també serà quasi zero (és inevitable un solapament en alguns aspectes de la implantació, donat que estic treballant al Departament d'Informàtica i per tant no seria just dir que és estrictament zero, però és una molt bona aproximació).

Cost de manteniment

Qualsevol solució que es proposi hauria de passar doncs per codi lliure, preferentment sobre Linux.

Ja que històricament s'ha treballat molt amb Linux i programari lliure al Departament d'Informàtica, i hi ha diversos sistemes importants implementats amb programari lliure (directori corporatiu amb OpenLDAP, sistema de correu amb *postfix*) hi ha l'expertesa necessària perquè es pugui mantenir amb fiabilitat qualsevol solució implantada sobre RedHat Linux; aquest manteniment inclou coses òbvies com les còpies de seguretat (*backups*) i les actualitzacions del sistema operatiu, tant a nivell de fitxers com de recuperabilitat global. L'impacte que pot tenir mantenir un servidor més és relativament moderat, un cop establerts procediments generals d'actuació per a algunes coses essencials com les esmentades.

Els costos més importants són:

- L'actualització de les dades dels servidors en el *monitor,* així com afegir-hi noves dades o nous servidors.

- El cost de manteniment del sistema. Aquest necessàriament haurà de tenir en compte la necessitat d'actualitzar els diferents mòduls del programari, tant els del servidor com els possibles *agents* que calgui instal.lar en els servidors clients que s'hauran de monitoritzar, a mesura que surtin noves versions i sobretot *pedaços* de seguretat.

- També cal preveure el cost d'obsolescència de la solució, cas que el producte pugui deixar d'estar mantingut, o surtin nous productes que facin el nostre obsolet.

2.2 Anàlisi de la situació actual.

Com ja s'ha comentat doncs, la situació actual és dispersa pel que fa a la monitorització. Si bé els equips de xarxa crítics es monitoritzen amb programari propietari per part de l'equip humà de comunicacions, la monitorització de la resta de sistemes és manual, visual i reactiva, i no hi ha cap eina que pugui oferir una visió global de l'estat dels sistemes.

L'estat dels sistemes, registrat a l'"Event viewer" de Windows i al *syslog* dels Linux no es miren amb regularitat.

S'ha configurat *logwatch* a alguns dels sistemes Linux més crítics, però aquest envia correus a una

bústia de suport que al capdavall sovint no es llegeixen, la qual cosa fa aquest sistema impràctic i poc útil.

Un altre component crític de la infraestructura és la cabina de discos o *emmagatzematge compartit*: per a aquest, existeix una eina en una màquina dedicada que un cop al dia algú es mira per si hi ha errors. L'aplicació és propietària.

Els servidors de virtualització reporten encara a una altra màquina dedicada, que té també un programari propietari instal.lat per a controlar-les, cal mirar-lo amb regularitat també.

S'ha subcontractat a una empresa externa a l'Administració X la monitorització d'alguns sistemes crítics, com pot ser el web de l'Administració X (que respongui peticions HTTP), el servidor d'entrada de correu SMTP, etc, però són uns poquíssims serveis del tot insuficients per tenir aquesta visió clara imprescindible, i per poder fer cap mena de *reporting* als òrgans de gestió i decisió de la Administració X.

Situació actual del CPD

Caldrà tenir en compte:

- Aproximadament 220 màquines virtuals, de les quals 183 servidors, a juny de 2011, que realment donen servei directe: SQL, fitxers, web, aplicacions web, etc. La resta són *desktops* virtuals o *escriptoris corporatius* , en Windows 7 i en Windows XP.

  L'entorn de virtualització és VMWare vSphere 4.0. El procés de virtualització de servidors a l'Administració X es va iniciar fa anys amb unes proves pilot amb VMWare ESX 2.5, i es va anar migrant el pilot fins a la versió 3.0. Amb la implantació de vSphere 3.5 és quan va començar el desplegament i la migració massius cap a la infraestructura de virtualització. Amb data de juny de 2011 hi ha uns 183 servidors virtualitzats en funcionament, proporcionant tota mena de serveis. Excloent els servidors físics que tenen l'hipervisor, també hi ha unes vuit màquines de mal virtualitzar, per diversos motius, algunes d'elles són:

- Un servidor físic de NFS d'alta càrrega, servint desenes de milers de fitxers petits per segon. El seu rendiment servint fitxers en un entorn virtualitzat era un ordre de magnitud inferior.
- Un servidor físic que virtualitza un VMS mitjançant un programari de virtualització propietari de Charon.
- Dos servidors corporatius en Cluster de Microsoft SQL Server 2000, que hostatgen de moment bona part de les bases de dades corporatives. Per a aquests es preveu una migració properament a entorn virtualitzat en Microsoft Windows 2008 R2 Server + SQL Server 2010 R2.
- 26 servidors físics virtualitzadors HP Blade BL460c
- 3 cabines de disc de canal de fibra de Bull, Optima 3000, 2900 i 2500.
- Molts equips físics de xarxa, monitoritzats ja per l'equip de comunicacions amb eines diverses.

Els sistemes operatius que es fan servir són Microsoft Windows i Linux.

Pel que fa a Microsoft Windows, hi ha (quasi) totes les versions empresarials dels darrers deu anys:

- Windows 2000: 9 servidors
- Windows 2000 Advanced Server: 2 servidors.
- Windows Server 2003, Enterprise Edition (32 bit): 3 servidors.

- Windows Server 2003, Standard Edition (32 bit): 51 servidors.
- Windows Server 2008 (64 bit) : 2 servidors.
- Windows Server 2008 R2 (64 bit): 23 servidors.
- Windows XP (entorn Escriptori Remot Corporatiu i Aules Virtuals) : 26 màquines + 50 màquines, que no cal monitoritzar.

Quant a Linux, hi ha una immensa majoria de Red Hat Enterprise Linux 5.x, i dos Debian.

- Linux genèric 32 bit: 2 servidors.
- Red Hat Enterprise Linux 5 (32 bit): 49 servidors
- Red Hat Enterprise Linux 5 (64 bit): 1 servidor.
- Debian 5.x: 2 servidors.

Es veu un predomini clar de sistemes de 32 bit, tant en entorn Windows com Linux. Fins recentment, aquests sistemes satisfeien adequadament les necessitats dels nostres usuaris i per tant, eren la plataforma de servidor homologada: Windows Server 2003 R2 x32 i RedHat Enterprise Linux 5.x 32-bit.

Però el fet que en el darrer any i mig haguem passat de la plataforma virtualitzadora VMWare ESX 3.5 a vSphere 4.0, que ha canviat d'arquitectura i ha passat de 32 bits a 64 bits, i la pròpia evolució dels sistemes operatius de servidor, juntament amb les demandes cada cop més exigents pel que fa als serveis que se'ns demanden ha fet que ja haguem instal.lat i adoptat com a nova plataforma servidora homologada Windows Server 2008 R2 , que únicament és en la versió de 64 bit, i que jo hagi decidir adoptar, per a la implantació del sistema monitoritzador, un servidor RedHat Linux 5.x en 64 bit, el primer que hi haurà a la Administració X. Així doncs, l'únic servidor RH5 de 64 bits que figura en la quantificació anterior és aquest: MONITOR.administraciox.cat

Infraestructura física.

L'abast d'aquest projecte hauria d'incloure la monitorització dels servidors i equips físics i dels servidors virtualitzats. Pel que fa als equips físics, és bo de saber si algun dels components crítics falla (discs durs, per exemple), i dels servidors virtuals volem assabentar-nos de qualsevol esdeveniment que pugui passar que afecti el rendiment o els serveis dels sistemes.

La monitorització de la infraestructura física ja es fa parcialment, amb eines disperses i diverses i equips humans diferents, normalment els responsables de cada servei. És per aquest motiu que la inclusió d'elements físics en la monitorització es deixarà per a estadis avançats d'aquest projecte, que probablement no s'acabi pel que fa al Departament d'Informàtica el juny de 2011.

Servidors de virtualització i emmagatzemament compartit.

La infraestructura de virtualització és formada per 25 servidors HP Blade BL460 C, cadascun amb dues CPU Intel Xeon E5345 de quatre nuclis i 32 GB de RAM, amb molt pocs problemes de rendiment i contenció fins ara. Tots ells estan connectats per canal de fibra al sistema d'emmagatzemament compartit abans explicat.

Les dades corporatives i els discs dels servidors virtualitzats es guarden a les LUNS creades en tres cabines de discs de canal de fibra (*Fiber Channel)* de l'empresa Bull, models Optima 2500, Optima 2800 i Optima 3000 amb aproximadament 20 TB d'espai. A data de juny de 2011 s'està a punt d'afegir-hi 40 TB més i una altra cabina Optima 3000, i 4 servidors blade nous HP BL 480 amb 64 GB de RAM.

Deu servidors *blade* i una cabina (la Optima xxxx) es troben al campus de Montilivi, tots ells connectats de manera transparent amb el campus principal del Barri Vell per dues fibres òptiques

d'alta velocitat de 2 GB de capacitat cadascuna; els 16 *blade* restants i les altres dues cabines de discs són al campus principal.

Existeix un programari propietari que corre sobre Windows per a gestionar i en certa manera monitoritzar aquestes cabines. En aquest, que cal controlar diàriament, es reporten els esdeveniments crítics, com discs que es fan malbé, controladores que s'espatllen, etc.
També hi ha instal.lat un altre programari d'HP sobre Windows per monitoritzar els HP Blade.

Tots dos sistemes i programaris es fan servir i són instal.lats en màquines diferents, ja que existeixen VLANs segmentades per a diversos usos i intencions, com explicaré més avall.
La intenció és integrar la monitorització dels esdeveniments més importants d'aquests equips dins de l'eina que triem, però això quedarà relegat a la part final del projecte o possiblement en una ampliació, donat que ja existeix una eina per a monitoritzar els elements de configuració d'aquestes cabines.

Estructura i topologia de la xarxa.

Tots els servidors del CPD del Departament d'Informàtica de l'Administració X són, des del punt de vista de la xarxa, ubicats en diverses VLANs segons la seva criticitat, accessibilitat pública des d'Internet, i propòsit.
Entre totes aquestes VLANs i també entre elles i tota la resta de VLANs de l'Administració X i Internet hi ha un tallafocs corporatiu, que només deixa passar els ports TCP/UDP/ICMP que s'han demanat específicament.

Això és així per motius històrics i d'herència, donat en el seu moment es va considerar que calia obrir en cada servidor només els ports de xarxa que se sabia que es necessitaven. Això ha comportat una extraordinària feina de manteniment i posada en marxa de qualsevol servei i servidor, però tanmateix ha obligat a un coneixement profund de què calia obrir en cada moment i perquè.

Així mateix, quan ha estat possible, s'ha engegat i configurat adientment el tallafocs local propi de cada sistema operatiu, en el cas de windows el “Windows Firewall” i per a Linux iptables, sovint replicant els ports que s'obrien en el tallafocs corporatiu.

Un breu inventari de les VLANs existents seria:

- VLAN200, subxarxa classe C amb adreçament privat (VLAN200/24), on hi ha els servidors crítics per a la Administració X. Dins d'aquests, s'hi inclouen els servidors de bases de dades corporatius amb dades crítiques/confidencials, els servidors d'aplicacions interns de l'Administració X que ataquen els diversos servidors de bases de dades; els servidors d'autenticació (OpenLDAP i Directori Actiu de Microsoft), etc. Són servidors d'ús intern, molts d'ells atacables des de tota l'Administració X ja que és on hi ha la majoria de clients, i uns pocs amb doble placa de xarxa (també en una altra VLAN amb adreçament públic).

- VLAN201, subxarxa classe C amb adreçament privat (VLAN201/24). Hi ha servidors de desenvolupament, molts d'ells accessibles des de tota l'Administració X.

- VLAN142, subxarxa classe C amb adreçament públic (VLAN142/24). Hi ha servidors que han de ser atacats des d'internet.

- VLAN203, subxarxa classe C amb adreçament privat (VLAN203/24). Aquesta subxarxa a més a més és *no enrutada*, i les màquines que hi són no figuren en el DNS. Aquesta és la VLAN on hi ha la infraestructura de virtualització.

- VLAN156, subxarxa classe C amb servidors hostatjats pel Departament d'Informàtica i que no mantenim. En principi es mantindran fora del sistema de monitorització, donat que la SLA que

tenim per a aquest servei no especifica que calgui fer-ne cap monitorització.

- VLAN206, subxarxa classe C amb adreçament privat (VLAN206/24). Servidors i clients d'escriptori remot amb clients Windows XP i servidors de Citrix sobre Windows 2003 i Windows 2008. També amb servidors d'aplicacions Citrix XenApp.

- VLAN140, subxarxa classe C amb adreçament públic (VLAN140/24) . Servidors i clients del servei d'Escriptori Remot Windows 7.

- VLAN128, subxarxa classe C amb adreçament públic (VLAN128/24) . Hi ha escriptoris de treball per al Departament d'Informàtica i algun servidor amb Windows Terminal Services, que caldrà monitoritzar.

- VLAN136, 4 subxarxes (136,137,138,139), on hi ha els molts d'escriptoris físics corporatius, la immensa majoria Windows i integrats en un un Directori Actiu. També hi ha alguns servidors de Terminal Services, que caldria monitoritzar.

- VLAN 8 – classe C amb adreçament privat (VLAN8/24). Hi ha els equips d'infraestructura bàsica com les cabines de disc. Caldrà monitoritzar-ho, però es deixarà per al final o per a una ampliació del projecte.

- VLAN202 – Hi ha les ILO del servidors HP Blade, les interfície de gestió remota dels servidors HP Blade. Les aplicacions que corren en les *enclosures* dels servidors *blade*, que permeten gestionar-les i gestionar-los.

En tot cas, si el que volem és monitoritzar aquesta estructura complexa, haurem de tenir en compte aquesta topologia de xarxa i la notable varietat de sistemes operatius.

2.3 Definició de requisits del sistema

- De codi obert i lliure.

- Cost (gairebé) zero, tant de programari com de manteniment.

- Facilitat d'ús.

- Extensible. Hauríem de poder monitoritzar qualsevol cosa que se'ns pugui acudir en el futur.

- Mantenible internament amb baix cost.

- Usable amb poca formació.

- Que pugui notificar de diversos esdeveniments i alertes configurables.

- Que pugui notificar a diferents responsables de diferents serveis i servidors, per múltiples vies: correu electrònic, SMS, etc.

- Que pugui monitoritzar diverses plataformes: Linux, Windows, equips físics per SNMP.

- Fer còpies de seguretat regulars de les dades generades.

- Sistema 24 x 7.

Durant el transcurs del projecte, ha sorgit un nou requeriment:

- Poder generar gràfiques sobre l'estat històric de diversos components dels sistemes, com el consum de processador i memòria dels servidors, per exemple.

Les prioritats serien:

- Poder saber quan un servei crític ha deixat de funcionar.

- Poder saber quan un disc dur és a punt d'omplir-se.

- Poder saber quan perdem connectivitat amb un equip.

- Notificar selectivament diverses persones en funció del servei que ha caigut, i per diversos mètodes.

2.4 Solucions possibles i elecció

Una de les coses que pot venir al cap a un projectista de TFM novell que es plantegi la monitorització d'un CPD són les eines "serioses" o propietàries, com puguin ser OpenView, etc.

Ara bé, a la Administració X hi ha sistemes de monitorització parcials, duts pels equips de sistemes i comunicacions del propi Departament d'Informàtica, que vigilen parts concretes i especialitzades de la infraestructura. Això sí, no hi havia una visió central del tipus "pissarra" (*dashboard*) . La majoria d'aquestes eines de monitorització comentades són de codi propietari, subministrades per cada fabricant per a vigilar el seu equip o la seva infraestructura. Així, un navegador web connectat a la VLAN on viuen les ILO (dispositiu IP de control dels equips HP) s'hi connecta, i les ILO tenen integrat el programari per a monitoritzar els dispositius *blade* i la cabina (*enclosure*) que els conté, i reportar-ne l'estat via web; és a dir, tenen un servidor web *embedded*. Tot això és naturalment propietari.
Així mateix, les cabines de discs durs d'emmagatzemament compartit funcionen similarment. El *front-end* de gestió és un navegador web, que es connecta a la cabina via HTTP per a gestionar-la.
Hi ha una sèrie de programes de de Cisco i altres fabricants que utilitza l'equip de comunicacions per a llur monitorització especialitzada.
El factor comú únic entre la majoria d'equips és la possibilitat de gestionar-los ni que sigui parcialment per SNMP.

D'altra banda, ja fa uns quants anys, quan adquiríem servidors de Dell, Dell "regalava" una versió d'HP Network Node Manager, gestor SNMP que s'integrava molt bé amb els servidors Windows, on s'havia d'instal·lar el programari de Dell, els agents que permetien que HP NNM pogués interrogar per SNMP aquests servidors. Realment el producte era bo, fàcil d'implementar, i funcionava correctament, però la monitorització d'altres elements fora dels servidors Dell amb Windows acabava topant amb limitacions del programari. Un estudi del que podia costar una solució amb HP NNM va fer descartar aquesta solució per diverses qüestions, entre les quals el preu.

Darrerament, amb l'adquisició dels servidors HP Blade BL460c que formen el nucli del sistema virtualitzador de l'Administració X, el Departament d'Informàtica va adquirir també el programari propietari d'Altiris per a vigilar la infraestructura, i per a administrar-la des de molts punts de vista:
- Desplegament automatitzat de nous servidors i de programari sobre aquests
- Monitorització activa i passiva per SNMP (polling i gestió de *traps*), i per agents propis.
- etc
Aquest programari era (és) molt complex de gestionar, la corba d'aprenentatge i posada en marxa incompatible amb el nivell de recursos del Departament d'Informàtica, i per tant tot i tenir-la legalment no s'ha fet servir. Aquest fet ens ha fet esbiaixar a favor de solucions més modulars, més extensibles (quan calgui, no d'entrada i sense que faci falta!), més senzilles, més abordables, obertes i amb ampli suport de la comunitat.
D'altres eines són no només cares d'adquisició, sinó que ho són també de consultoria i de formació del personal, i la corba d'aprenentatge que tenen per part del personal sovint no compensa el benefici que se'n pot obtenir per part d'un CPD mitjà com pugui ser el nostre.

Una altra i interessant alternativa pel que fa a les eines propietàries és el monstre "Microsoft Operations Manager". El producte és bo, amb moltes funcionalitats, propietàriament extensible i

modular, complex d'implementar i d'indubtable interès per a l'Administració X, donat l'elevat nombre de servidors i serveis que s'estan donant en l'entorn Microsoft. Ara bé, pel mateix fet de ser de Microsoft, és orientat als seus servidors i als seus productes. Aquesta alternativa no es descarta en el futur, donat que l'Administració X té contractes avantatjosos amb Microsoft, i el cost econòmic dels seus productes no seria excessiu.

El que es va buscar doncs va ser un sistema de monitorització:
- Open Source
- Senzill i ràpid d'implementar.
- Amb ampli suport de la comunitat, reconegut, i de trajectòria llarga.
- Extensible
- Que fos "realment" extensible, no només "conceptualment" o per disseny.

Com veurem, hi ha diversos sistemes que compleixen en bona part els punts anteriors, però algunes solucions són més fortes en alguns punts que crec crítics, com els de l'extensibilitat i ampli suport de la comunitat d'Internet.
El producte nuclear la implementació del qual proposaré i explicaré en aquest treball és Nagios, que compleix tot allò que volíem, i té alguns valors afegits. Així, per a Nagios es va comprovar la gran quantitat de *scripts*, *plugins*, i afegits diversos que hi havia a la web (wiki, exchange.nagios.org, etc) i l'enorme base d'ús. Es va pensar que si calia monitoritzar "a alt nivell" sistemes com SQL Server, Exchange, DHCP, seria ràpid i relativament senzill de fer-ho copiant/important la gran quantitat de scripts que es poden trobar.
I *last but not least*, i coincident en el temps amb el moment que es consideraven diverses alternatives, Nagios ens va ser recomanat per una consultora externa.

Una recerca de tots els productes de tercers que existeixen per a monitoritzar d'altres productes i entorns Open Source, bé que interessant, queda fora d'aquest projecte i de les intencions i possibilitats immediates del Departament d'Informàtica de l'Administració X, tot i que naturalment no es descarta en absolut.

OpenNMS (http://www.opennms.org/)

És un sistema relativament senzill de muntar, i que té alguns avantatges sobre Nagios. El primer, és que fa un *autodiscovery* dels equips de la xarxa. Per a fer-ho, els fa un *ping* i aquests han de contestar, és a dir, han de tenir ICMP obert i fer ICMP echo reply. Tenir això obert en els sistemes és també necessari per a Nagios, car aquest el primer que fa per saber si un *host* és viu és executar la comanda “check_host_alive” que el que fa és fer *pings* als sistemes. Ara bé, aquesta autodiscovery és molt eficient, perquè probablement fa servir *nmap* per a identificar-ne els ports TCP/UDP oberts i per a registrar tant els *hosts* com els ports TCP i serveis que poden estar-hi associats. Això fa molt poc feixuc el muntatge inicial d'un sistema monitoritzador.

Així doncs, aquesta alternativa sembla molt bona i fàcil, d'entrada. Perquè no es va adoptar?
- La documentació i la comunitat d'ús sembla molt més restringida que per a Nagios.
- L'extensibilitat és més acotada.
- És una eina més *tancada* pel que fa al model de desenvolupament, sense desmereixer-la en absolut però abusant del llenguatge i la comparació seria com comparar els models de desenvolupament de FreeBSD (openNMS) amb Linux (Nagios).

A l'annex A hi ha una petita mostra de les capacitats d'aquesta eina ben interessants.

Zabbix (www.zabbix.com).

Una altra plataforma de codi obert molt interessant, que no es va avaluar en profunditat per manca de temps i pels mateixos motius que s'han esmentat respecte d'OpenNMS pel que fa a l'aparent extensibilitat i suport de la comunitat.

Nagios

L'eina que proposo per a satisfer els requeriments originals és Nagios (http://es.wikipedia.org/wiki/Nagios , http://en.wikipedia.org/wiki/Nagios i www.nagios.org).

Nagios és una eina de codi lliure i obert, amb llicència GPL, i és essencialment i per disseny dues coses:

- és una plataforma, un *framework* que permet de programar de manera intel.ligent múltiples i petits programes monitoritzadors (*plugins*), escrits en qualsevol llenguatge que vulguem, perquè s'executin quan vulguem. Cada programet o *plugin* reporta el seu estat a Nagios, i Nagios vigila aquests valors de retorn.

- és una plataforma per notificar a qui nosaltres diem, per la manera que nosaltres determinementre d'un notable ventall de possibilitats configurables (correu electrònic, SMS, buscapersones,etc) els resultats reportats pels *plugins* que programem que s'executin, o millor dit, el canvi d'estat dels serveis i servidors, l'estat del qual aquests programets ens reporten, des de "correcte" a "avís" o "crític".

Nagios vol fer molt bé i de manera simple els dos punts anteriors, però va ser un dels objectius dels seus programadors el que fos fàcil de subministrar-li dades, i que les dades que ell aplega o genera es puguin subministrar a d'altres programes tercers.

Nagios té una interfície web per a controlar-ne el funcionament, i es configura de manera força laboriosa amb fitxers de text que, d'altra banda, són senzills i comprensibles, i que descriuen amb detall tots els objectes que monitoritzarem i com ho farem.

Nagios es guarda en una base de dades pròpia sobre fitxers dades històriques sobre quan els diversos *plugins* van canviar d'estat. Ara bé, no guarda el detall per exemple del consum de CPU o memòria, mancança que veurem que afecta un dels darrers requeriments del nostre client, i que s'haurà de suplir d'alguna manera. Això sí, pel seu disseny extensible i modular, pot subministrar aquestes dades que deiem que no guarda, però sí que ens va mostrant en temps real com a resultat de la sortida dels *plugins*, a d'altres eines o programaris. Això ens serà molt útil per a poder satisfer el darrer requeriment aparegut el maig de 2011.

Existeix, com en la majoria de projectes de codi lliure, una extensa i rica comunitat d'usuaris i programadors de Nagios, amb un gran nombre de *plugins*, scripts i instruccions per a monitoritzar incomptables sistemes *hardware* i de programari. A nosaltres ens n'interessaran molts, a mesura que calgui incorporar un nou sistema o un nou servei a la monitorització. Donat que la urgència més important per al Departament d'Informàtica era saber esdeveniments obvis, com quan un servei web deixa de respondre, o quan un disc dur és a punt d'omplir-se, aquesta vigilància "fina" s'anirà incorporant al Nagios gradualment, a mesura que el client consideri que val la pena rebre alertes de tota una sèrie de serveis molt extensa. Algunes referències interessants:

- http://library.nagios.com/library/products/nagioscore/manuals/
- http://wiki.nagios.org/index.php/Main_Page
- http://exchange.nagios.org/

Torno a la necessitat de guardar dades històriques que Nagios per defecte no fa. Nagios guarda en fitxers de text (*logfiles*) dades històriques com canvis d'estat, i per la disponibilitat d'un determinat servei o no (OK o "critical"), però no dades com les que he comentat anteriorment. Però un dels elements importants en l'arquitectura modular de Nagios 3.x és l'*event broker. Endollant-se* a aquest *broker,* tota una sèrie d'utilitats de tercer poden extreure les dades que aplega Nagios i processar-les. D'aquestes, vull destadar per motius que sortiran més endavant NDOUtils, un *addon* a Nagios (*http://nagios.sourceforge.net/docs/ndoutils/NDOUtils.pdf*), fet pel mateix Ethan Galstad, autor original del producte, que és dissenyat per a guardar la configuració i les dades dels events en una base de dades com *MySQL* o *postgresql*. Un dels components de NDOUtils és el mòdul per a l'*event broker* de Nagios, NDOMOD.O, que és carregat per Nagios en temps d'execució. Un cop carregat, NDOMOD pot accedir totes les dades i la lògica presents a Nagios; NDOMOD exporta les dades de configuració, així com informació sobre diversos esdeveniments en temps d'execució i els pot exportar per diversos mètodes (TCP o UNIX socket, fitxer). Un altre mòdul de NDOUTils, el dimoni NDO2DB, crea un *socket* UNIX o TCP en arrencar, i espera que els clients se li connectin i guarda les dades que aquests li subministren en una base de dades configurable.

La configuració paquet a paquet de tot això és farragosa. Enlloc de fer-ho tot "manualment", ens podrà servir l'eina *Centreon,* en parlaré aviat.

Nagios pot monitoritzar activament o passivament.

- Els *checks* actius són els més comuns i extesos; són quan Nagios llença el *check* i n'espera el resultat. Són els més indicats quan el procés que s'executa remotament és ràpidament i no requereix un procés que funciona durant un període llarg.

- Els checks passius. Els *checks* actius poden no ser útils quan un *check* triga molta estona, mitja hora per exemple, i no té sentit generar una alerta d'indisponibilitat d'un servei; en aquest cas fóra molt més útil que el *check* corrés tot sol en el servidor que cal monitoritzar, i en acabat aquest reportés l'estat a Nagios.

SNMP

El protocol *SNMP* acrònim *de Simple Network Management Protocol* és universalment implementat en multitud de dispositius connectats en xarxes IP. Hi ha diverses versions, la darrera versió és la 3, que suporta seguretat criptogràfica, amb autenticació de missatges si es vol, però no afegeix res substancialment important pel que fa a la monitorització. Així doncs, si finalment ho implementem, serà per senzillesa i facilitat d'implementació en la versió 2c. Si ho fem servir serà en VLANs controlades per nosaltres i només amb les capacitats de *només lectura*, cosa que ens permet de complir les polítiques de seguretat del Departament d'Informàtica, i que en cas de vulnerabilitat comportaria únicament una divulgació d'informació no crítica de metadades sobre els sistemes.

Nosaltres podríem fer-lo servir de dues maneres, per a satisfer els requeriments que se'ns plantegen.

- Monitorització activa: Podem fer que l'eina que escollim per monitoritzar faci *polling* per SNMP dels dispositius que volem comprovar. És la que voldríem fer servir.

- Monitorització passiva (*checks* passius): podem fer que la nostra eina reculli *traps* de SNMP generats pels dispositius, i que els tradueixi i vehiculi cap a un altre programari –Nagios,si s'escau-- perquè pugui notificar dels esdeveniments que decidim. El que ens caldria fer de manera aproximada per a implementar-ho amb Nagios ho veiem a http://ws.edu.isoc.org/data/2009/1734780084ac614b3a1377/Gestion_de_alarmas_traps_SNMP.ppt. El procediment és laboriós i farragós, i podria deixar-ho per a una extensió del projecte si em quedés amb Nagios com a eina única de monitorització.

Com és d'esperar, la implementació i suport de SNMP sobre Linux és excel.lent, i el nucli d'eines ens el dóna el paquet net-snmp. Aquest paquet du les eines per poder interrogar els equips remots per SNMP. Si el que vulguessim és recollir i processar les *traps* de SNMP, hauriem d'instal.lar *snmptt (*per SNMP Trap Translator). Sigui com sigui, ens caldrà obtenir les MIBs dels dispositius que ens interessin (p.ex. Veure https://www.ac.usc.es/docencia/ASRII/Tema_1html/node13.html). *Googlejant* s'obté àmplia i extensiva informació sobre tots aquests conceptes, un aprofundiment sobre els quals depassa l'abast d'aquest treball.

Centreon

*Centreon (*http://www.centreon.com/Centreon/product-overview.html i http://en.doc.centreon.com/CentreonArchitecture) podria ser considerat com un "supershell" o un *front-end* sobre Nagios. *Centreon* fa servir NDOUtils per a guardar en una instància de MySQL totes les dades que li calen. És també una aplicació/interfície web de configuració d'un Nagios, més tota una sèrie de plugins (que es guarden al mateix directori exactament que els plugins de Nagios (si ho compilem nosaltres, /usr/local/nagios/libexec).

Així mateix, Centreon integra monitorització per SNMP i gestió de traps i MIBS (http://en.doc.centreon.com/CentreonArchitecture#Centreon.27s_SNMP_Support).

Cacti

- És un *front-end* completament en PHP de RRDTool; pot graficar tota mena de dades guardades en una base de dades RRD (Round Robin Database). Té suport per SNMP, per tota la comunitat habituada a fer servir MRTG+RRDTool per monitoritzar entorns per SNMP; de fet és per a MRTG que es va fer servir per primer cop la base de dades RRD.

- NO notifica.

La manca de notificacions fa que descartem aquesta eina, que d'altra banda és molt versàtil i usada.

http://en.wikipedia.org/wiki/Cacti_(software)

http://www.cacti.net/what_is_cacti.php

http://www.espida.co.uk/index.php?option=com_content&view=article&id=25&Itemid=25

3 Anàlisi del sistema

3.1 Definició del sistema

Un sistema central monitoritzador en RedHat Enterprise Linux 5.x, amb Nagios 3.2.2. Aquest vigilarà:
- Sistemes Linux remots per ssh executant els plugins de Nagios localment en els sistemes, i tornant els resultats per SSH al monitor.
- Sistemes Windows mitjançant el *plugin check_nt* del monitor atacant els sistemes Windows pel port de NSClient++, que aquests tindran instal.lat localment.
- Sistemes genèrics per SNMP, si és possible.

Nagios no s'instal.larà des de cap repositori. És preferible baixar-se la darrera versió del producte i dels *plugins* i compilar-ho. Per aquest motiu, caldrà tenir instal.lat el compilador gcc i les *llibreries* de suport, així com totes les *llibreries* requerides per a compilar les opcions de Nagios i els *plugins.*

3.2 Requisits exactes del projecte

- Monitoritzar activament en general l'estat dels sistemes i els serveis: si són vius els servidors i els serveis que ofereixen.
- Notificar les fallades dels darrers als responsables de restablir-los el més aviat possible, per email i per SMS en principi.
- Crear diferents usuaris amb accessos al portal de monitorització, amb capacitat per veure selectivament els servidors dels quals són responsables.
- Fer còpies de seguretat de les dades aplegades pel sistema: els *logs* i les bases de dades.
- Servidor accessible només des d'unes poques adreces IP establertes.
- La seguretat s'ha de tenir present sempre: si és possible no obrir ports TCP/UDP en els servidors si no és necessari, i fer-ho amb la màxima seguretat.

3.3 Definició d'interfícies d'usuari

La interfície d'usuari s'accedirà des d'unes poques adreces IP determinades pels requeriments de seguretat.

La interfície de gestió és una aplicació web:

. http://monitor.administraciox.cat (només accessible des de la VLAN de servidors)

- https://monitor142.administraciox.cat on el servidor té la IP pública accessible si s'obre el *firewall* corporatiu, i la ruta per defecte.

La funcionalitat és la determinada pel producte; no s'hi farà cap extensió. Evidentment, s'adaptarà i adequarà a les necessitats del CPD del Departament d'Informàtica.
- L'ampliació d'última hora del projecte ha afegit una nova interfície de gestió, que acabarà a la llarga substituint l'anterior, i que és hostatjada en un altre servidor diferent de l'anterior.
- http://centreon.administraciox.cat

3.4 Especificació pla de proves

No hi haurà pla de proves. El que es demana és instal·lar un Nagios en un sistema operatiu Linux, i

fer-lo funcionar. El que es farà és monitoritzar per etapes diversos servidors i VLANs, i no es passarà a l'etapa següent fins haver reeixit en la fase prèvia. Els requeriment primordial és de fer una monitorització simple que inclou:
- el servidor és viu
- els discs durs no són plens
- la cpu no és al 100% durant un llarg període de temps

Això fa que les proves es reduiran a comprovar que la monitorització és satisfactòria en un servidor en cada arquitectura Linux i Windows en totes les seves versions de servidor.

Es monitoritzarà
- Primer, el propi servidor *monitor*, en la configuració per defecte.
- Segon, servidors Windows de la mateixa VLAN on hi ha la IP canònica de monitor.administraciox.cat. Això vol dir:
    - Un Windows 2000.
    - Un Windows Server 2003 i 2003 R2 (tots en arquitectura x32).
    - Un Windows Server 2008 (x32) i x64.
    - Un Windows Server 2008 R2 (que només és en x64).

A més a més dels paràmetres per defecte, es monitoritzarà si són engegats diversos serveis.

- Tercer, un servidor Linux en la pròpia VLAN (RHEL 5.x x32). Es monitoritzaran els serveis per defecte de la instal·lació de Nagios, i l'espai lliure de les diferents particions dels discs durs.

4 Disseny del sistema

4.1 Arquitectura

Servidor

Sistema operatiu i distribució Linux

Com ja hem comentat, la versió de sistema operatiu que s'ha decidit per al servidor amb Nagios és Red Hat Enterprise Linux 5. Veurem que per a Centreon, interfície d'alt nivell o *supershell* per a Nagios, el sistema que triarem és una Debian 5 x32.

Configuració de xarxa

Hem optat per posar en el servidor una placa de xarxa ethernet a cada VLAN on hi haurà màquines a configurar. Això ens simplifica el desplegament, però planteja interrogants de seguretat que poden fer-nos revertir en el futur aquesta decisió. Aquesta recança ve pel fet que la colonització per part d'atacants d'aquesta màquina facilitaria molt qualssevol posteriors atacs a tota la infraestructura sense cap restricció, i en certa manera treu sentit a un tallafocs corporatiu.
Una configuració alternativa i més segura, que requereix de més feina per identificar exactament tots els ports de la capa de xarxa involucrats en la monitorització, seria tenir una única interfície i obrir en el tallafocs corporatiu els ports necessaris per a la vigilància i aplegament de dades. En una configuració inicial en aquest supòsit, caldria:

- Posar una única interfície de xarxa a la VLAN200
- Obrir els ports de ssh (TCP 22), SNMP (UDP 161), TCP 12489 (NSClient++) i TCP 5666 (Nclient++ NRPE) de totes les VLANs de destí per ser monitoritzades, amb origen "monitor.administraciox.cat". Si volguessim rebre *traps* SNMP (*checks* passius), hauriem de permetre l'entrada de UDP 162.
- Reconfigurar tots els tallafocs de totes les màquines monitoritzades, per permetre ser atacades pel(s) port(s) que s'escaigui(n) dels quatre anteriors amb origen "monitor.administraciox.cat". Notem que una mateixa màquina pot ser monitoritzada per més d'una via (SNMP i NSClient++, per exemple).

Nagios

Instal.lació

Documentació prèvia i prerequisits.

Per a la instal.lació de Nagios sobre la plataforma escollida, Red Hat Enterprise Linux 5.x, és útil de llegir-se http://www.unixmen.com/linux-tutorials/151-how-to-install-nagios-in-rhel-centos-fedora, així com les instruccions del document "Nagios Install on Redhat EL 4", amb un procediment lleugerament diferent que sobre Red Hat 5 i Nagios 3, les versions actuals. També és interessant http://nagios.sourceforge.net/download/contrib/documentation/misc/Nagios%20Install%20on%20Redhat%20EL%204.doc .

A data de setembre de 2010, també hi havia http://nagios.sourceforge.net/docs/2_0/installing.html, pàgina actualment no existent (juny de 2010).

De tots els paquets que la documentació anterior ens indica que cal instal·lar com a prerequisit, només ens cal instal·lar explícitament *gd* i *gd-devel* (i llurs dependències, que s'instal.laran automàticament). Tota la resta de paquets ja és instal·lada amb la instal·lació de Red Hat EL que he feta, en què he triat "development" i "web server". També caldrà instal·lar *openssl-dev* per poder compilar suport per SSL per als *plugins* de Nagios:

```
# yum install openssl-dev
```

Caldrà baixar-se doncs els paquets de Nagios i els *plugins* i compilar-los, seguint de manera aproximada les instruccions anteriors.

Instal.lació de Nagios.

Creo l'usuari nagios, el grup nagcmd, i faig pertànyer al darrer els usuaris "nagios" i "apache".

```
# /usr/sbin/useradd -m nagios
# passwd nagios
# /usr/sbin/groupadd nagcmd
# /usr/sbin/usermod -a -G nagcmd nagios
# /usr/sbin/usermod -a -G nagcmd apache

# cd /root
# wget http://prdownloads.sourceforge.net/sourceforge/nagios/nagios-3.2.3.tar.gz
# tar -zxvf nagios-3.2.2.tar.gz
# cd nagios-3.2.2
```

Un cop baixats i descomprimits els paquets *nagios-3.2.2.tar.gz* i *nagios-plugins-1.4.15.tar.gz*, cadascun en els seus directoris, configurarem, compilarem i instal.larem tots dos paquets sota el directori "/usr/local" en el directori "nagios".

Per al Nagios:

```
# ./configure --prefix=/usr/local/nagios --with-cgiurl=/nagios/cgi-bin --with-htmurl=/nagios --with-nagios-user=nagios --with-nagios-group=nagios --with-command-group=nagcmd –with-pgsql-xdata
# make all, make install, make install-init, make install-commandmode
# make install-config, make install-webconf
# htpasswd -c /usr/local/nagios/etc/htpasswd.users usuariadministradornagios
```

Per als *plugins:*

```
# cd /root
# wget http://prdownloads.sourceforge.net/sourceforge/nagiosplug/nagios-plugins-1.4.15.tar.gz
# tar -zxvf nagios-plugins-1.4.15.tar.gz
# cd nagios-plugins-1.4.15
# ./configure --enable-perl-modules --with-nagios-user=nagios --with-nagios-group=nagios --with-openssl=/usr/include/openssl –enable-extra-opts
# make
# make install
```

Ara queda tot instal.lat sota "/usr/local/nagios", naturalment amb les excepcions del fitxer de configuració de l'aplicació web i els fitxers d'inicialització del dimoni, sota "/etc/httpd/conf.d" i "/etc/init.d", respectivament.

Configuració.

Nagios es configura mitjançant tot un seguit de fitxers de text que viuen essencialment al directori d'instal.lació i els seus subdirectoris (/usr/local/nagios).

A Nagios hi ha dos tipus de fitxers de configuració, els que contenen directives i els que contenen definicions. Imprescindibles, només hi ha els fitxers "nagios.cfg" i els que defineixen objectes.

A Nagios existeix una estructura jeràrquica d'herències; es poden definir patrons (*templates*) que la resta d'objectes poden referenciar amb la paraula "use". En fer-ho, n'hereten totes les propietats i definicions. Això estalvia molta feina ja que permet definir en un únic punt una configuració que es pot repetir centenars de vegades. Existeix tot un conjunt de regles de precedència pel que fa a les definicions i opcions, ben definit i que cal conéixer.

A /usr/local/nagios/etc:

- **/usr/local/nagios/etc/resource.cfg**

Fitxer amb el *path* als plugins ($USER1$), i amb contrasenyes i usuaris definits per mi per connectar-me a diversos serveis; notablement el client NSClient++ per a Windows amb contrasenya

```
##############################################################################
#
# RESOURCE.CFG - Sample Resource File for Nagios 3.2.1
#
# Last Modified: 09-10-2003
#
# You can define $USERx$ macros in this file, which can in turn be used
# in command definitions in your host config file(s).  $USERx$ macros are
# useful for storing sensitive information such as usernames, passwords,
# etc.  They are also handy for specifying the path to plugins and
# event handlers - if you decide to move the plugins or event handlers to
# a different directory in the future, you can just update one or two
# $USERx$ macros, instead of modifying a lot of command definitions.
#
# The CGIs will not attempt to read the contents of resource files, so
# you can set restrictive permissions (600 or 660) on them.
#
# Nagios supports up to 32 $USERx$ macros ($USER1$ through $USER32$)
#
# Resource files may also be used to store configuration directives for
# external data sources like MySQL...
#
##############################################################################

# Sets $USER1$ to be the path to the plugins
$USER1$=/usr/local/nagios/libexec

# Sets $USER2$ to be the path to event handlers
#$USER2$=/usr/local/nagios/libexec/eventhandlers

# Store some usernames and passwords (hidden from the CGIs)
#$USER3$=someuser
$USER4$=contrasenya_clients_windows
$USER5$=usuari_domini_windows_corporatiu_proves
$USER6$=contrasenya2

######## Final fitxer resource.cfg.
```

- **/usr/local/nagios/etc/cgi.cfg**

Fitxer on diem a Nagios on ha de trobar els principals fitxers de configuració, i com on li configurem el comportament i autenticació i privilegis de l'aplicació web. Els usuaris que aquí hi apareixen s'han de correspondre amb els que hem posat al fitxer "*htpasswd.users*" que comentarem tot seguit.

```
##############################################################
# CGI.CFG - Fitxer de configuració de Nagios
# Last Modified: Gener 2011
##############################################################
# MAIN CONFIGURATION FILE
main_config_file=/usr/local/nagios/etc/nagios.cfg

# PHYSICAL HTML PATH
physical_html_path=/usr/local/nagios/share

# URL HTML PATH
url_html_path=/nagios

# CONTEXT-SENSITIVE HELP
show_context_help=0

# PENDING STATES OPTION
use_pending_states=1

# AUTHENTICATION USAGE
use_authentication=1

# x509 CERT AUTHENTICATION
use_ssl_authentication=0

# DEFAULT USER
#default_user_name=guest

# SYSTEM/PROCESS INFORMATION ACCESS
# Ningu no hi té acces si no l'hi hem donat explicitament
authorized_for_system_information=usuariadministradornagios

# CONFIGURATION INFORMATION ACCESS
# Llista d'usuaris que poden veure-ho tot separats per coma.
# Usuaris "normals" nomes veuen info de config dels que en son
# contactes.
authorized_for_configuration_information=usuariadministradornagios

# SYSTEM/PROCESS COMMAND ACCESS
authorized_for_system_commands=usuariadministradornagios

# GLOBAL HOST/SERVICE VIEW ACCESS
authorized_for_all_services=usuariadministradornagios
authorized_for_all_hosts=usuariadministradornagios

# GLOBAL HOST/SERVICE COMMAND ACCESS
# These two options are comma-delimited lists of all usernames that
authorized_for_all_service_commands=usuariadministradornagios
authorized_for_all_host_commands=usuariadministradornagios

# READ-ONLY USERS
#authorized_for_read_only=user1,user2

# .............etc......línies escapçades per simplicitat .............

# DEFAULT STATUSMAP LAYOUT METHOD
# This option allows you to specify the default layout method
default_statusmap_layout=5

# DEFAULT STATUSWRL LAYOUT METHOD
default_statuswrl_layout=4

# .............etc......línies escapçades per simplicitat .............

# PING SYNTAX
ping_syntax=/bin/ping -n -U -c 5 $HOSTADDRESS$

# REFRESH RATE
# This option allows you to specify the refresh rate in seconds
refresh_rate=90

# ESCAPE HTML TAGS
# This option determines whether HTML tags in host and service
# status output is escaped in the web interface.  If enabled,
# your plugin output will not be able to contain clickable links.
escape_html_tags=1

# SOUND OPTIONS
```

```
# ......... etc.............

# URL TARGET FRAMES
# These options determine the target frames in which notes and
# action URLs will open.
action_url_target=_blank
notes_url_target=_blank

# LOCK AUTHOR NAMES OPTION
#         1 = lock author names (disallow editing)
lock_author_names=1

# ......... etc.............
# ------------------Final Fitxer cgi.cfg -------------------
```

- **/usr/local/nagios/etc/htpasswd.users**

Fitxer local a l'aplicació Nagios. Fitxer pla que emmagatzema parelles d'usuaris/contrasenyes per a l'autenticació bàsica d'usuaris HTTP per a l'aplicació. Es manté el fitxer amb la utilitat *htpasswd*, que instal·la Apache en el servidor web. És en el fitxer de configuració de l'aplicació web de Nagios "/etc/httpd/conf.d/nagios.conf" on li diem la configuració, i en particular la localització d'aquest fitxer:

```
#Fitxer "/etc/httpd/conf.d/nagios.conf"
# SAMPLE CONFIG SNIPPETS FOR APACHE WEB SERVER
# Last Modified: 11-26-2005
#
# This file contains examples of entries that need
# to be incorporated into your Apache web server
# configuration file.  Customize the paths, etc. as
# needed to fit your system.

ScriptAlias /nagios/cgi-bin "/usr/local/nagios/sbin"
<Directory "/usr/local/nagios/sbin">
#  SSLRequireSSL
   Options ExecCGI
   AllowOverride None
   Order allow,deny
   Allow from all
#  Order deny,allow
#  Deny from all
#  Allow from 127.0.0.1
   AuthName "Nagios Access"
   AuthType Basic
   AuthUserFile /usr/local/nagios/etc/htpasswd.users
   Require valid-user
</Directory>

Alias /nagios "/usr/local/nagios/share"
<Directory "/usr/local/nagios/share">
#  SSLRequireSSL
   Options None
   AllowOverride None
   Order allow,deny
   Allow from all
#  Order deny,allow
#  Deny from all
#  Allow from 127.0.0.1
   AuthName "Nagios Access"
   AuthType Basic
   AuthUserFile /usr/local/nagios/etc/htpasswd.users
   Require valid-user
</Directory>
```

Referència: http://httpd.apache.org/docs/2.0/programs/htpasswd.html

Un exemple del contingut podria ser:

```
usuariadministradornagios:contrasenya_encriptada
usuari_comunicacions:contrasenya_encriptada2
usuari_unitatdesenvolupament:contrasenya_encriptada2
```

- **/usr/local/nagios/etc/nagios.cfg**

Aquest fitxer controla tot el funcionament de Nagios. És un fitxer d'unes 1400 línies, extens i molt ben auto-documentat, amb una explicació molt bona de la configuració de tots els seus nombrosos paràmetres de configuració. Heus-ne ací un breu trosset, amb les coses més importants que he modificat, i algun comentari del perquè.

```
##############################################################################
# NAGIOS.CFG - Sample Main Config File for Nagios 3.2.1
##############################################################################
# LOG FILE
log_file=/usr/local/nagios/var/nagios.log

# OBJECT CONFIGURATION FILE(S)
# You can specify individual object config files as shown below:
# Efectivament, hi he posat dos fitxers com "comandes-remotes" i "grups-maquines"
#       el primer defineix les comandes remotes que monitoritzaran per SSH
#       els servidors Linux remots; el segon agrupa els servidors en agrupacions
#       logiques "maques" perque la interficie grafica me les ensenyi juntes.
cfg_file=/usr/local/nagios/etc/objects/commands.cfg
cfg_file=/usr/local/nagios/etc/objects/contacts.cfg
cfg_file=/usr/local/nagios/etc/objects/timeperiods.cfg
cfg_file=/usr/local/nagios/etc/objects/templates.cfg
cfg_file=/usr/local/nagios/etc/objects/comandes-remotes.cfg
cfg_file=/usr/local/nagios/etc/objects/grups-maquines.cfg
#cfg_file=/usr/local/nagios/etc/objects/comandes_afegides_per_nosaltres_quan_vulguem.cfg

#Tot seguit, llistat de tots els fitxers de configuracio de monitoritzacio de cada servidor.
# Hi ha una linia per cadascun dels aproximadament 70 servidors que monitoritzem.
cfg_file=/usr/local/nagios/etc/objects/servidor1.administraciox.cat.cfg
cfg_file=/usr/local/nagios/etc/objects/servidor2.administraciox.cat.cfg
cfg_file=/usr/local/nagios/etc/objects/servidorx.administraciox.cat.cfg

# Definitions for monitoring the local (Linux) host
cfg_file=/usr/local/nagios/etc/objects/localhost.cfg

# Definitions for monitoring a Windows machine
#cfg_file=/usr/local/nagios/etc/objects/windows.cfg
#comentat: no el fem servir

# Definitions for monitoring a router/switch, and printers
#cfg_file=/usr/local/nagios/etc/objects/switch.cfg
#cfg_file=/usr/local/nagios/etc/objects/printer.cfg
#comentat: no monitoritzem switchos ni impressores de moment

# You can also tell Nagios to process all config files (with a .cfg
# extension) in a particular directory by using the cfg_dir
# directive as shown below:
#cfg_dir=/usr/local/nagios/etc/servers
#interessant, no ho fem servir de moment, aniria bé
#       per organitzar el directori "/usr/local/nagios/etc/objects"
#       on ara mateix hi ha barrejats els fitxers de configuració de cada servidor
#       (en són uns 70) amb els fitxers de definició de comandes.

#com a mostra d'alguns pocs dels parametres de configuracio de Nagios he deixat
#       les linies seguents
# OBJECT CACHE FILE
object_cache_file=/usr/local/nagios/var/objects.cache
# PRE-CACHED OBJECT FILE
precached_object_file=/usr/local/nagios/var/objects.precache
# RESOURCE FILE
resource_file=/usr/local/nagios/etc/resource.cfg
# STATUS FILE
status_file=/usr/local/nagios/var/status.dat
```

```
# STATUS FILE UPDATE INTERVAL
status_update_interval=10
# NAGIOS USER
nagios_user=nagios
# NAGIOS GROUP
nagios_group=nagios
# EXTERNAL COMMAND OPTION
check_external_commands=1
# EXTERNAL COMMAND CHECK INTERVAL
#command_check_interval=15s
command_check_interval=-1
# EXTERNAL COMMAND FILE
command_file=/usr/local/nagios/var/rw/nagios.cmd
# EXTERNAL COMMAND BUFFER SLOTS
external_command_buffer_slots=4096
# LOCK FILE
lock_file=/usr/local/nagios/var/nagios.lock
# TEMP FILE
temp_file=/usr/local/nagios/var/nagios.tmp
# TEMP PATH
temp_path=/tmp

# EVENT BROKER OPTIONS
# Controls what (if any) data gets sent to the event broker.
# Values:  0      = Broker nothing
#         -1      = Broker everything
#         <other> = See documentation
event_broker_options=-1

# EVENT BROKER MODULE(S)
# This directive is used to specify an event broker module that should
# by loaded by Nagios at startup.  Use multiple directives if you want
#broker_module=/somewhere/module1.o
#broker_module=/somewhere/module2.o arg1 arg2=3 debug=0
# En la maquina "monitor" no es carrega cap modul, tot es comentat. Aixo es
#       a diferencia de la maquina on hem instal.lat centreon i nagios; en aquest cas
#       i com expliquem mes endavant, es carreguen els moduls de ndoutils, on la linia no es
#       comentada i es la seguent:
#       broker_module=/usr/local/nagios/bin/ndomod.o config_file=/usr/local/nagios/etc/ndomod.cfg
```

En el fitxer anterior, notem que la part d'"event broker options/modules" estem dient que tots els esdeveniments s'han de tenir en compte, i passar-los tots a qui vulgui aplegar-los. Així mateix, en el cas del servidor amb Nagios no es carrega cap mòdul, però –ja en parlaré més tard-- en el cas del servidor amb Centreon, es carrega el mòdul de les NDOUtils.

A /usr/local/nagios/etc/keys:

- Les claus públiques de tots els servidors que monitoritzarem.

A /usr/local/nagios/etc/objects/

- Els fitxers de configuració de monitorització per a tots i cadascun dels servidors que vigilarem remotament "nomservidor.cfg".
- templates.cfg

En aquest fitxer creo "patrons" o *templates*. Notem que puc els patrons poden heretar dels patrons, i dins d'un patró "fill" puc tornar a definir valors heretats de patrons "pare"; en aquest cas tenen precedència els valors que s'han definit per darrer cop, els del "fill". Les definicions de patrons d'aquest fitxer es fan servir pertot arreu, ja que aquí definim "hoste genèric", "hoste windows", "hoste linux", "servei generic", etc. És molt pràctic centralitzar les definicions d'objectes en objectes que els agrupin, així aquí a "templates" hi concentrem això, els "patrons".

El fitxer "templates" nostre:

```
###############################################################################
###############################################################################
# TEMPLATES.CFG - SAMPLE OBJECT TEMPLATES
```

```
################################################################################

################################################################################
# CONTACT TEMPLATES
################################################################################
# Generic contact definition template - This is NOT a real contact, just a template!
define contact{
  name             generic-contact          ; The name of this contact template
  service_notification_period     24x7      ; service notifications can be sent anytime
  host_notification_period        24x7      ; host notifications can be sent anytime
  service_notification_options    w,u,c,r,f,s
  host_notification_options       d,u,r,f,s
  service_notification_commands   notify-service-by-email  ; send service notifications via email
  host_notification_commands      notify-host-by-email     ; send host notifications via email
  register   0  ; DONT REGISTER THIS DEFINITION - ITS NOT A REAL CONTACT, JUST A TEMPLATE!
  }

#Definit nou contacte per enviar per SMS alertes. Josep 25/08/10
# Notem que es fa servir també un nou periode de temps que hem definit a timeperiods.cfg.
define contact{
       name                            sms-contact
       service_notification_period     24x7except00:00-07:00am
       host_notification_period        24x7except00:00-07:00am
       service_notification_options    w,u,c,r,f,s
       host_notification_options       d,u,r,f,s
       service_notification_commands   notify-service-by-sms
```

;;Aquí em caldrà haver definit la comanda “notify-service-by-sms”.

```
       host_notification_commands      notify-host-by-sms
       register                        0
       }

################################################################################
# HOST TEMPLATES
################################################################################

# Definicio d'un patro per a un host generic – NO es un host real, nomes un patro
define host{
       name                            generic-host    ; The name of this host template
       notifications_enabled           1           ; Host notifications are enabled
       event_handler_enabled           1           ; Host event handler is enabled
       flap_detection_enabled          1           ; Flap detection is enabled
       failure_prediction_enabled      1           ; Failure prediction is enabled
       process_perf_data               1           ; Process performance data
       retain_status_information       1     ; Retain status information across program restarts
       retain_nonstatus_information    1     ; Retain non-status information across program restarts
       notification_period           24x7          ; Send host notifications at any time
       register                        0
       }
```

Primer exemple d'herència, tots dos objectes *host* linux-server com windows-server heretaran de l'objecte “generic-host” definit tot just a sobre.

```
# Linux host definition template
define host{
       name                  linux-server
       use                   generic-host   ; Hereta del patro “generic-host”
       check_period          24x7   ; Per defecte, 24 hores al dia
       check_interval        5      ; Actively check the host every 5 minutes
       retry_interval        1      ; Schedule host check retries at 1 minute intervals
       max_check_attempts    10     ; Check each Linux host 10 times (max)
       check_command         check-host-alive ; Default command to check Linux hosts
       notification_period   workhours
; Aquí sobreescric la variable “notification_period” que hereto del patró “generic-host”.
;
       notification_interval 120    ; Resend notifications every 2 hours (120 minutes)
       notification_options  d,u,r  ; Only send notifications for specific host states
       contact_groups        admins ; Notifications get sent to the admins by default
       hostgroups            linux-servers
       register              0
       }
```

 # Definició del patro de host Windows, que hereta de “generic-host”.

```
define host{
```

```
        name                   windows-server
        use                    generic-host
        check_period           24x7
        check_interval         5
        retry_interval         1
        max_check_attempts     10
        check_command          check-host-alive
        notification_period    24x7
        notification_interval 30
        notification_options  d,r     ; Only send notifications for specific host states
        contact_groups admins         ; Notifications get sent to the admins by default
        hostgroups     windows-servers ; Host groups that Windows servers should be a member of
        register                      0
}
```

Tots els nostres sistemes Windows i Linux podran heretar les propietats bàsiques d'aquests dos patrons de host, amb la comanda *use*.

```
###############################################################################
# SERVICE TEMPLATES
###############################################################################
# Generic service definition template - This is NOT a real service, just a template!
define service{
        name                            generic-service     ; The 'name' of this service template
        active_checks_enabled           1    ; Active service checks are enabled
        passive_checks_enabled          1    ; Passive service checks are enabled/accepted
        parallelize_check               1
        obsess_over_service             1     ; We should obsess over this service (if necessary)
        check_freshness                 0     ; Default is to NOT check service 'freshness'
        notifications_enabled           1     ; Service notifications are enabled
        event_handler_enabled           1     ; Service event handler is enabled
        flap_detection_enabled          1     ; Flap detection is enabled
        failure_prediction_enabled      1     ; Failure prediction is enabled
        process_perf_data               1     ; Process performance data
        retain_status_information       1     ; Retain status information across program restarts
        retain_nonstatus_information    1     ; Retain non-status information across program restarts
        is_volatile                     0     ; The service is not volatile
        check_period                    24x7  ; The service can be checked at any time of the day
        max_check_attempts              3
        normal_check_interval         10  ;Check the service every 10 minutes under normal conditions
        retry_check_interval           2      ; Re-check the service every two minutes until
                ;a hard state can be determined
        contact_groups admins ; Notifications get sent out to everyone in the 'admins' group
        notification_options          w,u,c,r  ; Send notifications about warning, unknown,
                ;critical, and recovery events
        notification_interval           60    ; Re-notify about service problems every hour
        notification_period             24x7  ; Notifications can be sent out at any time
        register                        0
        }

# Local service definition template
define service{
        name   local-service         ; The name of this service template
        use    generic-service       ; Inherit default values from the generic-service definition
        max_check_attempts       4
        normal_check_interval    5   ; Check the service every 5 minutes under normal conditions
        retry_check_interval 1; Re-check the service every minute till a hard state can be determined
        register                 0
}

#Josep 15-9-2010 - Definicio d,un patro per a monitoritzar serveis remots
define service{
        name   remote-service        ; The name of this service template
        use    generic-service       ; Inherit default values from the generic-service definition
        max_check_attempts              4
        normal_check_interval           5
        retry_check_interval            1
        register                        0
}
```

- timeperiods.cfg

En aquest fitxer es defineixen els períodes de temps que els diferents objectes de configuració poden fer servir. S'especifiquen blocs de temps, com ara “de dilluns a divendres de 9:00 del matí a 6:00 de la tarda”.

Un fragment d'aquest fitxer:

```
###############################################################################
# TIME PERIODS
###############################################################################
# This defines a timeperiod where all times are valid for checks,
# notifications, etc.  The classic "24x7" support nightmare. :-)
define timeperiod{
        timeperiod_name 24x7
        alias           24 hores al dia, 7 dies de la setmana
        sunday          00:00-24:00
        monday          00:00-24:00
        tuesday         00:00-24:00
        wednesday       00:00-24:00
        thursday        00:00-24:00
        friday          00:00-24:00
        saturday        00:00-24:00
        }

# Definicio del periode de temps “hores de treball” o “workhours”
define timeperiod{
        timeperiod_name workhours
        alias           Horari Laboral Departament Informatica
        monday          08:00-17:45
        tuesday         08:00-17:45
        wednesday       08:00-17:45
        thursday        08:00-17:45
        friday          08:00-17:45
        }
#Aquí hi ha per exemple les hores de treball usuals al Departament d'Informàtica
###############################################################################
```

- windows.cfg

Fitxer que copiarem a cada fitxer de configuració de *host* windows que configurem (és a dir, per a cada Windows, crearem un fitxer com aquest substituint *host_name* pel nom de *host* real que volem monitoritzar. Seria un fitxer d'exemple.

- contacts.cfg

Aquí hi definim els contactes i els grups de contacte. Aquests darrers són els que podrem fer servir com a destinacions de contacte dels objectes *hostgroup* o *host.*

Ara mateix, hi ha un únic destinatari de totes les alertes. Aquestes s'envien per email a l'usuari de suport del Departament d'Informàtica, secció sistemes.

Això és així perquè tots els servidors *hereten* dels objectes *host windows-server i linux-server,* definits a *templates.cfg,* que hem vist que tenien com a *contact-groups* al grup *admins.*

```
###############################################################################
#
# CONTACTS
#
###############################################################################

# Just one contact defined by default - the Nagios admin (that's you)
# This contact definition inherits a lot of default values from the 'generic-contact'
# template which is defined elsewhere.
```

```
define contact{
 contact_name  nagiosadmin           ; Short name of user
 use    generic-contact  ; Inherit default values from generic-contact template (defined above)
 alias Nagios Admin  ; Full name of user
 email usuari_de_suport_de_sistemes@administraciox.cat ;
 }

define contactgroup{
 contactgroup_name   admins
 alias               Nagios Administrators
 members             nagiosadmin
 }
```

Hi ha d'altres contactes definits, però no se'ls envia encara cap notificació, donat que en la immensa majoria els responsables que un servei funcioni són la secció de sistemes, i és habitualment la intervenció d'aquesta secció la que resol el 99% dels problemes d'indisponibilitat.

- commands.cfg

  Fitxer important, perquè és on definim totes les comandes que farem servir des de Nagios. Aquestes comandes tenen una sintaxi similar a la de tots els objectes de *Nagios*, i amb el paquet ja vénen les necessàries pera monitoritzar el propi sistema, la qual cosa no és particularment útil, i unes quantes comandes que permeten monitoritzar diversos serveis locals i remots.

  Les dues comandes importants que hi he definit són les que permetran enviar alertes per SMS, si s'escau, i les he batejades “notify-host-by-sms” i “notify-service-by-sms”, tot seguint les instruccions del proveïdor “www.justvoip.com” (n'hi ha d'altres que es podrien fer servir).

```
# 'notify-host-by-email' command definition
define command{
        command_name    notify-host-by-email
        command_line    /usr/bin/printf "%b" "***** Nagios *****\n\nNotification Type:
$NOTIFICATIONTYPE$\nHost: $HOSTNAME$\nState: $HOSTSTATE$\nAddress: $HOSTADDRESS$\nInfo:
$HOSTOUTPUT$\n\nDate/Time: $LONGDATETIME$\n" | /bin/mail -s "** $NOTIFICATIONTYPE$ Host Alert:
$HOSTNAME$ is $HOSTSTATE$ **" $CONTACTEMAIL$
        }

# 'notify-service-by-email' command definition
define command{
        command_name    notify-service-by-email
        command_line    /usr/bin/printf "%b" "***** Nagios *****\n\nNotification Type:
$NOTIFICATIONTYPE$\n\nService: $SERVICEDESC$\nHost: $HOSTALIAS$\nAddress: $HOSTADDRESS$\nState:
$SERVICESTATE$\n\nDate/Time: $LONGDATETIME$\n\nAdditional Info:\n\n$SERVICEOUTPUT$" | /bin/mail -s
"** $NOTIFICATIONTYPE$ Service Alert: $HOSTALIAS$/$SERVICEDESC$ is $SERVICESTATE$ **" $CONTACTEMAIL$
        }

#Dos comandes posteriors afegides per poder enviar alertes per SMS.
# Josep 25/08/10
#
define command{
        command_name    notify-host-by-sms
        command_line    /usr/bin/wget --no-check-certificate
"https://www.justvoip.com/myaccount/sendsms.php?
username=el_nostre_usuari&password=la_nostra_contrasenya&from=nagiosAdmX&to=+34777777777&text='$NOTI
FICATIONTYPE$: $HOSTNAME$ is $HOSTSTATE$ ($OUTPUT$)' " -O /tmp/SMS.resultat.xml
        }
define command{
        command_name    notify-service-by-sms
        command_line    /usr/bin/wget --no-check-certificate
"https://www.justvoip.com/myaccount/sendsms.php?
username=el_nostre_usuari&password=la_nostra_contrasenya&from=MonitorAdmX&to=+34777777777&text='$NOT
IFICATIONTYPE$: $HOSTNAME$ is $HOSTSTATE$ ($OUTPUT$)' " -O /tmp/SMS.resultat.xml
        }
```

on he substituït les dades com l'usuari, la contrasenya i el telèfon de contacte SMS per dades fictícies.

He volgut aplegar totes les comandes definides per mi que permetran comprovar per SSH servidors Linux remots en un altre fitxer, *comandes-remotes.cfg:*

- comandes-remotes.cfg

Dins d'aquest fitxer defineixo totes les comandes o *checks* que fem servir per a tots els servidors Linux, donat que els fem per SSH. Per a cada "check" bàsic que fem, hem de crear una comanda que faci el mateix que el *plugin* de Nagios que hi ha instal.lat localment a cada servidor remot. Així, per a mirar com està la càrrega, definiré al servidor Nagios la comanda "check_remote_load" que invocarà el plugin de Nagios local "check_load", i així amb tots. Per simplificar l'ús i la legibilitat, faré que totes les comandes tinguin el mateix nom que el *plugin* que vull executar remotament, amb el prefix "check_remote_".

Fixem-nos que les macros (que podem imaginar com variables d'entorn pròpies de Nagios) $HOSTNAME, $HOSTADDRESS, $ARG1, $ARG2, $ARG3, $ARG4 es resolen respectivament al nom del servidor, l'adreça IP, port TCP que ataquem (que normalment serà el port 22 de SSH però no té perquè), llindar de *warning*, llindar crític, contrasenya "$ARG4" (que és definida a "resources.cfg" de què hem parlat anteriorment).

```
###############################################################################
#
# REMOTE COMMANDS
#
###############################################################################
# copiat de http://wiki.nagios.org/index.php/Howtos:checkbyssh_RedHat
# Josep 15-9-2010

# 'check_remote_disk' command definition
define command{
        command_name    check_remote_disk
        command_line    $USER1$/check_by_ssh -p $ARG1$ -l nagios -i /usr/local/nagios/etc/keys/
$HOSTNAME$ -H $HOSTADDRESS$ -C '/usr/local/nagios/libexec/check_disk -w $ARG2$ -c $ARG3$ -p $ARG4$'
        }

# 'check_remote_users' command definition
define command{
        command_name    check_remote_users
        command_line    $USER1$/check_by_ssh -p $ARG1$ -l nagios -i /usr/local/nagios/etc/keys/
$HOSTNAME$ -H $HOSTADDRESS$ -C '/usr/local/nagios/libexec/check_users -w $ARG2$ -c $ARG3$'
        }

# 'check_remote_load' command definition
define command{
        command_name    check_remote_load
        command_line    $USER1$/check_by_ssh -p $ARG1$ -l nagios -i /usr/local/nagios/etc/keys/
$HOSTNAME$ -H $HOSTADDRESS$ -C '/usr/local/nagios/libexec/check_load -w $ARG2$ -c $ARG3$'
        }

# 'check_remote_procs' command definition
define command{
        command_name    check_remote_procs
        command_line    $USER1$/check_by_ssh -p $ARG1$ -l nagios -i /usr/local/nagios/etc/keys/
$HOSTNAME$ -H $HOSTADDRESS$ -C '/usr/local/nagios/libexec/check_procs -w $ARG2$ -c $ARG3$ -s $ARG4$'
        }

# 'check_remote_swap' command definition
define command{
        command_name    check_remote_swap
        command_line    $USER1$/check_by_ssh -p $ARG1$ -l nagios -i /usr/local/nagios/etc/keys/
$HOSTNAME$ -H $HOSTADDRESS$ -C '/usr/local/nagios/libexec/check_swap -w $ARG2$ -c $ARG3$'
        }

# 'check_remote_ldap' command definition
define command{
        command_name    check_remote_ldap
        command_line    $USER1$/check_by_ssh -p $ARG1$ -l nagios -i /usr/local/nagios/etc/keys/
$HOSTNAME$ -H $HOSTADDRESS$ -C '/usr/local/nagios/libexec/check_ldap -H $HOSTNAME$ -b $ARG2$'
        }
```

- grups-maquines.cfg

Aquí agrupo les màquines per grups lògics o funcionals, a discreció. Això fa que la interfície gràfica

m'ho interpreti i me les ensenyi juntes.

```
###############################################################################
# HOST GROUP DEFINITION
###############################################################################
define hostgroup{
        hostgroup_name  exchange-servers ; The name of the hostgroup
        alias           Exchange Servers ; Long name of the group
        members        exsrv1,exsrv2
        }
define hostgroup{
        hostgroup_name  sql-server2000
        alias           SQL Server 2000
        members       sql2k1,sql2k2,sql2k3
        }
```

Un paràmetre que no faig servir però que podria ser útil, en el cas d'un redisseny del sistema de notificacions, seria contact-groups; ens permetria d'enviar notificacions selectivament en funció de l'agrupació de hosts.

- D'altres fitxers

On vulgui, puc definir els meus fitxers de configuració, mentre els carregui des de "nagios.cfg" per exemple.

A la URL http://nagioswiki.com/wiki/index.php/Checking_NT_Services_with_Nagios hi ha un fitxer de comandes definit molt interessant, que simplifica la feina de vigilar tot un munt de serveis i de comptadors relevants en entorn Windows. Entre d'altres, vigila servidors de DHCP, DNS, errors de *logon,* de SMB (molt relevants per a servidors de fitxers), memòria, fitxers de paginació; diversos paràmetres relevants d'un servidor Exchange, de servidors de SQL Server, etc.

Com es pot veure doncs, hi ha una gran quantitat de material i documentació per a monitoritzar el que convingui, i el document tot just citat mostra com n'és de potent l'opció "COUNTER" del *plugin check_nt,* que veurem tot seguit.

Del que es tracta llavors ja no és tant del "com", sino del què. I el què requereix una acurada planificació i coneixement de l'entorn Windows i dels seus paràmetres i comportaments "normals" i en estat estable, sobretot per a poder posar els paràmetres de *warning* i *critical* corresponents i amb significat. Una mala configuració del sistema en aquest punt pot arribar a convertir un monitor per Nagios en un generador de *spam* ineficient i molest.

```
#taken from CentOS 4.x
#-------------------------------
# checks_windows.cfg
# Comandes molt interessants, per a us futur. Caldra adaptar els noms de macros per al nostre entorn (veure resources.cfg).
# Josep Juny 2011
#-------------------------------
define command{
        command_name    check_nt_disk
        command_line    $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v USEDDISKSPACE -l $ARG1$ -w $ARG2$ -c $ARG3$
        }
define command{
        command_name    check_nt_cpuload
        command_line    $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v CPULOAD -l $ARG1$
        }
define command{
        command_name    check_nt_uptime
        command_line    $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v UPTIME
        }
define command{
        command_name    check_nt_clientversion
        command_line    $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v CLIENTVERSION
        }
define command{
        command_name    check_nt_process
        command_line    $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v PROCSTATE -l $ARG1$
        }
define command{
        command_name    check_nt_service
        command_line    $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v SERVICESTATE -d SHOWALL -l $ARG1$
```

```
        }
define command{
        command_name    check_nt_memuse
        command_line    $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v MEMUSE -w $ARG1$ -c $ARG2$
        }
define command{
        command_name    check_nt_pagingfile
        command_line    $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v COUNTER -l "\\Paging File(_Total)\\% Usage","Paging
File usage is %.2f %%" -w $ARG1$ -c $ARG2$
        }
#DHCP - Queue Length due to conflicts
define command{
       command_name     check_nt_DHCP_queue_length
       command_line     $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v COUNTER -l "\\DHCP Server\\Conflict Check Queue
Length","Waiting in DHCP Queue due to Conflict is %.f" -w 2 -c 5
        }
#DHCP - Active Queue
define command{
       command_name     check_nt_DHCP_active_queue_length
       command_line     $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v COUNTER -l "\\DHCP Server\\Active Queue
Length","Waiting in DHCP Queue for Normal Processing is %.f" -w 15 -c 30
        }
#DHCP - Average Response Time
define command{
       command_name     check_nt_DHCP_average_response_time
       command_line     $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v COUNTER  -l "\\DHCP Server\\Milliseconds per packet
(Avg)","Average DHCP Server Response in is %.f" -w 70 -c 250
        }

#DNS - Recursive Query Failures
define command{
       command_name     check_nt_DNS_recursive_query_failures
       command_line     $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v COUNTER  -l "\\DNS\\Recursive Query
Failure/sec","DNS Recursive Queries are failing at %.f per second" -w 5 -c 80
        }
#DNS - Recursive Query Timeouts
define command{
       command_name     check_nt_DNS_recursive_query_timeouts
       command_line     $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v COUNTER  -l "\\DNS\\Recursive Query
TimeOut/sec","DNS Recursive Queries are failing because Timed Out at %.f per second" -w 5 -c 80
        }
#DNS - Secure Update Failures
define command{
       command_name     check_nt_DNS_secure_update_failures
       command_line     $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v COUNTER -l "\\DNS\\Secure Update Failure","DNS
Secure Update Failures since last Service Restart is %.f" -w 1 -c 15
        }
#DNS - Total Queries Received per Second
define command{
       command_name     check_nt_DNS_total_queries_per_sec
       command_line     $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v COUNTER  -l "\\WINS Server\\Failed
Queries/sec","Total Queries received per second is %.f" -w 3 -c 5
        }
#Logon Errors
define command{
       command_name     check_nt_logon_errors
       command_line     $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v COUNTER -l "\\Server\\Errors Logon","Logon Errors
since last reboot is %.f" -w 50 -c 150
        }
#SMB General System Errors
define command{
       command_name     check_nt_SMB_general_system_errors
       command_line     $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v COUNTER -l "\\Server\\Errors System","SMB Errors
due to Server problems is %.f" -w 2 -c 20
        }
#SMB Blocking Requests Rejected
define command{
       command_name     check_nt_SMB_blocking_requests_rejected
       command_line     $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v COUNTER -l "\\Server\\Blocking Requests
Rejected","SMB Blockiing requests rejected due to insufficient free resources is %.f Server Parameters need adjusting" -w 10 -c 100
        }
#CPU Load over 10 Min, 60 Min, 24 hrs
define command{
       command_name     check_nt_cpu_avg
       command_line     $USER1$/check_nt -H $HOSTADDRESS$ -v CPULOAD -l 10,60,95,60,60,95,1440,60,95
        }
#Memory Pool Nonpaged Peak
define command{
       command_name     check_nt_memory_pool_nonpaged_peak
       command_line     $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v COUNTER  -l "\\Server\\Pool Nonpaged Peak","Maxium
number of bytes of nonpaged pool which should be same as installed physical memory is %.f"
        }
#Memory Pool Paged Failures
define command{
       command_name     check_nt_memory_pool_paged_failures
       command_line     $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v COUNTER -l "\\Server\\Pool Paged Failures","Number
of times allocation from the page pool have failed is %.f Physical RAM or paging file too small" -w 2 -c 50
        }
#Paging File Usage
define command{
       command_name     check_nt_paging_file_useage
       command_line     $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v COUNTER  -l "\\Paging File(_Total)\\%
Usage","Paging file usage is %.2f %%" -w 30 -c 60
        }
#Paging File Peak Usage
define command{
       command_name     check_nt_paging_file_peak_usage
       command_line     $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v COUNTER  -l "\\Paging File(_Total)\\% Usage
Peak","Paging file Peak usage is %.2f %%" -w 80 -c 90
        }
#System PTEs with 3GB switch

define command{
       command_name     check_nt_system_PTE_with_3GB
       command_line     $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v COUNTER -l "\\Memory\\Free System Page Table
Entries","Number of Page Table Entries not being used is %.f Thresholds set for testing /3GB switch on or off" -w 8000 -c 5000
        }
```

```
#Registry Quota in Use (percent)
define command{
       command_name     check_nt_registry_quota_in_use
       command_line     $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v COUNTER -l "\\System\\% Registry Quota In
Use","Percent Quota in use is %.2f %%" -w 60 -c 85
        }
#Server Work Queue Length
define command{
       command_name     check_nt_server_work_queues
       command_line     $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v COUNTER -l "\\Server Work Queues(0)\\Queue
Length","Current work queue which is an indication of Processing Load is %.f " -w 4 -c 7
        }
#Disk Usage
define command{
       command_name     check_nt_disk_usage
       command_line     $USER1$/check_nt -H $HOSTNAME$ -v USEDDISKSPACE -l c -w 80 -c 90
        }
#Disk Queue Length
define command{
       command_name     check_nt_queue_length
       command_line     $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v COUNTER -l "\\PhysicalDisk(_Total)\\Avg. Disk Queue
Length","Average number of both read and write requests queued is %.2f Consider a faster disk array" -w 1 -c 5
        }
#Pool Nonpaged Bytes no3GB switch
define command{
       command_name     check_nt_pool_nonpaged_bytes_no3GB_switch
       command_line     $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v COUNTER -l "\\Memory\\Pool Nonpaged Bytes","Pool
Nonpaged Bytes is %.f  Thresholds set for testing /3GB switch off" -w 200000000 -c
        }
#Pool Nonpaged Bytes 3GB switch
define command{
       command_name     check_nt_pool_nonpaged_bytes_3GB_switch
       command_line     $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v COUNTER -l "\\Memory\\Pool Nonpaged Bytes","Pool
Nonpaged Bytes is %.f Thresholds set for testing /3GB switch on" -w 100000000 -c 110000000
        }
#Pool Paged Bytes no3GB switch
define command{
       command_name     check_nt_pool_paged_bytes_no3GB_switch
       command_line     $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v COUNTER -l "\\Memory\\Pool Paged Bytes","Pool Paged
Bytes is %.f Thresholds set for testing /3GB switch off" -w 300000000 -c 320000000
        }
#Pool Paged Bytes 3GB switch
define command{
       command_name     check_nt_pool_paged_bytes_3GB_switch
       command_line     $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v COUNTER -l "\\Memory\\Pool Paged Bytes","Pool Paged
Bytes is %.f  Thresholds set for testing /3GB switch on" -w 200000000 -c 220000000
        }
#Printer not ready errors since last restart
define command{
       command_name     check_nt_printer_not_ready
       command_line     $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v COUNTER -l "\\Print Queue(_Total)\\Out of Paper
Errors","Out of Paper Printer Errors since last Service restart is %.f" -w 1 -c 3
        }
#Printer Out of Paper errors since last restart
define command{
       command_name     check_nt_printer_out_of_paper_errors
       command_line     $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v COUNTER -l "\\Print Queue(_Total)\\Out of Paper
Errors","Out of Paper Printer Errors since last Service restart is %.f" -w 1 -c 3
        }
#SMTP Local Delivery Queue
define command{
       command_name     check_nt_smtp_local_delivery
       command_line     $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v COUNTER -l "\\SMTP Server(_Total)\\Local Queue
Length","Number of Messages waiting in queue for Local Recipients is %.f" -w 5 -c 15
        }
#SMTP Remote Delivery Queue
define command{
       command_name     check_nt_smtp_remote_delivery_queue
       command_line     $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v COUNTER -l "\\SMTP Server(_Total)\\Remote Queue
Length","Number of Messages waiting in queue for Remote Recipients is %.f" -w 25 -c 50
        }
#Exchange Active User Count
define command{
       command_name     check_nt_exchange_active_user_count
       command_line     $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v COUNTER -l "\\MSExchangeIS\\Active User Count" -w 4
-c 10
        }
#Exchange Connection Count
define command{
       command_name     check_nt_exchange_connection_count
       command_line     $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v COUNTER -l "\\MSExchangeIS\\Connection Count" -w
100 -c 250
        }
#Exchange Delivery Time
define command{
       command_name     check_nt_exchange_delivery_time
       command_line     $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v COUNTER -l "\\MSExchangeIS(_Average Delivery
Time)\\%%Usage","Average Delivery Time is %.2f%%" -w 2 -c 10
        }
#Exchange Maximum Users
define command{
       command_name     check_nt_exchange_maximum_users
       command_line     $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v COUNTER -l "\\MSExchangeIS\\Connection Count" -w
100 -c 250
        }
#Exchange Messages Delivered/Minute
define command{
       command_name     check_nt_exchange_messages_delivered_per_min
       command_line     $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v COUNTER -l  "MSExchangeIS Mailbox(_Total)\\Messages
Delivered/min" -w 25 -c 120
        }
#Exchange Messages Submitted/Minute
define command{
       command_name     check_nt_exchange_messages_submitted_per_min
       command_line     $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v COUNTER -l  "\\MSExchangeIS
Mailbox(_Total)\\Messages Submitted/min" -w 5 -c 35
        }
```

```
#Exchange Receive Queue
define command{
       command_name     check_nt_exchange_receive_queue
       command_line     $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v COUNTER -l "\\MSExchangeIS Mailbox(_Receive Queue
Size)\\%%Usage","Queue Length is %.2f" -w 3 -c 15
        }
#Exchange Send Queue
define command{
       command_name     check_nt_exchange_send_queue
       command_line     $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v COUNTER -l  "\\MSExchangeIS(_Send Queue Size)\\%
%Usage","Queue Length is $.2f%%" -w 3 -c 15
        }
#SQL Database Data Files Size (Total)
define command{
       command_name     check_nt_sql_database_files_size
       command_line     $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v COUNTER -l "\\SQLServer:Databases(_Total)\\Data
File(s) Size (KB)","SQL Server Databases Datafile size total is %.f" -w 5 -c 30
        }
#SQL Database Log Files Size (Total)
define command{
       command_name     check_nt_sql_database_log_files_size
       command_line     $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v COUNTER -l  "\\SQLServer:Databases(_Total)\\Log
File(s) Size (KB)","SQL Server Databases Logfile size total %.f" -w 10 -c 100
        }
#SQL Database Data File Size (Individual)
define command{
       command_name     check_nt_sql_database_data_file_size_individual
       command_line     $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v COUNTER -l
"\\MSSQL$SBSMONITORING:Databases(_Total)\\Data File(s) Size (KB)","SQL Server Databases Datafile size total is %.f" -w 5 -c 30
        }
#SQL Database Log File Size (Individual)
define command{
       command_name     check_nt_sql_database_log_file_size_individual
       command_line     $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v COUNTER -l
"\\MSSQL$SBSMONITORING:Databases(_Total)\\Log File(s) Size (KB)","SQL Server Databases Logfile size total %.f" -w 10 -c 100
        }
#SQL Server Service
define command{
       command_name     check_nt_sql_server_service
       command_line     $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v SERVICESTATE -l mssqlserver
        }
#SQL Database Size
define command{
       command_name     check_nt_sql_database_size
       command_line     $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v COUNTER -l "\\SQLServerDatabases(_Data Files() Size
(KB)\\%%Usage","Databases Size is $.2f%%" -w 30000 -c 70000
        }
#SQL Server Deadlocks/second
define command{
       command_name     check_nt_sql_server_deadlocks_per_sec
       command_line     $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v COUNTER -l "\\SQLServer:Locks(_Total)\\Number of
Deadlocks/sec","SQL Server Deadlocks per second total %.f" -w 1 -c 5
        }
#SQL Server Connections
define command{
       command_name     check_nt_sql_server_connections
       command_line     $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v COUNTER -l "\\SQLServer:General Statistics\\User
Connections","SQL Server Connections are %.f" -w 20 -c 40
        }
#SQL Database Data size (Total)
define command{
       command_name     check_nt_sql_database_data_size
       command_line     $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v COUNTER -l "\\SQLServer:Databases(_Total)\\Data
File(s) Size (KB)","SQL Server Databases Datafile size total is %.f" -w 5 -c 30
        }
#SQL Database Log size (Total)
define command{
       command_name     check_nt_sql_database_log_size
       command_line     $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v COUNTER -l "\\SQLServer:Databases(_Total)\\Log
File(s) Size (KB)","SQL Server Databases Logfile size total %.f" -w 10 -c 100
        }
#SQL Replication Agents (SharePoint Instance)
define command{
       command_name     check_nt_sql_replication_agents
       command_line     $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v COUNTER -l "\\MSSQL$Sharepoint:Replication
Agents\\Running","Number of SQL Server Replication Agents running are %.f"
        }
#SQL Database Log size (SharePoint Instance)

define command{
       command_name     check_nt_sql_database_log_size_sharepoint
       command_line     $USER1$/check_nt -H $HOSTADDRESS$ -s password -p 12489 -v COUNTER -l
"\\MSSQL$Sharepoint:Databases(_Total)\\Log File(s) Size (KB)","SQL Server Databases Logfile size total %.f" -w 10 -c 30
          }
```

Monitorització de Windows mitjançant NSClient++.

Per a vigilar els servidors remots Windows, farem servir NSClient++ http://www.nsclient.org o http://www.nsclient.org/nscp/.

NSClient és un programa per a Windows, amb llicència GPL, i una versió per a *Windows* de 32 bits i per a *Windows* de 64 bits. El programa ens el baixem de http://nsclient.org/nscp/downloads .

Aquest programa permet que els plugins *check_nt* i *check_nrpe* puguin interrogar el sistema Windows.

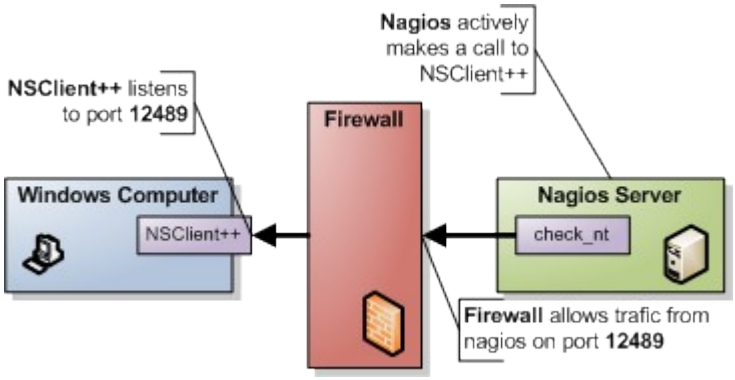
Així mateix, instal·la els components necessaris al Windows per a permetre monitorització activa i monitorització passiva. Amb aquesta darrera opció, el propi servidor Windows pot enviar avisos i alertes al servidor Nagios, però no la farem servir, nosaltres farem només monitorització activa.

Podem trobar una excel.lent explicació de l'arquitectura de NSClient++ i els seus diferents nivells funcionals a http://www.nsclient.org/nscp/wiki/doc/usage/nagios.

Veiem que NSClient++ pot escoltar els dos ports TCP de NSClient (*checks* per *check_nt*) i NRPE (*checks* per *check_nrpe*). NRPE també es pot fer servir per a monitoritzar Linux, però no el farem servir ni per a Linux ni per a *Windows*. Tots dos funcionen de manera arquitecturalment molt similar.

Nosaltres triarem la vigilància per NSClient++ pur, sense NRPE, per simplicitat i perquè farem en inici únicament *checks* bàsics. El mètode "*NSClient*" fa servir el plugin "*check_nt",* i aquest que porta els *checks* bàsics necessaris incorporats.

Per fer vigilància per NRPE hem de fer i definir *scripts* a cada servidor per a fer funcionar els *checks* bàsics. Ara això no cal, la funcionalitat de monitorització bàsica ve incorporada a NSClient.



Els checks es fan des de Nagios, amb el *plugin check*_nt. Fent "# /usr/local/nagios/libexec/check_nt -h" trobem totes les opcions de la comanda. Usualment farem servir almenys les següents opcions (adequades a cada servidor); per exemple:

- check_nt!CLIENTVERSION: Versió del client:

- check_nt!UPTIME: *Uptime*

- check_nt!CPULOAD!-l 5,80,90 : càrrega de la CPU, a l'exemple, dels darrers 5 minuts. Genera un avís *warning* quan ha estat d'un 80% i una alerta *critical* quan ha estat del 90%.

- check_nt!MEMUSE!-w 80 -c 90 : ús de memòria, amb un *warning* quan estiguem al 80% i un *critical* quan estiguem al 90%.

- check_nt!USEDDISKSPACE!-l e -w 80 -c 90: espai de disc usat al disc E: amb un *warning* quan estiguem al 80% i un *critical* quan estiguem al 90%.

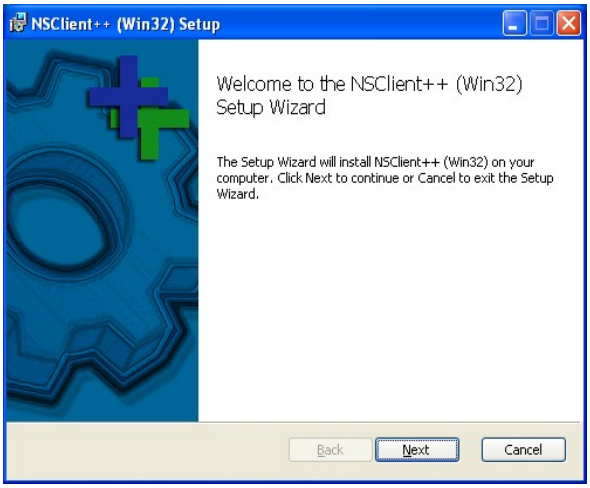
- check_nt!SERVICESTATE!-d SHOWALL -l SERVEI: estat del servei SERVEI de Windows. La descripció del servei és la que figura a "Service Name" quan ens mirem les propietats d'un servei de Windows qualsevol.

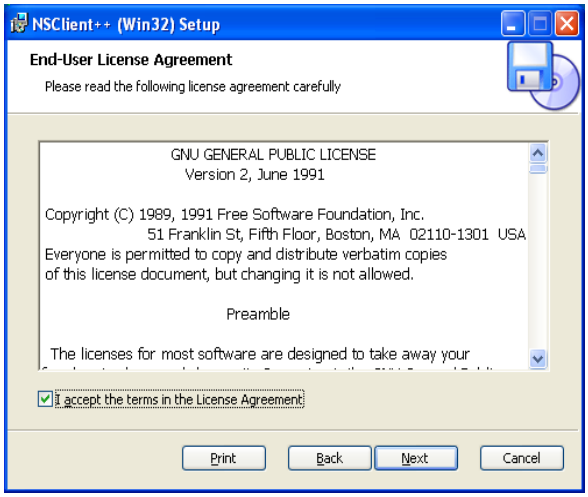
És molt interessant, tot i que inicialment no la faré servir, l'opció COUNTER, que permet interrogar qualsevol comptador de rendiment (*performance counter)* de Windows.

Instal.lació de NSClient++ en un servidor Windows 2000/2003/2003 R2 de 32 bit

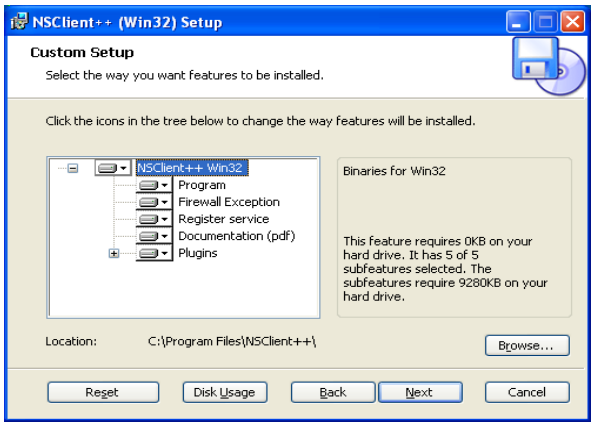
Tota la configuració del client que introduirem en les pantalles gràfiques d'instal·lació es troba en el fitxer de text "c:\program files\NSClient++\NSC.ini", situat a l'arrel del directori d'instal·lació del programa, i que podem modificar quan vulguem.

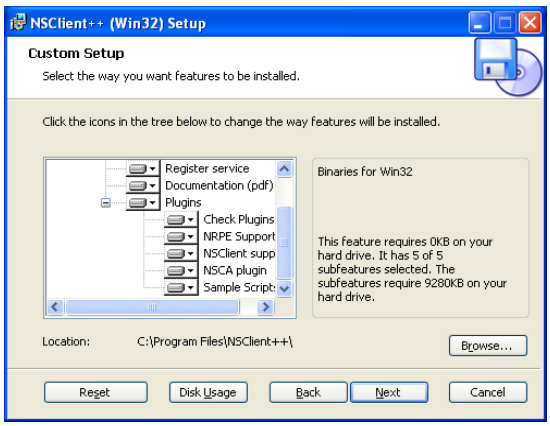
Acceptem la llicència:

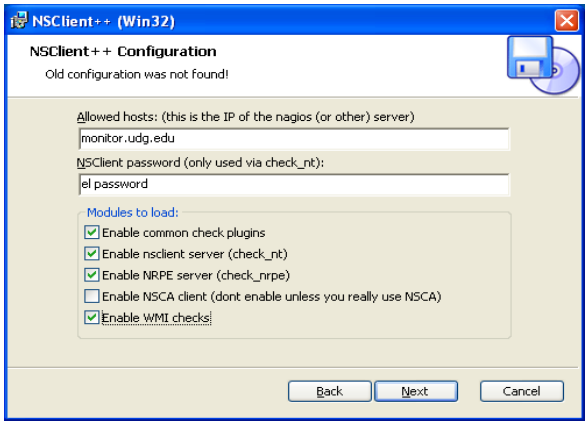


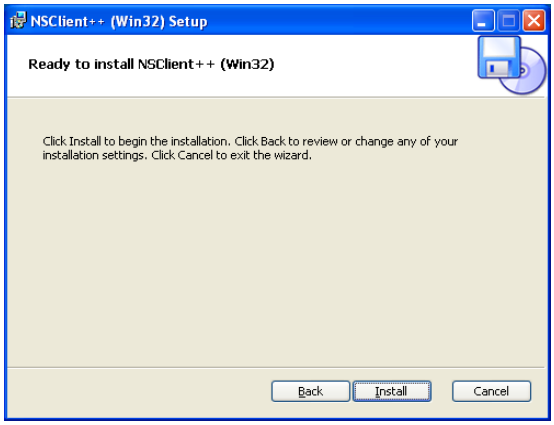


Seleccionem totes les opcions (per defecte):

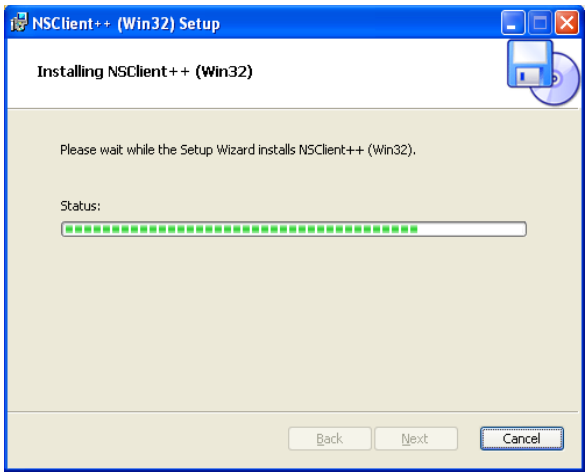


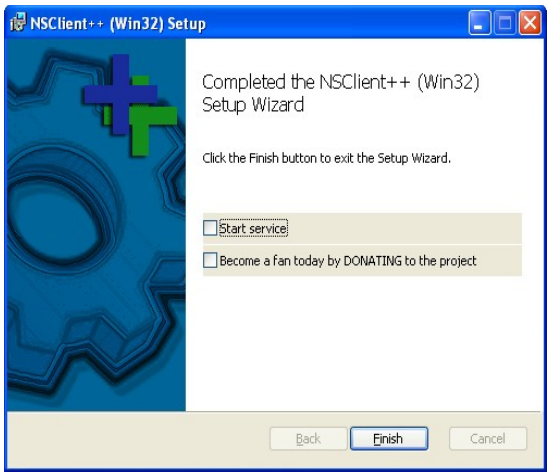






Deshabilitem NSCA (Nagios Service Check Acceptor, mecanisme de notificació passiva de *checks* a Nagios, que no farem servir en principi.

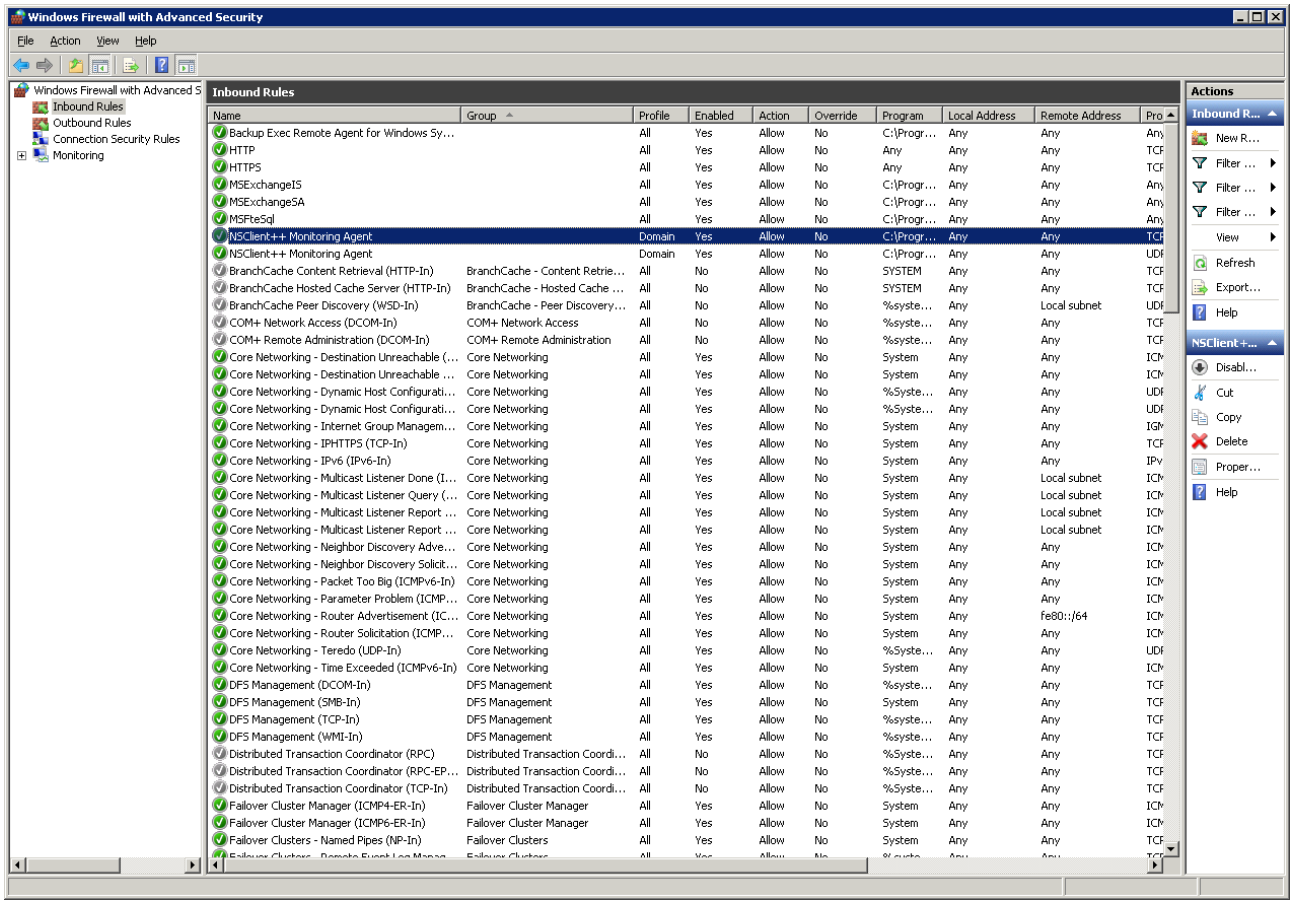




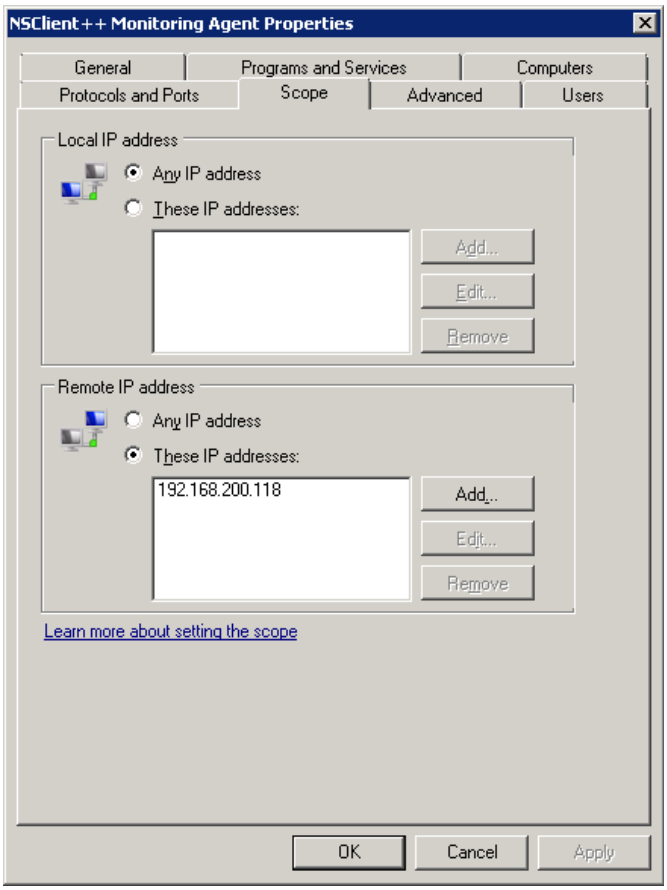
Així doncs, NSClient++ s'instal.la com un *servei* de Windows, l'anàleg en el món Microsoft dels *daemons* del món UNIX.

Posteriorment, caldrà configurar el tallafocs local dels servidors per a permetre les connexions entrants als ports que vulgui obrir el servei de NSClient++. La manera més senzilla en sistemes Windows 2003 és afegir una “excepció” al tallafocs per a l'executable, “NSClient++.exe”.
Això li diu al tallafocs que permeti les connexions entrants per a qualsevol port en el qual l'executable del servei estigui escoltant.

La instal.lació en servidors Windows 2008 /2008 R2 és idèntica, però el tallafocs de Windows ha canviat en aquestes dues versions, i cal configurar una excepció per al programa NSClient++ com abans, però ara cal dir que l'”àmbit” és “Domain” i “public”, altrament el tallafocs rebutja les peticions de monitorització. Val la pena fer notar que hi ha una versió de NSClient++ per a Windows en arquitectura de 32 bits i una per a 64 bits, aquesta darrera és la que instal.lem sobre Windows Server 2008 R2:



I en el tallafocs m'asseguro també que només accepta les peticions amb la IP del *Nagios:*



Creació del fitxer de configuració de Nagios "nom_servidor_windows.cfg" (a l'exemple "servidormailboxesexchange" per a *nova_maquina en el servidor de monitorització.*

Fa servir com a patró "windows-server" (definit a *templates.cfg*).

```
define host{
        use             windows-server
        host_name       servidormailboxesexchange
        alias           servidormailboxesexchange - Exchange 2010
        address         192.168.200.200
        }

define service{
        use                     generic-service
        host_name               servidormailboxesexchange
        service_description     NSClient++ Version
        check_command           check_nt!CLIENTVERSION
        }
define service{
        use                     generic-service
        host_name               servidormailboxesexchange
        service_description     Uptime
        check_command           check_nt!UPTIME
        }
define service{
        use                     generic-service
        host_name               servidormailboxesexchange
        service_description     CPU Load
        check_command           check_nt!CPULOAD!-l 5,80,90
        }
define service{
        use                     generic-service
        host_name               servidormailboxesexchange
        service_description     Memory Usage
        check_command           check_nt!MEMUSE!-w 80 -c 90
        }
define service{
        use                     generic-service
        host_name               servidormailboxesexchange
        service_description     C: - Disc Sistema
        check_command           check_nt!USEDDISKSPACE!-l c -w 80 -c 90
        }
define service{
        use                     generic-service
        host_name               servidormailboxesexchange
        service_description     W3SVC - Servidor Web
        check_command           check_nt!SERVICESTATE!-d SHOWALL -l W3SVC
        }
define service{
        use                     generic-service
        host_name               servidormailboxesexchange
        service_description     Exchange Information Store
        check_command           check_nt!SERVICESTATE!-d SHOWALL -l MSExchangeIS
        }
define service{
        use                     generic-service
        host_name               servidormailboxesexchange
        service_description     Exchange RPC Client Access
        check_command           check_nt!SERVICESTATE!-d SHOWALL -l MSExchangeRPC
        }
define service{
        use                     generic-service
        host_name               servidormailboxesexchange
        service_description     Exchange Mail Submission Services
        check_command           check_nt!SERVICESTATE!-d SHOWALL -l MSExchangeMailSubmission
        }
define service{
        use                     generic-service
        host_name               servidormailboxesexchange
        service_description     Exchange Replication
        check_command           check_nt!SERVICESTATE!-d SHOWALL -l MSExchangeRepl
        }
define service{
```

```
        use                     generic-service
        host_name               servidormailboxesexchange
        service_description     Exchange System Attendant
        check_command           check_nt!SERVICESTATE!-d SHOWALL -l MSExchangeMailboxAssistants
        }

define service{
        use                     generic-service
        host_name               servidormailboxesexchange
        service_description     Exchange AD Topology
        check_command           check_nt!SERVICESTATE!-d SHOWALL -l MSExchangeADTopology
        }
define service{
        use                     generic-service
        host_name               servidormailboxesexchange
        service_description     Exchange Service Host
        check_command           check_nt!SERVICESTATE!-d SHOWALL -l MSExchangeServiceHost
        }
# Create a service for monitoring the Explorer.exe process
# No monitoritzem cap procés, no ens fa falta.
#define service{
#       use                     generic-service
#       host_name               servidormailboxesexchange
#       service_description     Explorer
#       check_command           check_nt!PROCSTATE!-d SHOWALL -l ProcesPerMonitoritzar.exe
#       }
```

Notem que el que cal per a vigilar algun paràmetre del servidor és crear, en el fitxer de configuració de la vigilància del servidor concret, el que a Nagios en diem un *servei*. Aquest objecte *service* hereta normalment d'un patró (*generic-service*) les propietats relevants, i conté el nom del *host* o servidor que volem vigilar, una descripció del servei, i sobretot la comanda o *plugin* que invocarem per a efectuar la vigilància real.

Monitorització de Linux: per SSH.

Monitoritzar per SSH és:

- Senzill

- Segur

- Fàcil d'implementar, bé que laboriós.

Tot seguit, descriuré el procediment per incorporar una màquina qualsevol per ser monitoritzada per *Nagios*, sense entrar en el detall específic de la configuració de la màquina. Aquest detall usualment inclou o hauria d'incloure la monitorització d'algun servei especial que aquesta màquina està oferint.

Un excel.lent resum del funcionament de *ssh* sobre Linux el podem trobar a http://support.suso.com/supki/SSH_Tutorial_for_Linux.

Incorporar una nova màquina per ser monitoritzada per ssh amb nagios

- Instal·lar, si no hi és, dimoni de SSH a la màquina, escoltant el port per defecte (tcp 22)
- Crear, a la nova màquina, la parella de claus pública-privada que servirà per a autenticar la connexió ssh que efectuarà la màquina ***monitor*** cap a aquesta nova màquina:

Seguint http://wiki.nagios.org/index.php/Howtos:checkbyssh_RedHat

Primer, creem un usuari que és el que remotament es connectarà:

```
/usr/sbin/useradd -m nagios
```

Tot seguit, creem la clau de *login* SSH, de 1024 bits i encriptada amb l'algoritme DSA:

```
cd /home/nagios
mkdir .ssh
ssh-keygen -t dsa -b 1024 -f .ssh/id_dsa
```

com que no voldrem que en la connexió calgui entrar cap mena de contrasenya, quan l'instal·lador ens la demani premerem “intro” dos cops. Després farem:

```
cat .ssh/id_dsa.pub >> .ssh/authorized_keys
chown -R nagios:nagios .ssh
chmod 750 .ssh
chmod 640 .ssh/*
cat .ssh/id_dsa
```

Aquí, haurem protegit convenientment les claus contra lectura de qualsevol usuari del sistema excepte *nagios* i grup *nagios.* Alhora, serà autoritzat qualsevol sistema que tingui la clau pública “id_dsa” (l'hem afegit a l'anell de claus autoritzades de l'usuari *nagios*).

Ara copiarem la clau .ssh/id_dsa a la màquina monitoritzadora, i li'n canviarem el nom per “nom_servidor” per diferenciar-la de totes les altres claus de tots els altres servidors que monitoritzarem. El directori de la màquina “monitor” on les deixem totes és “/usr/local/nagios/etc/keys”. Després fem que *nagios:nagios* en sigui el propietari i l'usuari *nagios* l'únic que pot llegir-lo.

```
chown nagios:nagios <file name>
chmod 600 <file name>
```

Tot seguit, la instal.lació dels plugins

Aquesta és anàlega a la instal.lació que hem fet en el servidor. Primer, cal baixar-se i descomprimir *nagios-plugins-1.4.15.tar.gz; a*ssegurar-se que tenim els *headers* d'openssl instal·lats, perquè funcioni l'opció de compilació "with-openssl" : ***#yum install openssl-dev***

Donat que la *umask* torna *0022,* això vol dir que la màscara de permisos per defecte ja és 755 i per tant els permisos de destí ja són correctes.

Testejar i posar a punt la connexió des de monitor --> nom_servidor

```
# su - nagios
#./usr/local/nagios/libexec/check_by_ssh -p 22 -l nagios -i
/usr/local/nagios/etc/keys/remote1 -H 192.168.0.x -C
'/usr/local/nagios/libexec/check_disk -w 20% -c 10% -p /'
```

Des del servidor *monitor,* impersonem l'usuari *nagios* i fem una connexió cap a la nova màquina com ho farà el programari *nagios* (que recordem-ho, corre amb l'usuari *nagios*) executant el *plugin* "check_by_ssh" cap a la *nova_maquina* (paràmetre -H ip_nova_maquina), i presentant la clau pública (*-i /usr/local/nagios/etc/keys/nova_maquina*) de l'usuari *nagios* a la *nova_maquina*, que hem creat abans.

El primer cop en fer-ho manualment ens demana que ho acceptem, diem "yes", i el segon cop que executem la mateixa comanda ja funciona correctament, donat que al fitxer "/home/nagios/.ssh/knowh_hosts" ja hi ha posat la clau pública del servidor contra el qual ens estem connectant (que hem acceptat responent "yes"). Qualsevol connexió posterior no preguntarà res sempre i quan el servidor remot s'identifiqui amb la mateixa clau.

Creació del fitxer de configuració de Nagios "nom_servidor_linux.cfg" per a nova_maquina en el servidor de monitorització.

Copiaré un fitxer de configuració de qualsevol altre servidor similar, i l'editaré convenientment per adequar-lo als serveis que cal monitoritzar. Per al cas d'exemple, és una màquina que munta per NFS dues particions d'un servidor de fitxers. Així doncs, vigilarem els serveis "portmapper", "status" i "nlockmgr"
Monitoritzarem les dues particions anteriors, a més a més de la "/boot", la *root "/", i* la *swap.*
Per a fer aquesta vigilància, farem servir un patró que haurem definit prèviament a "

```
###############################################################################
###############################################################################
#
# HOST DEFINITION
#
###############################################################################
###############################################################################

# Define a host for the remote machine

define host{
        use                     linux-server
        host_name               nova_maquina
        alias                   alias-de-la-maquina.administraciox.cat
        address                 192.168.200.200
        }


###############################################################################
###############################################################################
#
# SERVICE DEFINITIONS
#
###############################################################################
###############################################################################
```

```
# Notem que hi ha un primer parametre a la majoria de comandes, "22" que es el port

#Notem que hi ha un primer parametre a la majoria de comandes definides
# que es el port de SSH; hi es per al cas que el vulguessim canviar en les maquines
# monitoritzades; en aquest cas nomes hauriem de canviar el numero. Josep 15-9-2010


# Define a service to "ping" the remote machine
# El defineixo per a cada màquina que monitoritzo. Això es podria optimitzar fent servir "templates", ja
# ho ferem en una fase posterior.

define service{
        use                             remote-service         ; Name of service template to use
        host_name                       nova_maquina
        service_description             PING
        check_command                   check_ping!100.0,20%!500.0,60%
        }

# Define a service to check the disk space of the root partition
# on the remote machine.  Warning if < 20% free, critical if
# < 10% free space on partition. O el que és mateix, "warning" a 80$ i "critical" a 90%.

define service{
        use                             remote-service         ; Name of service template to use
        host_name                       nova_maquina
        service_description             Particio root
        check_command                   check_remote_disk!22!10%!5%!/
        }

define service{
        use                             remote-service         ; Name of service template to use
        host_name                       nova_maquina
        service_description             Particio boot
        check_command                   check_remote_disk!22!20%!10%!/boot
        }

define service{
        use                             remote-service         ; Name of service template to use
        host_name                       nova_maquina
        service_description             Particio /users (muntada per NFS)
        check_command                   check_remote_disk!22!20%!10%!/users
        }

define service{
        use                             remote-service         ; Name of service template to use
        host_name                       nova_maquina
        service_description             Particio /NASestudiants (muntada per NFS)
        check_command                   check_remote_disk!22!20%!10%!/NASestudiants
        }


# Define a service to check the number of currently logged in
# users on the remote machine.  Warning if > 20 users, critical
# if > 50 users.
# Molts d'usuaris "logats" no hi han de ser, normalment.
define service{
        use                             remote-service         ; Name of service template to use
        host_name                       nova_maquina
        service_description             Usuaris Actuals
        check_command                   check_remote_users!22!3!5
        }


# Define a service to check the number of currently running procs
# on the remote machine.  Warning if > 250 processes, critical if
# > 400 users.
define service{
        use                             remote-service         ; Name of service template to use
        host_name                       nova_maquina
        service_description             Processos Totals
        check_command                   check_remote_procs!22!150!200!RSZDT
        }

define service{
        use                             remote-service         ; Name of service template to use
        host_name                       nova_maquina
        service_description             Carrega actual
        check_command                   check_remote_load!22!5.0,4.0,3.0!10.0,6.0,4.0
        }

define service{
        use                             remote-service         ; Name of service template to use
        host_name                       nova_maquina
        service_description             Us del Swap
        check_command                   check_remote_swap!22!20!10
        }

define service{
        use                             remote-service         ; Name of service template to use
        host_name                       nova_maquina
```

```
        service_description             RPC - portmapper
        check_command                   check_rpc_remot!22!portmapper!2
        }

define service{
        use                             remote-service         ; Name of service template to use
        host_name                       nova_maquina
        service_description             RPC - nlockmgr
        check_command                   check_rpc_remot!22!nlockmgr!1,3,4
        }

define service{
        use                             remote-service         ; Name of service template to use
        host_name                       nova_maquina
        service_description             RPC - status
        check_command                   check_rpc_remot!22!status!1
        }

# Define a service to check SSH on the remote machine.
# Disable notifications for this service by default, as not all users may have SSH enabled.

define service{
        use                             remote-service         ; Name of service template to use
        host_name                       nova_maquina
        service_description             SSH
        check_command                   check_ssh!-p 22
        notifications_enabled           0
        }
```

Centreon

Com he explicat a l'apartat 2, Centreon és un *supershell* construit sobre Nagios, que afegirà al sistema ja muntat *monitor* la capacitat de monitoritzar els sistemes per SNMP amb tot un seguit de *checks* ja incorporats, a més a més de la possibilitat de recuperar les dades històriques recollides per Nagios i aplegades per NDOUtils i d'aquí per CentStorage cap a la base de dades MySQL que instal.larem com a requisit previ.

Pel que fa al programari pròpiament, des de la pàgina www.centreon.com podem baixar-nos les dues versions estables de *Centreon*, la 2.1.13 que va sortir el març de 2011 i la que acaba de sortir aquest juny de 2011, la 2.2.1.

La documentació que es pot trobar per *Internet* sobre el producte és raonablement succinta; vull dir que no s'estén en gaires floritures i no hi ha cap tutorial "for dummies" per a fer-ho. Especialment, les instruccions per a instal·lar-ho sobre Red Hat són molt esquemàtiques. Tanmateix, donat que la plataforma Linux homologada al Departament d'Informàtica de l'Administració X és Red Hat Enterprise 5, el primer que cal fer és provar-ho d'instal·lar sobre RedHat seguint el manual d'instal·lació, que trobem a http://forum.centreon.com/showthread.php/6042-Centreon-2-on-RHEL , no gaire bo. Segueixo finalment les instruccions de http://en.doc.centreon.com/Setup:Centos/Fedora/RHEL.

Instal.lació sobre Red Hat 5.x Enterprise

- Primer que res, fil per randa, instal·lar els prerequisits. Tot OK, algunes coses ja eren instal·lades (probablement per les comandes anteriors, alguna d'elles).

- Tot seguit, cal configurar l'SNMP. Seguint http://en.doc.centreon.com/Setup:SNMPConfiguration he configurat el meu fitxer "/etc/snmp/snmpd.conf. He triat el mateix nom de comunitat proposat com a "segur" en el document.

Provo si funciona *snmpwalk* (http://www.it-slav.net/blogs/2008/11/11/install-and-configure-snmp-on-rhel-or-centos/) interrogant el propi servidor per la interfície de *loopback*:

```
# snmpwalk -v 1 -c public -O e 127.0.0.1
```

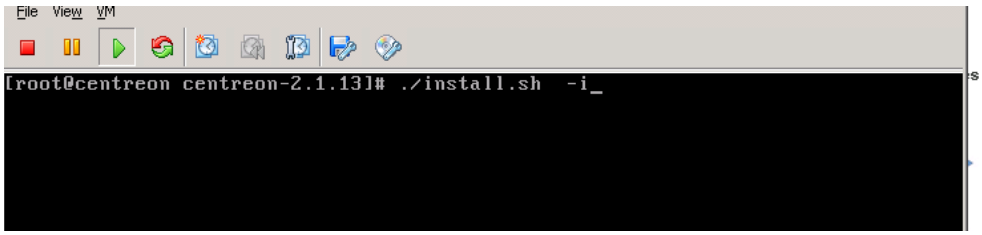
I efectivament em retorna l'output esperat: sap interrogar-se ell mateix.

Tot seguit, cal instal.lar el Nagios que estarà al cor de Centreon. Els paràmetres de compilació no són gaire diferents dels que van ser en instal.lar la màquina *monitor* de què hem parlat en tot el projecte. En tot cas, és una *prova de concepte*, per tant si funciona tot bé al final ja ho documentaré, si convé repetint el procés.

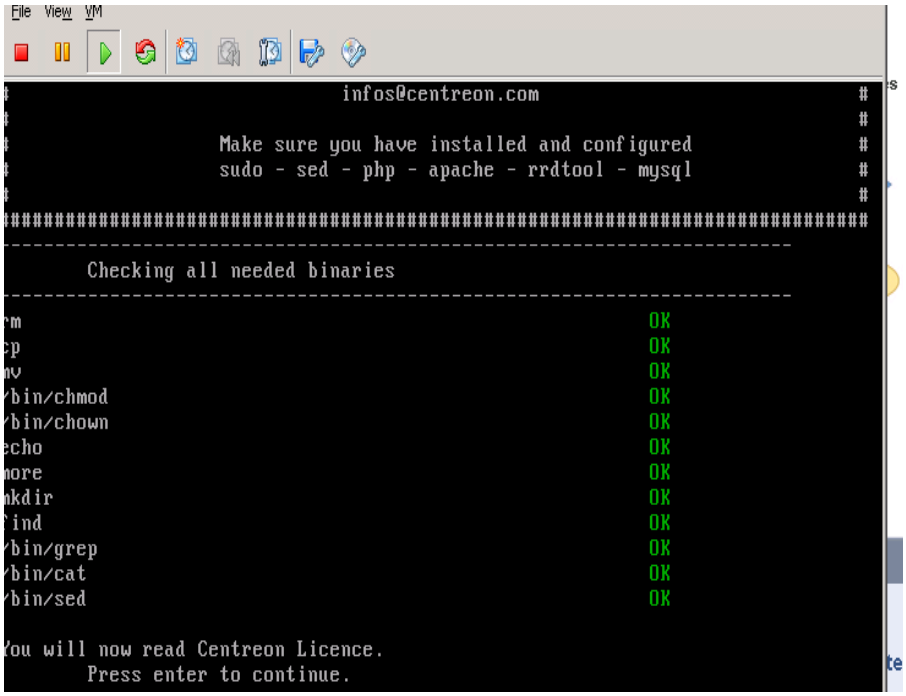
Tots els passos s'esdevenen amb èxit, fins que arribo a la compilació de centreon. És aquí quan obtinc errors.

Heus aquí el procés:

Per instal.lar-ho, em baixo el paquet, el descomprimeixo, o executo l'instal.lador:
./install.sh -i



L'instal.lador ens fa acceptar la llicència GPL, diem "y"



Diem "y" a totes les preguntes següents posteriors:

```
library.  If this is what you want to do, use the GNU Library General
Public License instead of this License.

Do you accept GPL license ?
[y/n], default to [n]:
> y
-------------------------------------------------------------------------
        Please choose what you want to install
-------------------------------------------------------------------------

Do you want to install : Centreon Web Front
[y/n], default to [n]:
> y

Do you want to install : Centreon CentCore
[y/n], default to [n]:
> y

Do you want to install : Centreon Nagios Plugins
[y/n], default to [n]:
> y

Do you want to install : Centreon Snmp Traps process
[y/n], default to [n]:
```

```
File  View  VM
-------------------------------------------------------------------------

Where is your Centreon directory?
default to [/usr/local/centreon]
>
Path /usr/local/centreon                                    OK

Where is your Centreon log directory
default to [/usr/local/centreon/log]
>
Path /usr/local/centreon/log                                OK

Where is your Centreon etc directory
default to [/etc/centreon]
>
Path /etc/centreon                                          OK

Where is your Centreon generation_files directory?
default to [/usr/local/centreon]
>
Path /usr/local/centreon                                    OK

Where is your Centreon variable library directory?
default to [/var/lib/centreon]
```

I veig que em surt un error, que després de seguir els passos d'instal.lació documentats a la wiki de Centreon, es queixa que no troba un fitxer:

```
Path /etc/centreon                                          OK

Where is your Centreon generation_files directory?
default to [/usr/local/centreon]
>
Path /usr/local/centreon                                    OK

Where is your Centreon variable library directory?
default to [/var/lib/centreon]
>
Path /var/lib/centreon                                      OK

Where is your CentPlugins Traps binary
default to [/usr/local/centreon/bin]
>
Path /usr/local/centreon/bin                                OK

Where is the RRD perl module installed [RRDs.pm]
default to [/usr/lib/perl5/RRDs.pm]
>
/usr/lib/perl5/RRDs.pm is not a valid file.                 CRITICAL

Where is the RRD perl module installed [RRDs.pm]
default to [/usr/lib/perl5/RRDs.pm]
```

No hi és! Ho comprovo:

```
Red Hat Enterprise Linux Server release 5.6 (Tikanga)
Kernel 2.6.18-238.5.1.el5 on an x86_64

centreon login: root
Password:
Last login: Fri Jun 10 11:14:31 on tty1
[root@centreon ~]# find / -name "RRDs.pm"
[root@centreon ~]# _
```

En aquest punt, i després d'investigar força estona, opto per provar el procediment d'instal.lació sobre la plataforma en la qual hi ha més documentació, com és Centreon sobre Debian 5.x. En aquest cas, la instal·lació reeixirà.

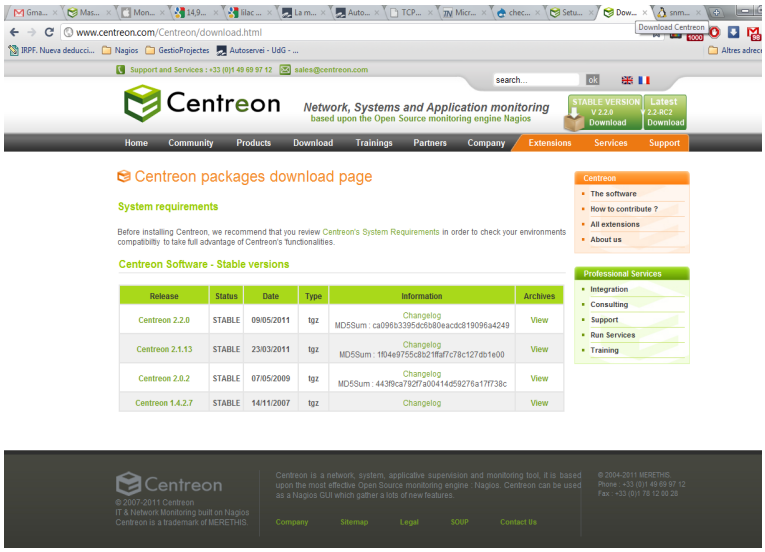
Per últim, comentar que hi ha un excel.lent *tutorial* de com instal.lar Centreon sobre CentOS (derivat de Red Hat Enterprise) http://nagioswiki.com/wiki/index.php/Installing_Centreon_on_Centos_5 , que obviaré malgrat les similituds entre Red Hat Enterprise i CentOS; els repositoris són diferents i no paga la pena provar CentOS ara mateix.

Instal.lació sobre una Debian.

Ben recentment, han sortit dues noves versions de tots dos productes, Debian 6 i Centreon 2.2.0. Després de provar la instal.lació de Centreon 2.2.0 i que no funcioni correctament, acabaré instal.lant la versió més documentada sobre la plataforma més usada, com són Centreon 2.1.13 (que

va sortir el març de 2011) sobre Debian 5 en 32 bits.

Em baixo totes dues versions del producte, i començo per instal.lar la darrera versió de Centreon, 2.2.0 sobre Debian 5 x32.



Instruccions:

http://planetit.ws/linux/debian/nagios-3-with-centreon-21-on-debian-based-distributions.html
però sobretot:
http://en.doc.centreon.com/Setup:CompileNagios

Preparatius i prerequisits:
http://en.doc.centreon.com/Setup:Prerequisite/Debian/Ubuntu

Instal.lo doncs tots els paquets requerits:

```
#apt-get install sudo mailx lsb-release build-essential apache2 apache2-mpm-prefork php5 php5-mysql php-pear php5-ldap php5-snmp php5-gd mysql-server libmysqlclient15-dev rrdtool librrds-perl libconfig-inifiles-perl libcrypt-des-perl libdigest-hmac-perl libdigest-sha1-perl libgd-gd2-perl snmp snmpd libnet-snmp-perl libsnmp-perl libgd2-xpm libgd2-xpm-dev libpng12-dev postfix
```

La instal.lació de *MySQL* ens demana d'introduir la contrasenya de l'usuari privilegiat, cosa que fem, i anotem per més tard.

Instal.lació Nagios: http://en.doc.centreon.com/Setup:CompileNagios

La instal.lació de Nagios aquí és com vaig fer-la per a *monitor*.
Primer que res, creació dels usuaris i grups que farà servir *Nagios*.

```
/usr/sbin/useradd -m nagios
/usr/sbin/usermod -G nagios nagios
/usr/sbin/groupadd nagcmd
/usr/sbin/usermod -G nagcmd nagios
/usr/sbin/usermod -G nagcmd www-data
```

Donat que *apache* sobre Debian corre amb l'usuari www-data, aquí afegim *www-data* dins del grup *nagcmd*, a diferència del que feia sobre Red Hat.

```
cd /root
 wget http://prdownloads.sourceforge.net/sourceforge/nagios/nagios-3.2.3.tar.gz
 tar -zxvf nagios-3.2.3.tar.gz
 cd nagios-3.2.3
  ./configure --prefix=/usr/local/nagios --with-command-group=nagcmd --enable-
nanosleep --enable-event-broker
```

Baixem el paquet amb *wget* i configurem. Al final del procés de configuració, ens surt el sumari de les opcions que hem triat:

```
---------------------------------------------------------------------
** Configuration summary for nagios 3.2.3 10-03-2010 ***:

 General Options:
 -------------------------
        Nagios executable:  nagios
        Nagios user/group:  nagios,nagios
       Command user/group:  nagios,nagcmd
            Embedded Perl:  no
             Event Broker:  yes
        Install ${prefix}:  /usr/local/nagios
                Lock file:  ${prefix}/var/nagios.lock
   Check result directory:  ${prefix}/var/spool/checkresults
           Init directory:  /etc/init.d
  Apache conf.d directory:  /etc/apache2/conf.d
             Mail program:  /usr/bin/mail
                  Host OS:  linux-gnu

 Web Interface Options:
 ------------------------
                 HTML URL:  http://localhost/nagios/
                  CGI URL:  http://localhost/nagios/cgi-bin/
 Traceroute (used by WAP):  /usr/sbin/traceroute

---------------------------------------------------------------------
```

Després haurem de fer successivament tots els “make” següents:

```
#make install
```

Instal.la el programa principal, els CGIs, i els fitxers HTML.

```
#make install-init
```

Instal.la l'script d'inici a /etc/init.d

```
#make install-commandmode
```

Instal.la i configura els permisos en el directori que tindrà els fitxers de comandes externes (si n'hi posem).

```
#make install-config
```

Instal.la els fitxers de configuració de mostra a /usr/local/nagios/etc.

```
#make install-webconf
```

Instal.la els fitxers de l'aplicació web del Nagios al directori de configuració de l'*apache.*

Si volem entrar a la interfície de gestió web, en caldrà donar d'alta els usuaris amb la seva contrasenya per entrar a l'aplicació; farem servir el mecanisme habitual per a generar un fitxer de contrasenyes per a aplicacions amb autenticació simple per a apache:

```
# htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin
```

per a donar d'alta l'usuari *nagiosadmin* amb la seva contrasenya; i així successivament per a d'altres usuaris, sempre concatenant el resultat de la comanda amb el fitxer ja existent *htpasswd.users*

Instal.lació plugins Nagios: http://en.doc.centreon.com/Setup:CompileNagiosPlugins

Primer, els requeriments:

```
#aptitude install dnsutils fping libssl-dev libldap2-dev libpq-dev
libradiusclient-ng-dev smbclient
```

Després, ens baixem el paquet. Com que tenim una instal.lació mínima amb *netinstall* de Debian, naturalment sense interfície gràfica, em baixaré el paquet amb *wget:*

```
#cd /root
#wget http://prdownloads.sourceforge.net/sourceforge/nagiosplug/nagios-plugins-
1.4.15.tar.gz
#tar -zxvf nagios-plugins-1.4.15.tar.gz
#cd nagios-plugins-1.4.15

#./configure --enable-perl-modules --with-nagios-user=nagios --with-nagios-
group=nagios --enable-extra-opts --with-openssl=/usr/include/openssl

----------------
config.status: creating po/Makefile
            --with-apt-get-command: /usr/bin/apt-get
              --with-ping6-command: /bin/ping6 -n -U -w %d -c %d %s
               --with-ping-command: /bin/ping -n -U -w %d -c %d %s
                       --with-ipv6: yes
                      --with-mysql: /usr/bin/mysql_config
                   --with-openssl: yes
                    --with-gnutls: no
               --enable-extra-opts: yes
                       --with-perl: /usr/bin/perl
             --enable-perl-modules: yes
                     --with-cgiurl: /nagios/cgi-bin
               --with-trusted-path: /bin:/sbin:/usr/bin:/usr/sbin
                   --enable-libtap: no
-------------------
```

Compilo i instal.lo el programa, reinicio l'apache i el nagios, i :

```
#make
#make install

#/etc/init.d/apache2 restart
#/etc/init.d/nagios restart
```

```
#ln -s /etc/init.d/nagios /etc/rcS.d/S99nagios
```

Com hem vist en parlant dels mòduls que afegeixen extensibilitat a Nagios, NDOUtils és un d'ells i Centreon el necessita per a guardar en la base de dades MySQL –que instal.larem tot seguit-- les dades històriques.

Instal.lació NDOUtils : http://en.doc.centreon.com/Setup:ndoutils

```
# cd /root
# wget http://sourceforge.net/projects/nagios/files/ndoutils-1.x/ndoutils-
1.4b9/ndoutils-1.4b9.tar.gz

# tar -zxvf ndoutils-1.4b9.tar.gz
# cd ndoutils-1.4b9

# ./configure --prefix=/usr/local/nagios/ --enable-mysql --disable-pgsql \
   --with-ndo2db-user=nagios --with-ndo2db-group=nagios
```

Ens informa de les opcions que hem triat, ok!:

```
--------------
 *** Configuration summary for ndoutils 1.4b9 10-27-2009 ***:

 General Options:
 -------------------------
 NDO2DB user:    nagios
 NDO2DB group:   nagios

Review the options above for accuracy.  If they look okay,
type 'make' to compile the NDO utilities.
-----------------
```

Tot seguit, seguim les instruccions i compilem les utilitats NDO i seguim les instruccions, copiant els mòduls de NDO per a Nagios i els fitxers de configuració dins dels directoris que toquen, amb els permisos que toquen. Després modifiquem els scripts d'inicialització del sistema perquè s'engegui el dimoni de NDOUtils sempre que es reinicïi el sistema.

```
# make
# cp ./src/ndomod-3x.o /usr/local/nagios/bin/ndomod.o
#cp ./src/ndo2db-3x /usr/local/nagios/bin/ndo2db
# cp ./config/ndo2db.cfg-sample /usr/local/nagios/etc/ndo2db.cfg
# cp ./config/ndomod.cfg-sample /usr/local/nagios/etc/ndomod.cfg
# sudo chmod 774 /usr/local/nagios/bin/ndo*
# sudo chown nagios:nagios /usr/local/nagios/bin/ndo*

# cp ./daemon-init /etc/init.d/ndo2db
# chmod +x /etc/init.d/ndo2db
# update-rc.d ndo2db defaults
```

Fins aquí els prerequisits.

Instal.lació Centreon: http://en.doc.centreon.com/Setup:Centreon2

Em baixo el paquet amb wget:

```
#  wget http://download.centreon.com/centreon/centreon-2.2.0.tar.gz
```

Primer intentaré la instal.lació amb el paquet més nou, acabat de sortir.
El descomprimeixo i me'n vaig al directori; afegeixo el directori “bin” de Nagios al *path* i instal.lo seguint la documentació de *Centreon:*

```
# tar zxvf centreon-2.2.0.tar.gz
# cd centreon-2.2.0
# export PATH="$PATH:/usr/local/nagios/bin/"
# ./install.sh -i
```

La sortida del procés:

```
here is your SNMPTT binaries directory
default to [/usr/local/centreon/bin/]
>
/usr/local/centreon/bin/                                    OK
Finding Nagios group :                                      nagios
Finding Apache user :                                       www-data
Preparing Centreon temporary files
/tmp/centreon-setup exists, it will be moved...
Change macros for CentPluginsTraps                          OK
Installing the plugins Trap binaries                        OK
Change macros for snmptrapd.conf                            OK
Change macros for snmptt.ini                                OK
Install : snmptrapd.conf                                    OK
Install : snmp.conf                                         OK
Install : snmptt.ini                                        OK
Install : snmptt                                            OK
Install : snmpttconvertmib                                  OK
Create /etc/centreon/instCentPlugins.conf                   OK
################################################################################
#                                                                              #
#                 Go to the URL : http://centreon.administraciox.cat/centreon/
#
#                            to finish the setup                               #
#                                                                              #
#                  Report bugs at http://forge.centreon.com                    #
#                                                                              #
#                         Thanks for using Centreon.                           #
#                          -----------------------                             #
#                        Contact : infos@centreon.com                          #
#                          http://www.centreon.com                             #
#                                                                              #
################################################################################
```

Posteriorment, donat que no funcionava correctament, repetiré aquesta darrera part del procés amb Centreon 2.1.13; em caldrà desinstal.lar la versió prèvia 2.2.0 i per a fer-ho, caldrà:

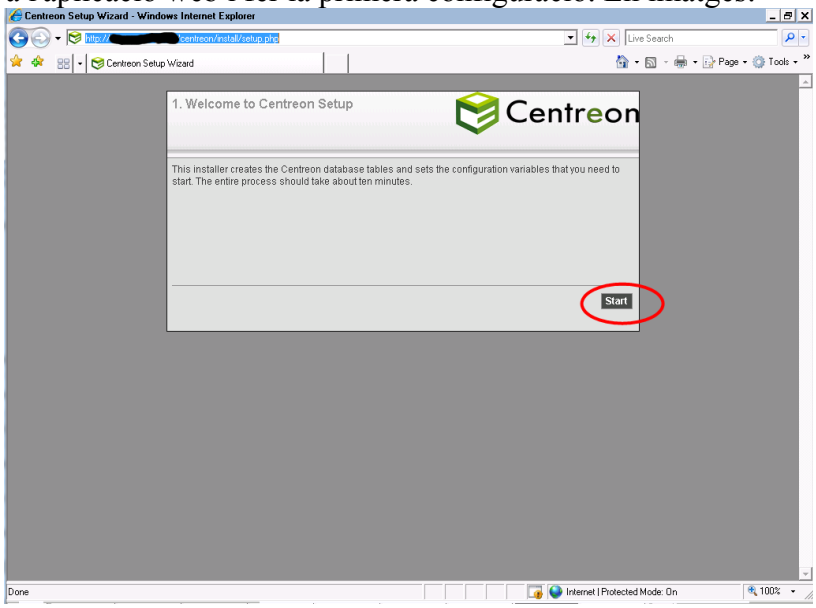
- Esborrar, al *MySQL* instal·lat localment, les bases de dades que *Centreon* hi deixa, i tot seguit esborrar els directoris on viu l'aplicació; com que sóc jo qui ha instal.lat les aplicacions compilant-les “manualment”, només em caldrà esborrar els directoris on sé que hi ha alguna cosa:

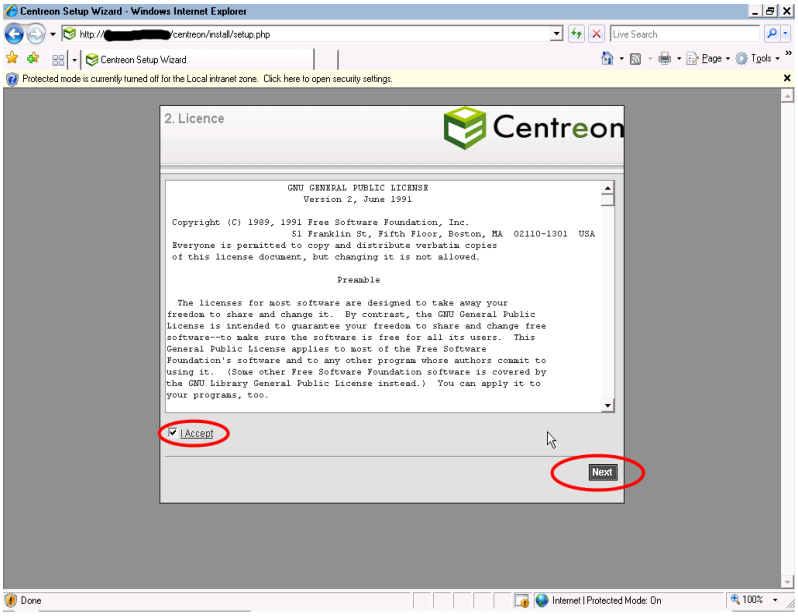
```
#mysql> drop database centreon;
#mysql> drop database centstatus;
#mysql> drop database centstorage;

#rm -rf /usr/local/centreon
#rm -rf /etc/centreon
#rm -rf /var/lib/centreon
#rm -rf /var/run/centreon
```

I tot seguit refer la instal.lació amb la versió 2.1.13. Ara sí que tot funcionarà bé.

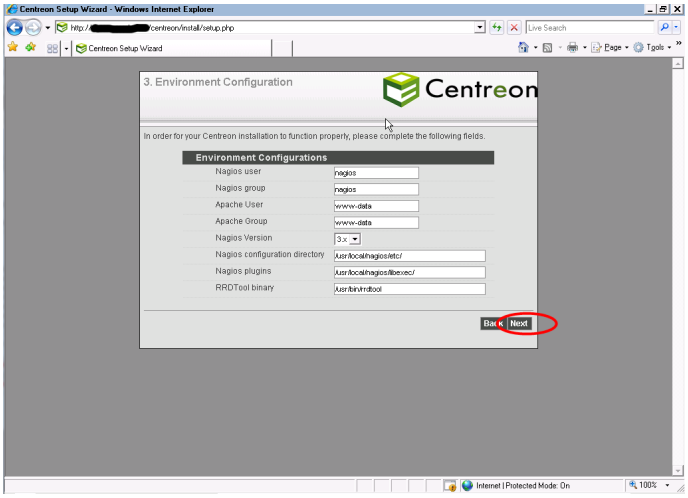
Configuració Centreon – Interfície Web.

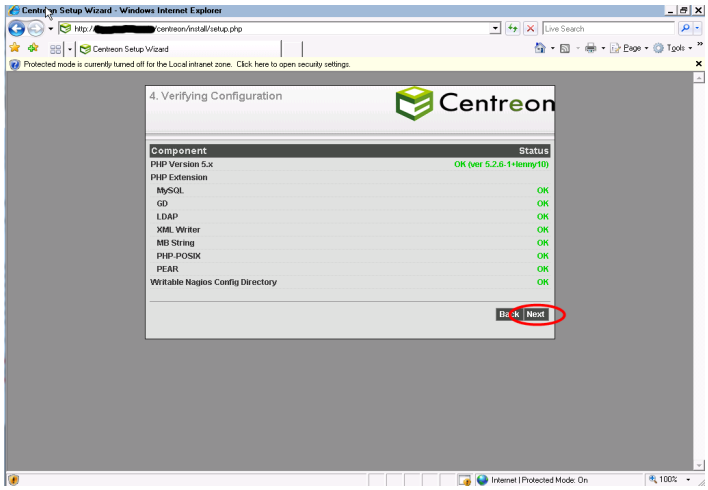
Ara hauré d'entrar a l'aplicació web i fer la primera configuració. En imatges:

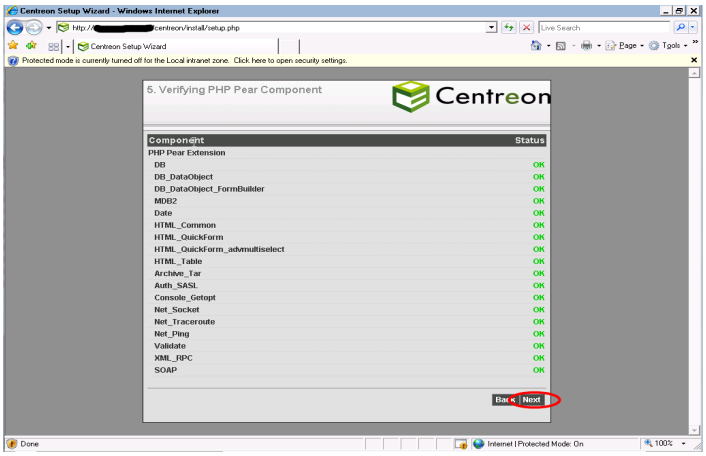




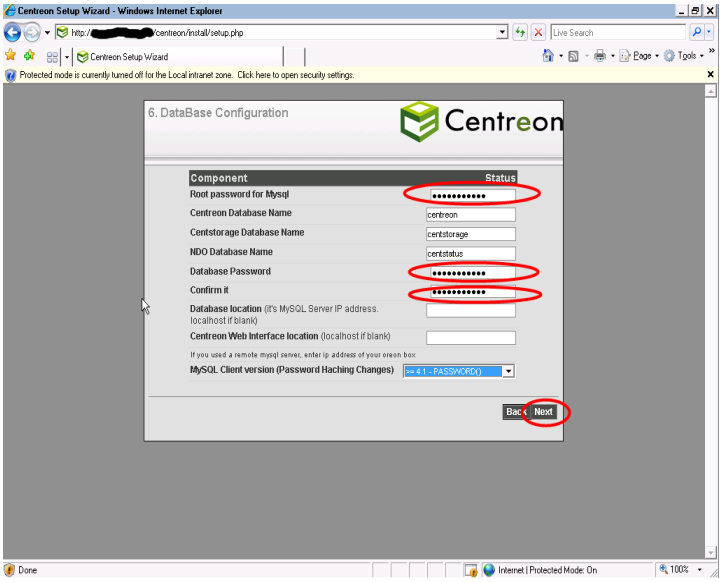
Acceptem la llicència, i confirmem els *path* que surten. Tot OK, ja sabem que s'instal·larà sobre el nostre *Nagios,* els camins són correctes.

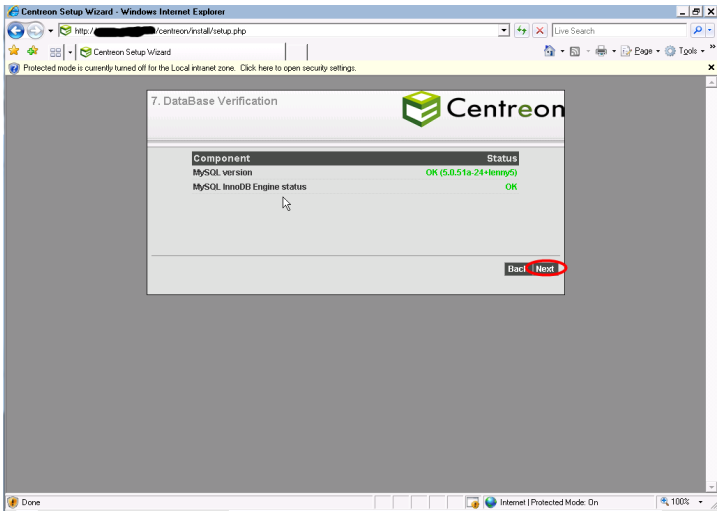




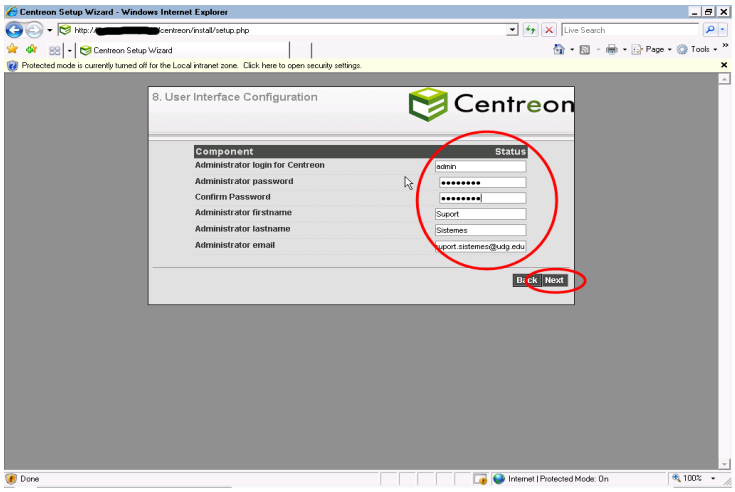


Naturalment, l'instal.lador ens demanarà la contrasenya del *root* de MySQL, ens proposarà els noms de les bases de dades on deixar la seva informació, i el nom de la base de dades de NDO d'on *pouarà* aquesta informació.

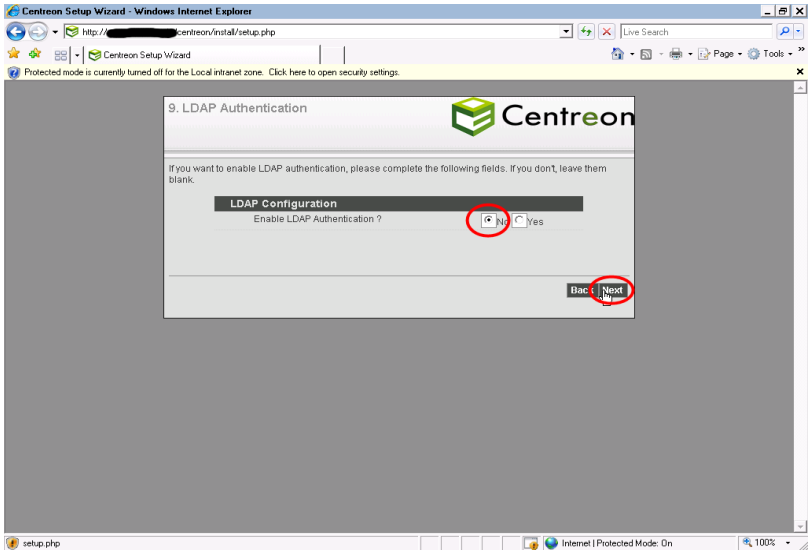


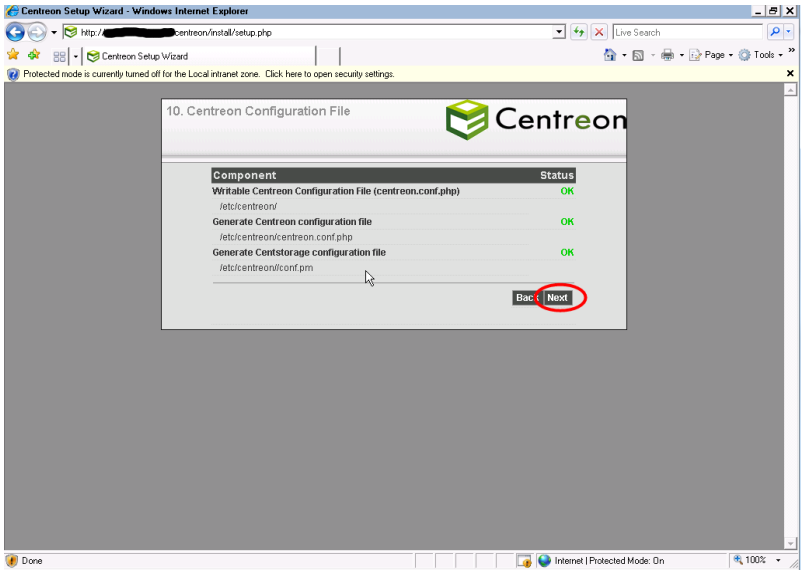


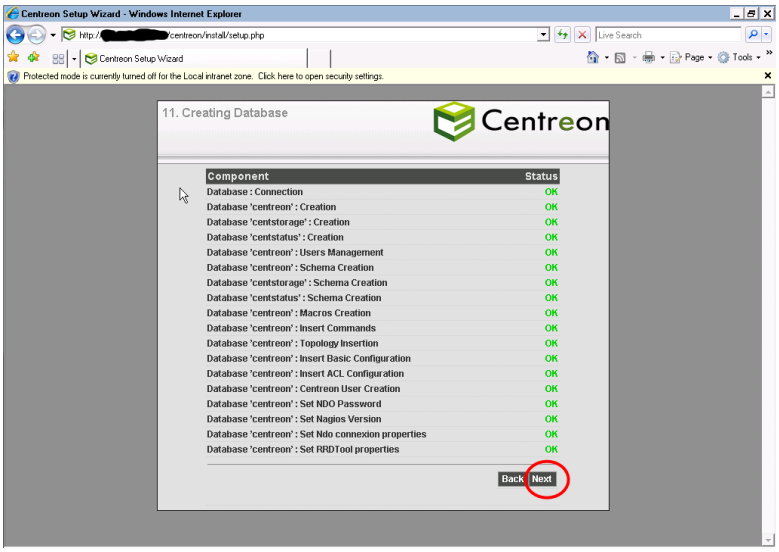
Trio l'usuari i contrasenya de l'administrador de *Centreon:*

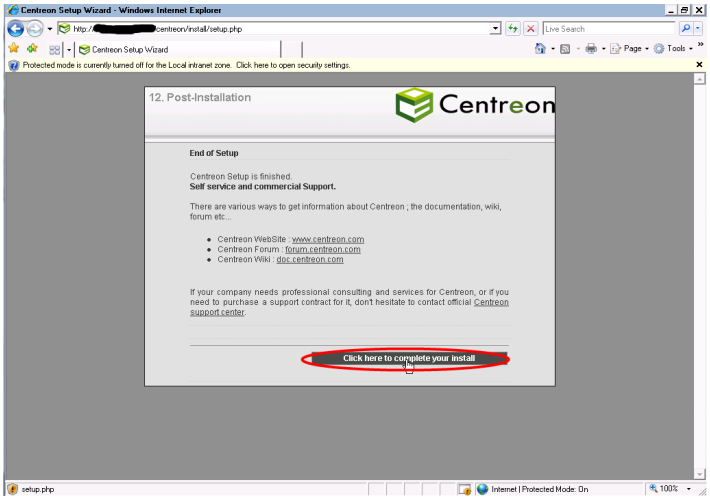


No farem servir autenticació per LDAP; poso que “No”.


Centreon Setup Wizard - Windows Internet Explorer
http://[redacted]centreon/install/setup.php
Centreon Setup Wizard
Protected mode is currently turned off for the Local intranet zone. Click here to open security settings.
9. LDAP Authentication
Centreon
If you want to enable LDAP authentication, please complete the following fields. If you don't, leave them blank.
LDAP Configuration
Enable LDAP Authentication ?
No
Yes
Back
Next
setup.php
Internet | Protected Mode: On
100%


Centreon Setup Wizard - Windows Internet Explorer
http://[redacted]centreon/install/setup.php
Centreon Setup Wizard
Protected mode is currently turned off for the Local intranet zone. Click here to open security settings.
10. Centreon Configuration File
Centreon
Component
Status
Writable Centreon Configuration File (centreon.conf.php)
OK
/etc/centreon/
Generate Centreon configuration file
OK
/etc/centreon/centreon.conf.php
Generate Centstorage configuration file
OK
/etc/centreon//conf.pm
Back
Next
Done
Internet | Protected Mode: On
100%


Centreon Setup Wizard - Windows Internet Explorer
http://[redacted]centreon/install/setup.php
Centreon Setup Wizard
Protected mode is currently turned off for the Local intranet zone. Click here to open security settings.
11. Creating Database
Centreon
Component
Status
Database : Connection
OK
Database 'centreon' : Creation
OK
Database 'centstorage' : Creation
OK
Database 'centstatus' : Creation
OK
Database 'centreon' : Users Management
OK
Database 'centreon' : Schema Creation
OK
Database 'centstorage' : Schema Creation
OK
Database 'centstatus' : Schema Creation
OK
Database 'centreon' : Macros Creation
OK
Database 'centreon' : Insert Commands
OK
Database 'centreon' : Topology Insertion
OK
Database 'centreon' : Insert Basic Configuration
OK
Database 'centreon' : Insert ACL Configuration
OK
Database 'centreon' : Centreon User Creation
OK
Database 'centreon' : Set NDO Password
OK
Database 'centreon' : Set Nagios Version
OK
Database 'centreon' : Set Ndo connexion properties
OK
Database 'centreon' : Set RRDTool properties
OK
Back
Next



I ja hi som, s'ha completat correctament la instal.lació i configuració inicials.

Tallafocs.

Regles de tallafocs per a Centreon:
https://help.ubuntu.com/community/IptablesHowTo#Basic%20iptables%20howto

Permetrem a les sessions establertes de rebre tràfic:

```
# iptables -A INPUT -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT
```

També permetrem tot el tràfic entrant amb destinació al port TCP 22 de SSH:

```
# iptables -A INPUT -p tcp --dport ssh -j ACCEPT
```

Com que l'aplicació Centreon és *web,* permetrem també el tràfic web entrant:

```
# iptables -A INPUT -p tcp --dport 80 -j ACCEPT
# iptables -A INPUT -p tcp --dport 443 -j ACCEPT
```

Per al Backup Exec, necessitem el port de l'agent RALUS (tcp 10000) i els de connexió dinàmica quan es fa el backup, el rang TCP 65471:65535 (http://www.symantec.com/business/support/index?page=content&id=TECH43417). Així doncs, caldrà fer diferents coses.
-Editar /etc/services a "centreon" perquè on deia "webmin" digui "ndmp", "Network Data Management Protocol".

```
# iptables -A INPUT -p tcp -s montsalvat --dport ndmp -j ACCEPT
# iptables -A INPUT -p tcp -s montsalvat.administraciox.cat --dport 65471:65535
-j ACCEPT
```

Ara, hem d'acceptar els "echo request, ICMP tipus 8" i l'"echo reply", ICMP tipus zero:

```
iptables -A INPUT -p icmp --icmp-type 8 -s 0/0 -d centreon.administraciox.cat -m
state --state NEW,ESTABLISHED,RELATED -j ACCEPT

iptables -A OUTPUT -p icmp --icmp-type 0 -s centreon.administraciox.cat -d 0/0
-m state --state ESTABLISHED,RELATED -j ACCEPT

iptables -A INPUT -p icmp --icmp-type 8 -s 0/0 -d localhost -m state --state
NEW,ESTABLISHED,RELATED -j ACCEPT

iptables -A OUTPUT -p icmp --icmp-type 0 -s localhost -d 0/0 -m state --state
ESTABLISHED,RELATED -j ACCEPT
```

i perquè Centreon es pugui fer *ping* a ell mateix, cal acceptar *icmp echo request* amb origen ell mateix i destinació a qualsevol de les seves interfícies:

```
iptables -A INPUT -p icmp --icmp-type 8 -s 127.0.0.1 -d 0/0 -m state --state
ESTABLISHED,RELATED -j ACCEPT
```

Un cop escrits tots els paquets que admetem, rebutjarem qualsevol altre paquet d'entrada.
Farem també una altra cosa molt important: rebutjar qualsevol paquet que arribi d'una interfície amb destinació a una altra; això no ho necessitem per res, la màquina no és un *router* ni té perquè enrutar paquets d'una VLAN a una altra.

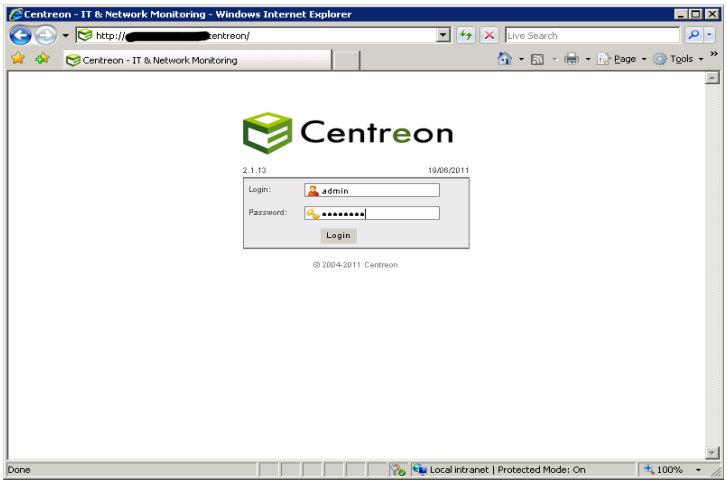
Rebutjem tot l'altre tràfic entrant – fem un default deny unless excepte que ho permetem explícitament:

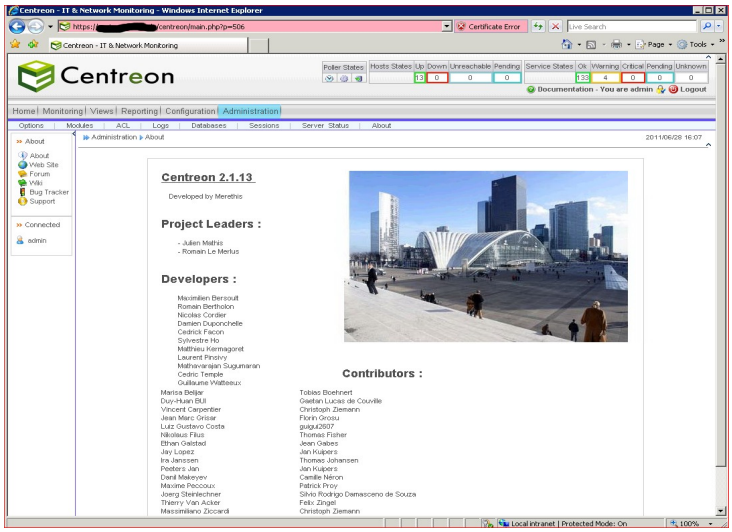
```
# iptables -A INPUT -j REJECT
# iptables -A FORWARD -j REJECT
```

És important de rebutjar qualsevol paquet del qual se'n vulgui fer “forward”, ja que no volem que el nostre equip enruti mai res, per seguretat.
Després, per tal que les regles de tallafocs que acabo de definir sobrevisquin un reinici, segueixo el procediment de http://wiki.debian.org/iptables, i creo el fitxer “/etc/network/if-pre-up.d/iptables”,. Aquest fitxer és un simple script que invoca “iptables-restore” i aquest s'alimenta del fitxer generat amb “iptables-save” que he deixat a “/etc/iptables.conf”.

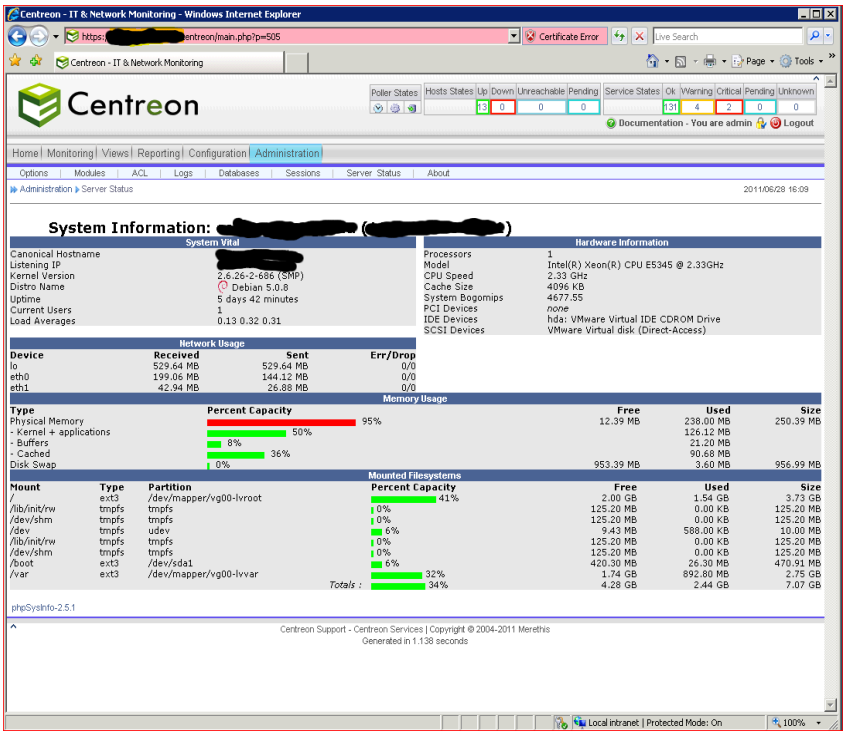
Procés de desplegament i implantació

El sistema amb Centreon es va instal.lar el 10 de maig de 2011 a la màquina *centreon.administraciox.cat*. Les captures de pantalla s'han fet amb una setmana de diferència algunes d'elles, d'aquí que el nombre de *hosts* que apareix monitoritzat sigui lleugerament diferent, ja que es van incorporant màquines a la monitorització.





El client es queixa del certificat perquè no té instal.lat localment l'entitat certificadora que ha emès el certificat per a l'Apache que hostatja el Centreon; no té a veure amb el projecte.



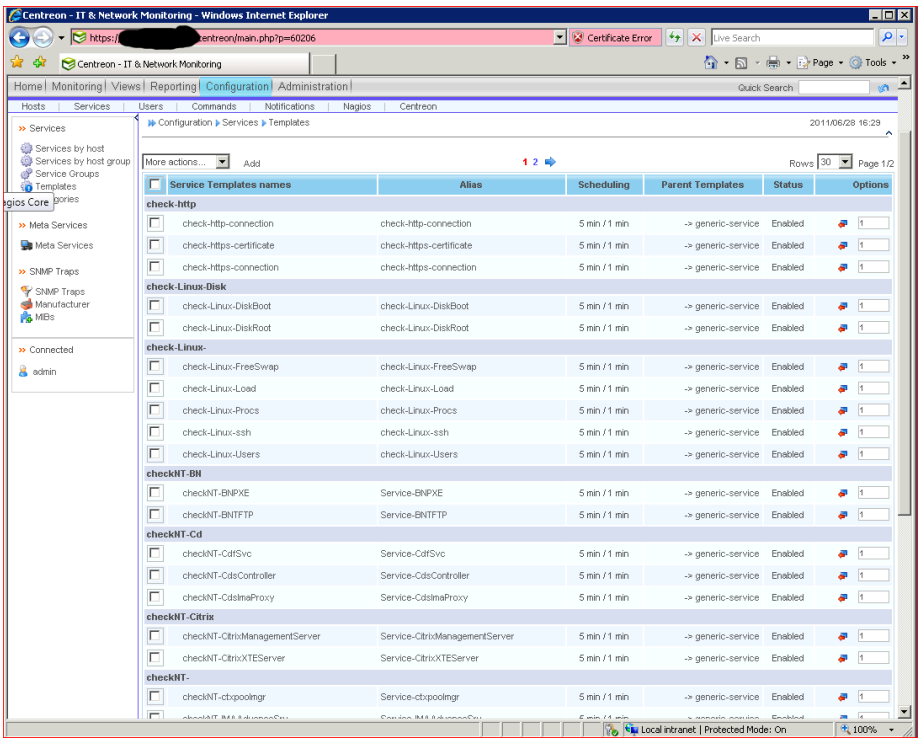
Per a provar-ho, hem definit

a) Templates de host:

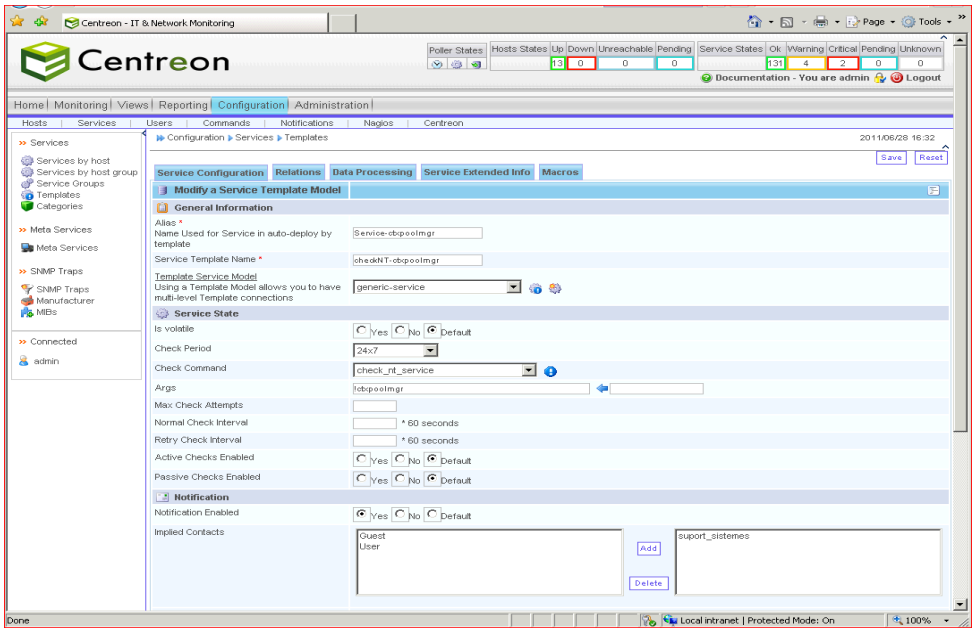
- Servers-Windows
- Servers-Linux
- Servers-Win2K
- Servers-Win2K3

A més a més dels que hi ha definits per defecte.

b) Serveis per a monitoritzar per *check_nt* els serveis de Citrix (que estiguin vius), hereten del patró *generic-service;* per a monitoritzar els serveis d'HTTP i HTTPS, i per mirar que el certificat de servidor no estigui caducat. Veiem que ràpidament, Centreon ens permet d'implementar *checks* de més laboriosa configuració editant a mà els fitxers de configuració de *Nagios*.



Aquí la definició (o modificació) d'un servei, en aquest cas “Service-ctxpoolmgr”, al qual li donaré el nom “checkNT-ctxpoolmgr”; heretarà del *template* “generic-service”, i li diré que notifiqui 24x7

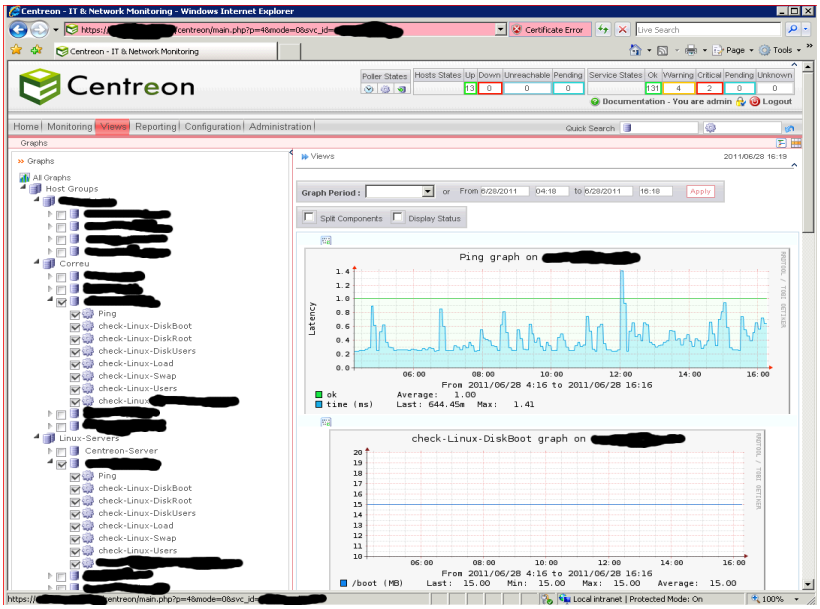


Finalment, a sota (no surt a la imatge), a “Notification” també i diré a qui ha de notificar, en el nostre cas el contacte “suport_sistemes”, al qual notificarem per email a l'adreça electrònica de la secció de sistemes.

Veiem doncs que Centreon permet, a alt nivell, de fer tot allò que feiem “a mà” amb Nagios, d'editar tots els fitxers d'objectes, i d'organitzar de manera eficient l'arbre d'herències.

Constatem també que Centreon ens satisfà dos requeriments importants:

- Poder obtenir gràfiques històriques de dades de Nagios. Aquest, sorgit amb el projecte de certificació per ISO20000 d'alguns dels processos del Departament d'Informàtica, serà ara possible. Es podrà triar el període, exportar les gràfiques, etc. Totes aquestes possibilitats vénen donades pels diferents components de l'arquitectura modular de Centreon, que hem anat explicant tot al llarg del treball.



Veiem aquí la gràfica dels serveis que monitoritzem de una màquina; podem triar-ne el període. La interfície web de Centreon m'agrupa, a l'esquerra de la imatge, tots els servidors agrupats sota el mateix "hostgroup", això és molt còmode. S'identifiquen els hostgroups "Correu" i "Linux-Servers", entre d'altres.

- Poder consultar per SNMP. Això era una necessitat important, donat que permet de monitoritzar equips per als quals no existeix cap altra via d'interrogació, usualment perquè són sistemes tancats (equips de xarxa amb programari *embedded*) o bé perquè no és possible de compilar-hi per exemple *nagios-plugins* o d'instal.lar-hi cap client natiu que pugui contestar per NRPE o qualsevol altre protocol que no sigui SNMP.

Per exemple, he definit, per a provar-ho, un servei "SNMP-Disk-C", amb alias "Disk-C", que hereta de *generic-service*, i que farà servir com a comanda *check_centreon_remote_storage* , definida probablement dins del fitxer *commands.cfg* del Nagios local; aquesta comanda invocarà alhora el binari que ja ve precompilat amb *Centreon* (amb els *plugins* de Nagios que incorpora *Centreon*) que veiem a la segona imatge de sota *check_centreon_snmp_remote_storage.*

File Edit View Favorites Tools Help

Centreon - IT & Network Monitoring

Centreon

Poller States	Hosts States	Up	Down	Unreachable	Pending	Service States	Ok	Warning	Critical	Pending	Unknown
		13	0	0	0		131	4	2	0	0

Documentation - You are admin Logout

Home | Monitoring | Views | Reporting | Configuration | Administration

Hosts | Services | Users | Commands | Notifications | Nagios | Centreon

» Services
- Services by host
- Services by host group
- Service Groups
- Templates
- Categories

» Meta Services
- Meta Services

» SNMP Traps
- SNMP Traps
- Manufacturer
- MIBs

» Connected
- admin

Configuration ▸ Services ▸ Templates 2011/06/28 16:41

Save Reset

Service Configuration | Relations | Data Processing | Service Extended Info | Macros

Modify a Service Template Model

General Information

Alias * / Name Used for Service in auto-deploy by template: Disk-C

Service Template Name *: SNMP-DISK-C:

Template Service Model / Using a Template Model allows you to have multi-level Template connections: generic-service

Service State

Is volatile: Yes No Default

Check Period:

Check Command: check_centreon_remote_storage

Args: !C!80!90!$USER2$!2

Max Check Attempts:

Normal Check Interval: * 60 seconds

Retry Check Interval: * 60 seconds

Active Checks Enabled: Yes No Default

Passive Checks Enabled: Yes No Default

Notification

Notification Enabled: Yes No Default

Implied Contacts: Guest, suport_sistemes, User — Add, Delete

100%

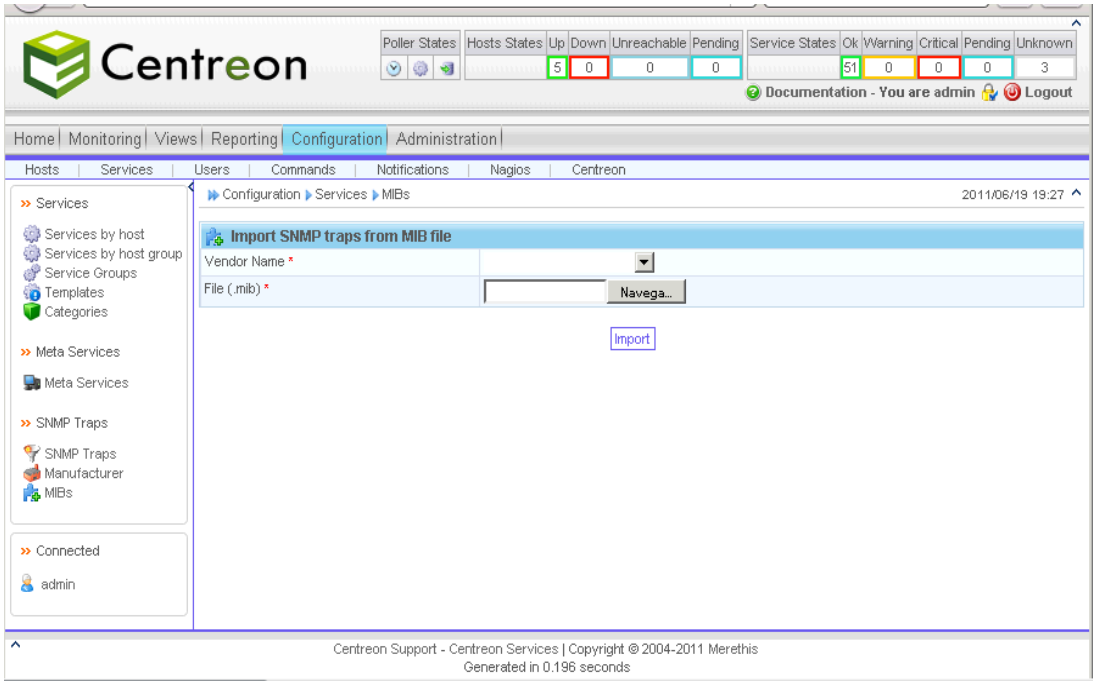
Centreon - IT & Network Monitoring - Windows Internet Explorer

View a Command

Check

Command Name	check_centreon_remote_storage
Command Line	$USER1$/check_centreon_snmp_remote_storage -H $HOSTADDRESS$ -n -d $ARG1$ -w $ARG2$ -c $ARG3$ -C $ARG4$ -v $ARG5$
Command Type	( ) Notification (x) Check
Check	
Notif	

Com és lògic en un sistema d'aquest tipus, es facilita la poder importar les MIBs SNMP per a poder dur a terme la monitorització de qualsevol equip:



4.2 Requisits d'implantació

Seguretat

Només l'usuari administrador, conegut per la secció de sistemes, tindrà accés total sense restriccions al sistema.

Tota la resta d'usuaris que s'hi donaran d'alta tindran l'accés per defecte als sistemes dels quals en són contactes; aquest és el comportament per defecte de Nagios.

L'accés a les aplicacions web de Nagios i Centreon és per HTTPS, per protegir el tràfic contra espies, ja que l'antenticació és pel mecanisme d' *.htaccess* d'Apache.

5 Desenvolupament

5.1 Planificació de les activitats d'integració del sistema

Setembre de 2010 – juny 2011:

Instal.lació de Nagios sobre Red Hat 5

Monitorització primers servidors Windows

Monitorització per SSH de Linux

Gradual incorporació de la resta de sistemes fins a juny de 2011.

Maig de 2011 – Juny de 2011

Instal.lació i primera configuració d'un nou sistema Centreon – Nagios. Proves pilot amb Red Hat i amb Debian.

Inici de la incorporació de tot l'entorn ja monitoritzat amb Nagios-*monitor* cap al nou sistema *Centreon-Nagios* de la màquina *centreon.administraciox.cat.* Incorporació *pilot* dels servidors de l'aula virtual a la monitorització per Centreon.

5.2 Llicència

La llicència de tots els productes instal·lats a pertot és GPL. No s'ha desenvolupat cap mòdul de programari de cap mena, per tant no s'ha generat nou codi que hagi d'estar subjecte a una llicència diferent. El projecte ha consistit en un pur desplegament de Nagios i *Centreon*. vigilant entorns Linux i Windows únicament.

5.3 Entorn de desenvolupament

No hi ha hagut entorn de desenvolupament. Sí que s'han fet proves d'assaig i error d'instal·lació de Centreon, però tota la resta s'ha anat construint de mica en mica, incorporant lenta i sistemàticament nous sistemes a la vigilància.

5.4 Documentació

No hi ha una documentació específica generada per a documentar el sistema implantat. La documentació sobre l'ús i instal·lació del producte *Nagios* ha estat tota referenciada en el present treball, i permet d'afegir nous sistemes a la monitorització bàsica, cosa que n'era la pretensió. Essencialment, la incorporació d'un nou sistema a la monitorització *Nagios* comporta l'execució dels procediments explicats a l'apartat 3.

6 Implantació

6.1 Formació

Donat que el sistema de monitorització que s'ha muntat va adreçat al Departament d'Informàtica de l'Administració X, els usuaris són tècnics de sistemes i de comunicacions del propi Departament d'Informàtica. Els administradors de sistemes, encarregats de la infraestructura de servidors, seran els encarregats d'administrar i gestionar el servidor. Caldrà explicar-los què és i perquè serveix Nagios, que hem explicat en el present projecte a l'apartat 1.4.

Serà interessant també d'insistir-los que les possibilitats de monitorització són molt extenses i abarquen pràcticament qualsevol cosa rellevant que pugui afectar la disponibilitat dels serveis. En aquest sentit, se'ls haurà d'explicar els mecanismes de funcionament de *check_nt, check_nrpe* i la monitorització per SSH. També valdrà la pena insistir que si ens convé, podem programar-nos nosaltres mateixos els nostres propis *scripts* i *plugins* perquè es puguin integrar en el sistema de monitorització i reportar a Nagios.

Així mateix, quan calgui examinar un nou servei, un dels primers llocs on anar a mirar seran les aportacions de la comunitat. Comprovarem que allò que vulguem fer no sigui ja entre la gran quantitat de *plugins* i extensions a Nagios que es poden descarregar des de http://exchange.nagios.org, i http://exchange.nagios.org/directory/Addons/Monitoring-Agents.

6.2 Implantació del sistema, proves i nivell de servei

La implantació del sistema ha estat gradual. El setembre de 2010 es va instal·lar el servidor *monitor.administraciox.cat* sobre un sistema Red Hat Enterpise Linux 5.x, aleshores el sistema Linux *homologat* al Departament d'Informàtica de l'Administració X. Gradualment, s'hi han anat incorporant fins a 73 servidors i 708 *serveis* per a monitoritzar.

Durant la fase d'incorporació de servidors al sistema, s'ha anat comprovant que efectivament funciona i ha complert les expectatives i necessitats inicials.

6.3 Acceptació del sistema

Donat que el sistema satisfà una necessitat de fa temps del Departament d'Informàtica, és un sistema acceptat ja abans d'haver-lo implementat. Les necessitats que cobreix són les mínimes requerides i les suficients per l'abast del projecte i pel que feia falta.

El sistema és de codi lliure i obert, s'integra perfectament en la infraestructura del Departament d'Informàtica, que ja disposa de les eines necessàries per a mantenir sistemes Linux. És extensible, configurable i de relativament fàcil incorporar nous serveis i servidors.

6.4 Manteniment

Actualització del sistema operatiu (automàtic).

El servidor *monitor.administraciox.cat* és un RedHat Enterprise Linux 5. Donat que tenim contracte de suport amb RedHat, el servidor s'ha incorporat al sistema d'actualitzacions centralitzades proporcionat per aquesta empresa, i segueix per tant la roda d'actualitzacions programades pel Departament d'Informàtica.

El servidor amb *Centreon, centreon.administraciox.cat,* és una Debian 5, que ha esdevingut *old-*

stable amb data de febrer de 2011, té encara manteniment d'actualitzacions de seguretat durant uns pocs anys. Naturalment, al final del cicle de vida de la distribució caldrà plantejar-se'n una migració cap a una nova versió de la Distribució.

Actualitzacions del programari

Nagios és relativament estable pel que fa al nombre de versions del producte que es van treient. Quan vaig començar la implementació del producte a finals d'agost / principis de setembre de 2010, ho vaig fer amb la versió 3.2.1 publicada el 9 de març de 2010.

L'1 de setembre de 2010 es va publicar la versió 3.2.2, i vaig decidir fer l'actualització del producte a la nova versió; cal recordar que la intefície web de *Nagios* ens avisa si hi ha una nova versió publicada (opció configurable, és clar).

Afortunadament, vaig fer còpia de seguretat de tots els fitxers de configuració (/usr/local/nagios/etc) ja que l'actualització va sobreescriure'ls. Una restauració d'aquests va fer que tot tornés a funcionar sense cap interrupció i sense cap malfuncionament. Hi va haver canvis cosmètics en els fitxers de configuració que no afectaven en res la funcionalitat de l'aplicació.

El procediment d'actualització pot arribar a ser laboriós, en el sentit que es pot deduir dels paràgrafs anteriors: no hi ha un procediment, o no l'he sabut trobar, d'actualització automàtica del sistema, i cal mirar la documentació per veure si ha canviat algun paràmetre de la sintaxi dels fitxers “cfg” de configuració.

Malgrat la bona experiència descrita amb l'actualització, si no és que hi ha una vulnerabilitat greu de seguretat que cal apedaçar, o una incorporació d'una nova funcionalitat interessant, el sistema es quedarà en la versió actual 3.2.2 durant un temps de rodatge prudencial.

Incorporació de nous servidors i serveis.

A mesura que s'incorporen nous servidors i nous serveis a la infraestructura, caldrà anar-los afegint a la monitorització seguint els procediments descrits en el treball.

Especialment, hi haurà una feina molt important de manera global per a totes les màquines i sistemes, que serà la monitorització *a alt nivell* de serveis i sistemes. Aquesta necessitarà prèviament d'una anàlisi i documentació del comportament dels serveis, i caldrà decidir quins paràmetres d'aquests són crítics per a llurs funcionament i rendiment, quines són les respostes correctes i en quin temps han de donar-les, etc. Opino que tot aquest apartat donaria per uns quants TFMs especialitzats, i és una feina molt interessant i que cal fer per a tenir una monitorització amb sentit i amb alt valor afegit.

Còpies de seguretat de les dades.

És important de remarcar que la gran majoria dels nostres servidors són virtualitzats, *monitor.administraciox.cat* i *centreon.administraciox.cat* entre ells.

La infraestructura de virtualització és altament redundant en tots els seus aspectes, tant pel que fa al maquinari que és redundant en totes les seves peces i a tots els nivells com pel programari virtualitzador:
- Servidors físics *en Cluster*, amb components redundats: fonts d'alimentació, memòria, plaques de xarxa,
- doble subministrament elèctric per les companyies,
- cabines d'emmagatzemament de disc en dos campus diferents amb dades replicades
i un massa llarg etcètera que no ve al cas descriure amb exhaustivitat.

Així doncs, el *crash* d'un sistema virtualitzat, com són tots aquests servidors als quals fem referència, és improbable, amb la qual cosa la pèrdua de les dades també ho és.

No obstant això, es fan còpies de seguretat programades de la majoria dels sistemes. En aquest sentit, i per a la infraestructgura de virtualització en VMWare, tenim comprades les llicències de l'agent per a VMWare per al sistema Symantec Backup Exec, de manera que des del nostre sistema de *backups* central fem regularment –un cop al mes-- còpia de seguretat TOTAL de tots els servidors. Això ens permet poder restaurar qualsevol d'ells a l'estat on estava a finals de cada mes de l'any (durant un any) i en quatre punts de l'any (finals de desembre, març, juny, i setembre) per dates més llunyanes a un any, de manera que la recuperació o consulta de qualsevol cosa crítica esdevé facilíssima: només cal restaurar una altra instància del servidor i copiar-ho cap a la instància en explotació.
Així mateix, instal.lem en tots els servidors el *Remote Agent for Linux Servers (RALUS)* de Symantec, que permet accelerar significativament la còpia de seguretat *granular* de fitxers i carpetes individuals dins del servidor. Només cal configurar el programari central *Backup Exec* perquè faci còpies dels directoris i fitxers que ressenyarem tot seguit, amb les periodicitats adequades.

La salvaguarda és relativament simple en el cas de Nagios, i un xic més complexa per al sistema Centreon-Nagios.

Backups de Nagios.

Tota la configuració valuosa de Nagios és a "/usr/local/nagios/etc" i els plugins que haguem pogut fer nosaltres a "/usr/local/nagios/libexec" ;dic haguem perquè la immensa majoria són provinents dels paquets instal.lats *nagios-plugins*.

Així doncs, si empaquetem aquests directoris i en fem còpia a un altre lloc, tindrem tot el que ens cal. Aquesta còpia i empaquetament podem fer-los de diverses maneres:
a.1) Un script al *crontab* que regularment faci:

```
"tar cvfz /backups/nagios_cfg.tar /usr/local/nagios/etc"
"tar cvfz /backups/nagios_plugins.tar /usr/local/nagios/libexec"
```

i després un altre que copïi regularment el contingut del directori "/backups" a un espai d'emmagatzemament compartit del qual se'n facin còpies de seguretat regulars, com per exemple un *share* SMB de qualsevol sistema de fitxers via *samba client* o *mount -t cifs;* o un disc compartit per NFS del qual se'n facin també còpies.

L'emmagatzemament a l'Administració X l'hem centralitzat en uns pocs servidors de dades, cap als quals tothom –usuaris i servidors – es connecten als directoris que se'ls han assignat, i el sistema de còpies en fa còpies regulars i programades. La solució anterior deixaria en un d'aquests servidors aquests fitxers.

Aquesta còpia dels fitxers "tar" generats es pot fer també amb el nostre programari de còpies *Symantec Backup Exec,* per al qual existeix un agent per a Linux (RALUS: Remote Agent for Linux Servers) que instal.lem quan convé als servidors Linux, i que acceleren i faciliten la còpia centralitzada del que calgui des del servidor Backup Exec, que es connecta als clients remots per copiar-ne el que convingui. Realment, per al petit volum de dades que generem amb aquests fitxers de configuració, això és innecessari bé que factible.

Backups per a Nagios-Centreon

Aquí tenim la complexitat afegida que hi ha un altre producte com Centreon instal.lat sobre Nagios. A més a més dels directoris anteriors, caldrà fer còpia de seguretat de les 3 bases de dades de Centreon que s'emmagatzemen a MySQL, i de tots els fitxers i logs generats per Centreon.
Hi ha moltes referències a *internet* amb scripts que es programen al *crontab* per a fer-ho. Bàsicament, amb la comanda *mysqldump* fem còpia de seguretat de l'esquema i les dades de totes les bases de dades, de manera consistent, així que podem restaurar-les posteriorment en un altre servidor. La BD on es guarda la configuració és "centreon", però farem còpia de "centreon" "centstorage" i "centstatus". Per a la base de dades "Centreon", la comanda seria:

```
#mysqldump -u Usuari -h ServidorCentreon -pContrasenyaSecreta Centreon >
/backups/centreon.sql
```

que naturalment caldria fer per a totes tres bases de dades; a http://www.3open.org/d/nagios/centreon/backup_centreon_database hi podem veure un de tants *scripts* de mostra que podrem copiar i adaptar.

A http://sysadminconcombre.blogspot.com/2010/11/backup-centreon.html hi ha un *script* que fa còpia de TOT nagios i centreon. No tot això és necessari per a guardar la configuració.

- /usr/local/nagios : directori d'instal.lació de Nagios, com hem explicat en el TFC.
- /usr/local/centreon : directori d'instal.lació de Centreon. La instal.lació de Centreon també crea els directoris següents:

- /var/lib/centreon
- /etc/centreon : fitxers de configuració de Centreon
- /etc/init.d : hi ha els scripts d'inicialització de nagios i Centreon
- /etc/httpd : hi ha la configuració d'Apache per a l'aplicació virtual de Centreon

D'altra banda, quan el volum de dades hagi crescut molt, serà convenient *gzip-ar* els fitxers de backup creats, a fi de poder estalviar espai.

7 Conclusions i assoliments

Un cop sabuts els requeriments del Departament d'Informàtica de l'Administració X, i després d'una anàlisi de les eines propietàries que s'han fet servir i que s'empren actualment en aquest Departament, s'ha escollit el binomi *Nagios – Centreon* per a poder monitoritzar el CPD. Aquests productes satisfan amb escreix:

Els requeriments inicials del projecte:

- Monitoritzar servidors i serveis heterogenis en un entorn multiplatforma.

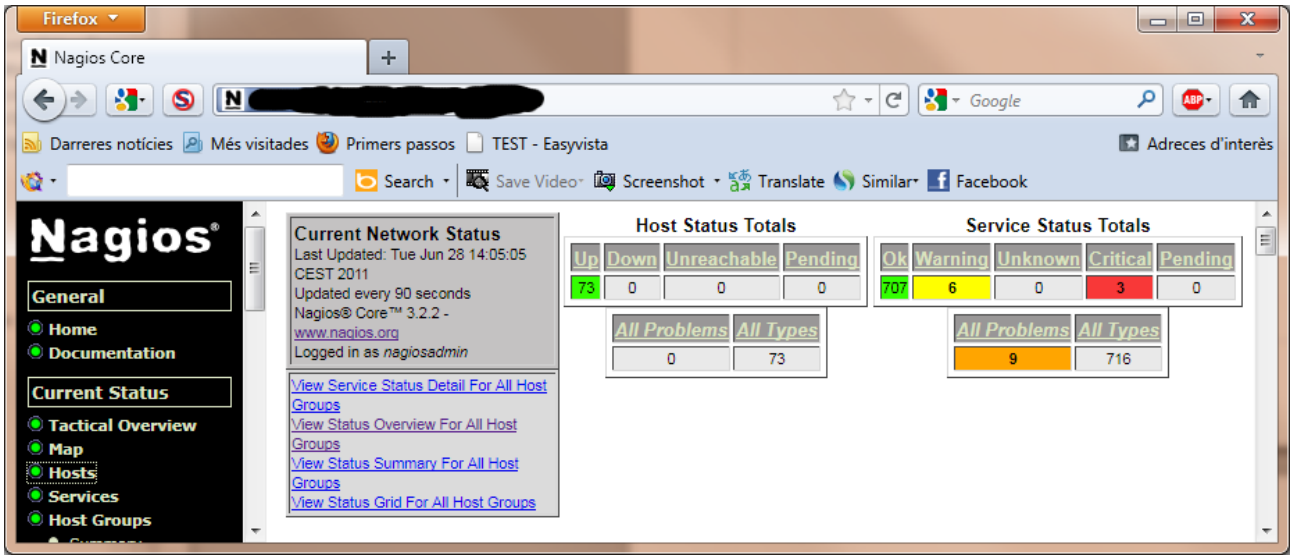
- Notificar selectivament qui convingui dels serveis/servidors que li siguin propis, i per mètodes diversos com email i SMS.

- Extensibilitat, graduabilitat en la implementació.

I els requeriments sorgits amb posterioritat (amb *Centreon*):

- Poder fer gràfiques en el temps de la disponibilitat i capacitat dels sistemes.

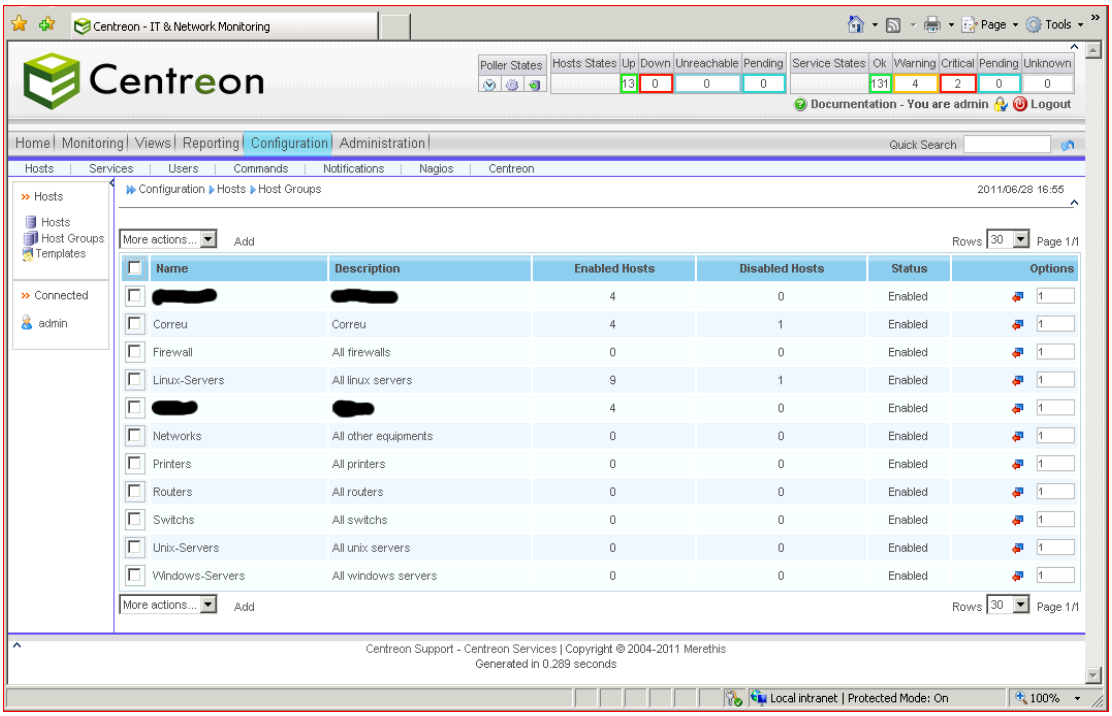
La fita inicial i primordial era la implementació de Nagios en el CPD. Així, amb aquest producte i a data de juny de 2011, s'ha assolit la monitorització de 73 servidors i 707 *serveis* :



En aquesta primera fase del projecte de monitorització, amb *Nagios*, no es va assolir la monitorització per SNMP dels sistemes. Es va veure que seria molt més simple de fer-ho amb *Centreon.*

Per diversos condicionants que s'han explicat en la memòria, excepte comptades excepcions no es fa monitorització a alt nivell dels serveis.

Durant el projecte , va sorgir la necessitat de poder obtenir gràfiques històriques que mostressin la capacitat i la disponibilitat dels sistemes. Es va veure que *Centreon* aportava aquesta funcionalitat, i que a més a més simplificava i millorava el desplegament de la monitorització. Així doncs, el maig de 2011 es va començar una prova pilot amb *Centreon*, i a data de juny ja hi ha un sistema monitoritzador amb *Centreon* cap al qual s'haurà de migrar tota la plataforma. Amb data de finals de juny hi ha 15 servidors que es monitoritzen també amb *Centreon.*



Una feina molt important i laboriosa serà la implementació d'una *monitorització fina* o *d'alt nivell.* Aquesta haurà d'interrogar els sistemes sobre els paràmetres rellevants de rendiment i de funcionalitat, per poder determinar quines són les causes que un determinat servei estigui embussat o donant un rendiment pobre, interrogant per exemple als sistemes Windows els paràmetres rellevants que també podem preguntar a un *Performance Monitor.* La dificultat d'això darrer no rau en la implementació tècnica de la interrogació, sobre la qual hi ha abundant documentació a la comunitat, sinó en tenir un molt bon coneixement del sistema que es vol preguntar, i en poder fer una correcta gestió dels resultats obtinguts si se'n té la capacitat humana.

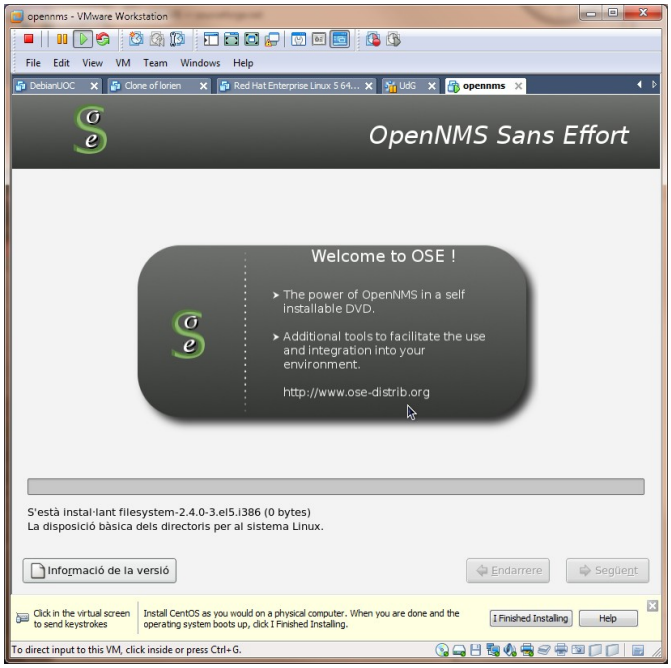
Per tal de preservar la confidencialitat deguda a l'Administració X, en aquest treball no s'inclouran els fitxers de configuració reals del sistema monitoritzador, bàsicament el contingut del directori /usr/local/nagios/etc, donat que revelarien l'estructura, els noms i les adreces IP reals i això no és desitjat pel client.

8 Annex A. Breu mostra de dues alternatives.

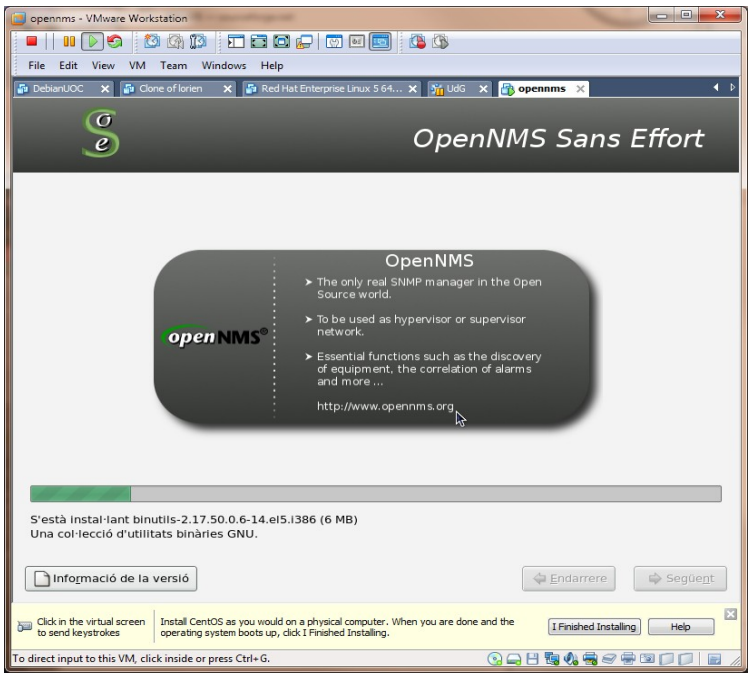
8.1 OpenNMS (http://www.opennms.org/)

És un sistema relativament senzill de muntar, i que té alguns avantatges sobre Nagios. El primer, és que fa un *autodiscovery* dels equips de la xarxa. Per a fer-ho, els fa un *ping* i aquests han de contestar, és a dir, han de tenir ICMP obert i fer ICMP echo reply. Tenir això obert en els sistemes és també necessari per a Nagios, car aquest el primer que fa per saber si un *host* és viu és executar la comanda "check_host_alive" que el que fa és fer *pings* als sistemes. Ara bé, aquesta autodiscovery és molt eficient, perquè probablement fa servir *nmap* per a identificar-ne els ports TCP/UDP oberts i per a registrar tant els *hosts* com els ports TCP i serveis que poden estar-hi associats. Això fa molt poc feixuc el muntatge inicial d'un sistema monitoritzador.

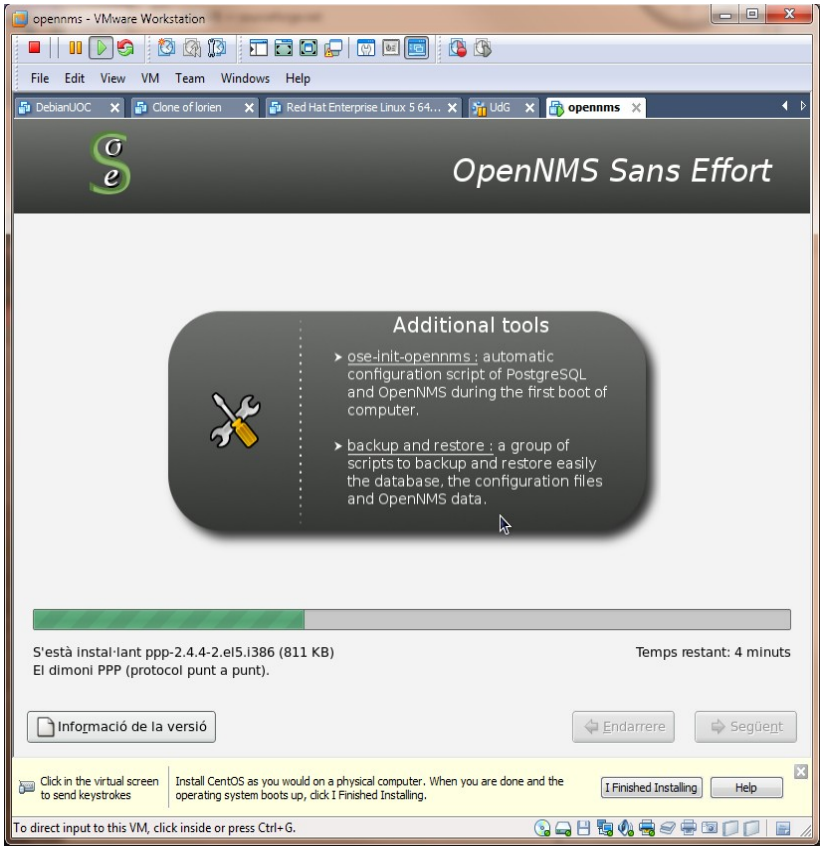
Tot just a sota un petit assaig de muntar el sistema OpenNMS a la meva VLAN domèstica; l'assaig és suficient per a percebre les potencialitats del sistema. Es pot baixar tota una *appliance* de la web de l'eina des de http://www.ose-distrib.org/en/telecharger.html , muntada sobre CentOS i fàcilment configurable seguint les instruccions de http://www.ose-distrib.org/en/documentations/62-installation-en-mode-standalone-iso.html?catid=41%3Ainstallation-via-le-fichier-iso



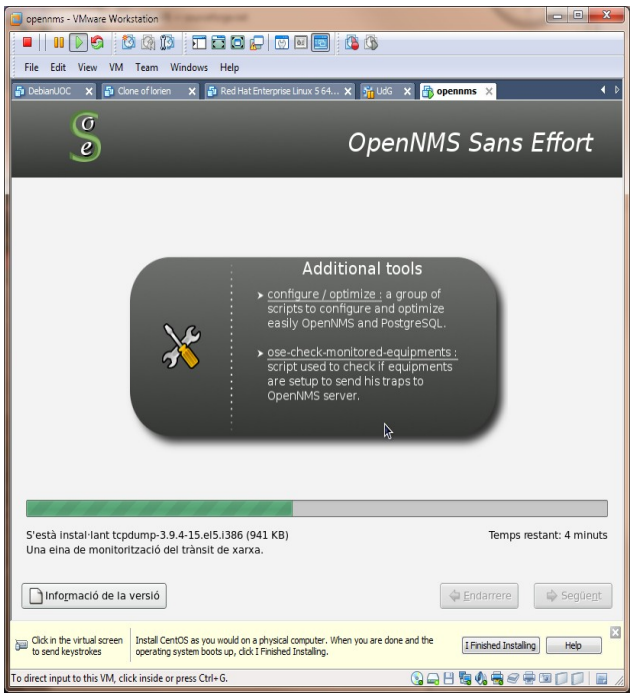
Pantalla d'instal.lació



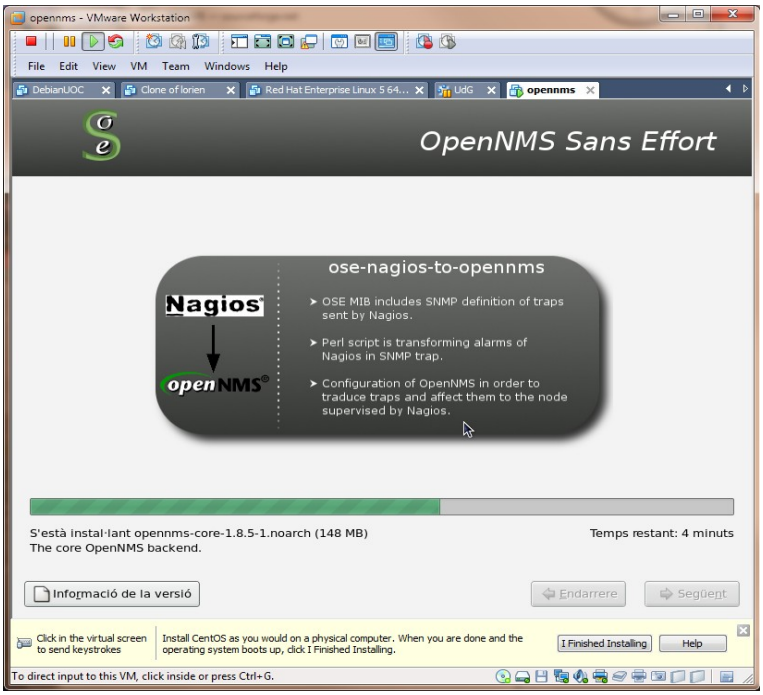
Porta integrada gestió per SNMP, i descobreix els equips i els serveis que ofereixen (alguns).



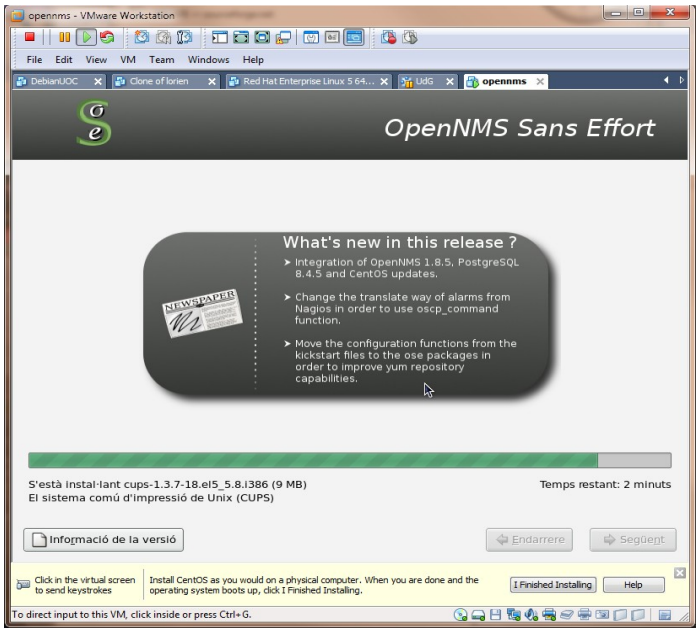
Aquesta distribució ja porta les eines d'autoconfiguració del PostreSQL que necessita OpenNMS per a guardar-hi tota la informació aplegada, i els *scripts* que en fan backup i restore.



Interroga per SNMP els equips per a saber si aquests li enviaran les *traps* SNMP.

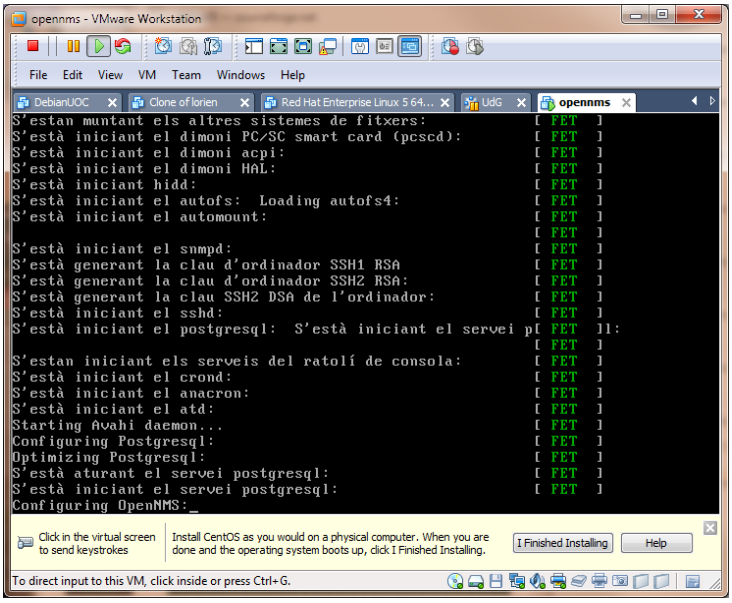


S'integra amb Nagios (tothom vol integrar-s'hi ;-))

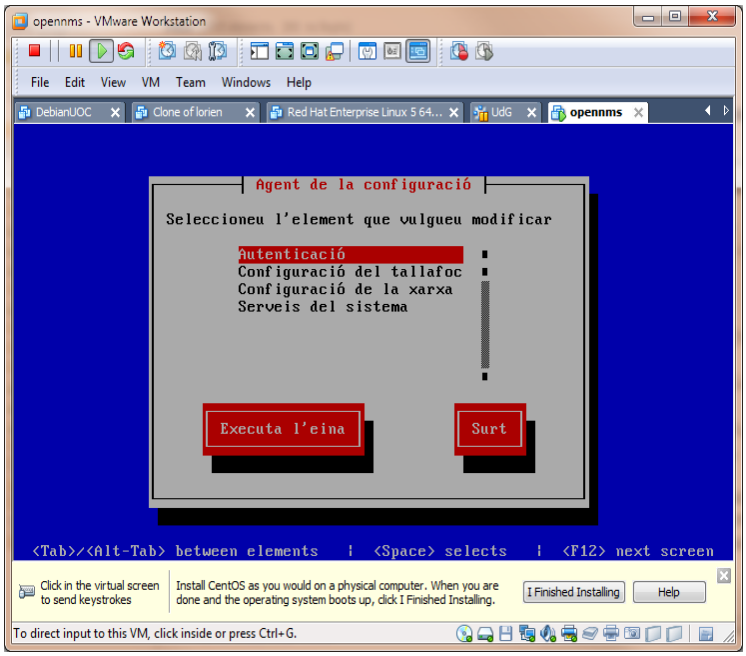


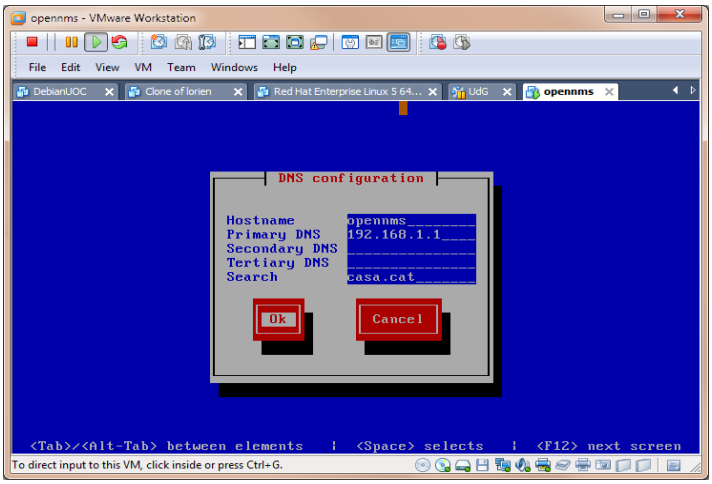
L'actualització de l'equip serà automàtica, gràcies a la integració amb els repositoris de CentOS, PostgreSQL i OpenNMS.
Realment, la llista de capacitats és impressionant i molt, molt interessant.

En reiniciar el sistema:

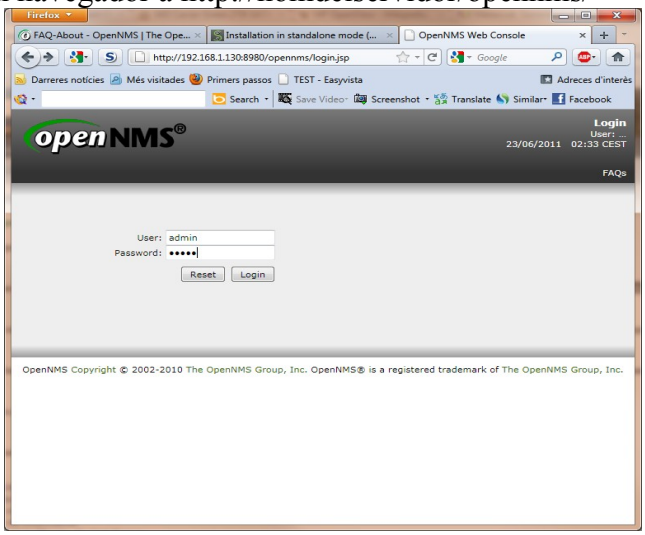


... podem configurar els paràmetres típics:

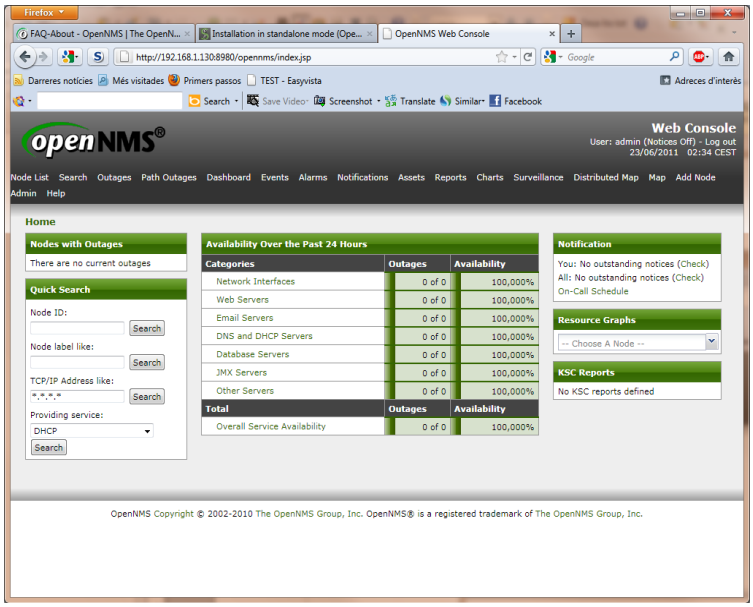


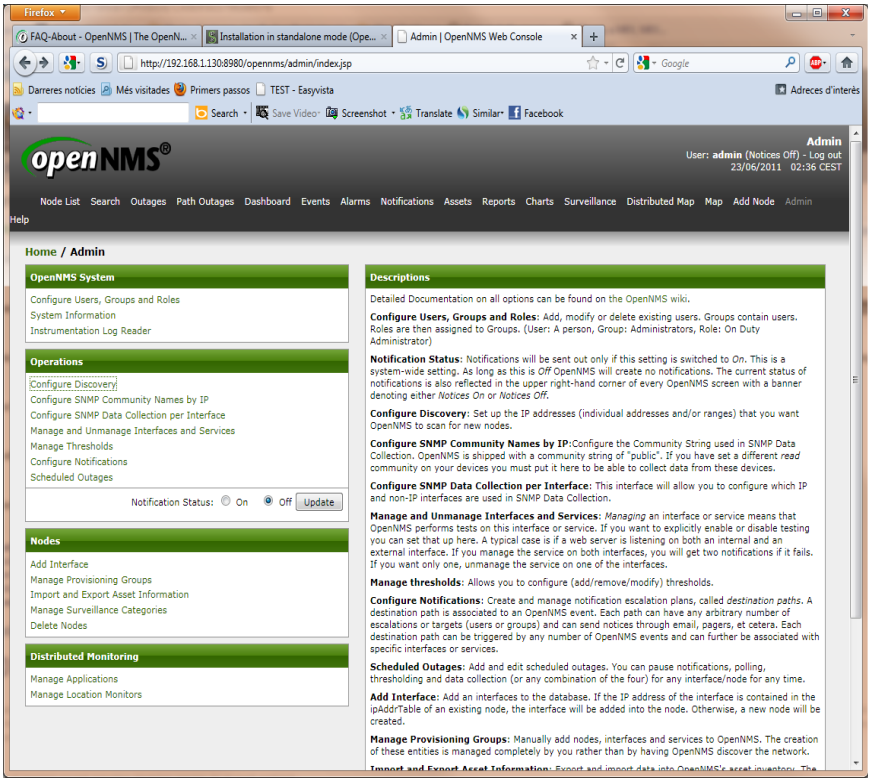


http://www.opennms.org/wiki/FAQ-About:
Ara només cal apuntar un navegador a http://nomdelservidor/opennms/

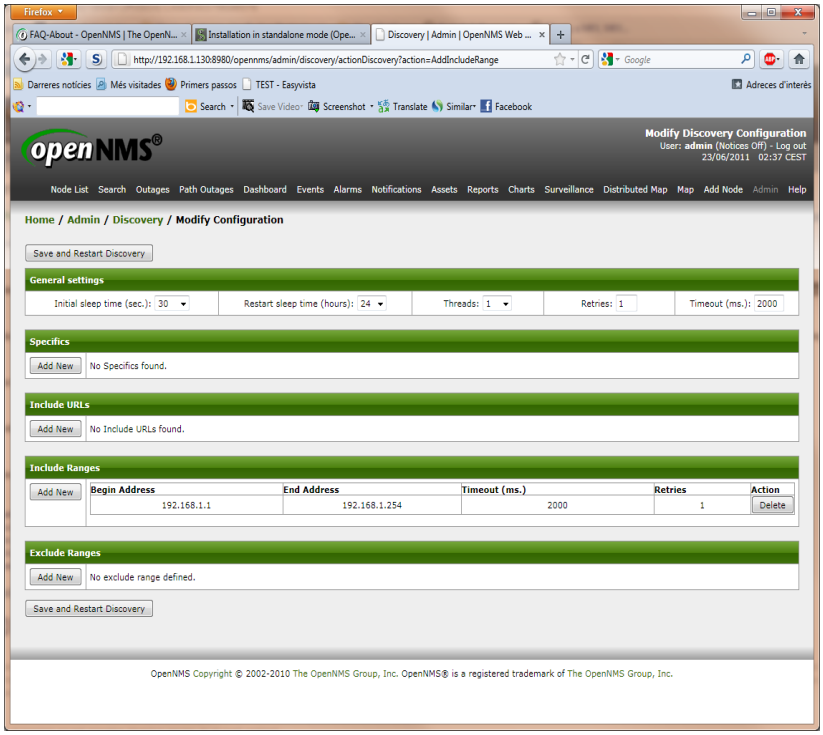


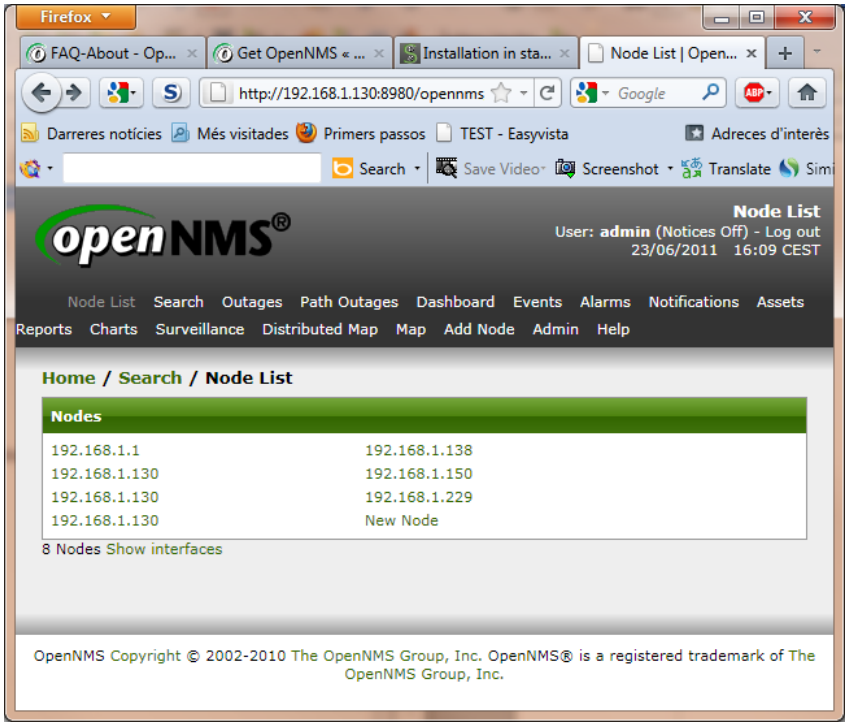
Mirem.... i configurem:



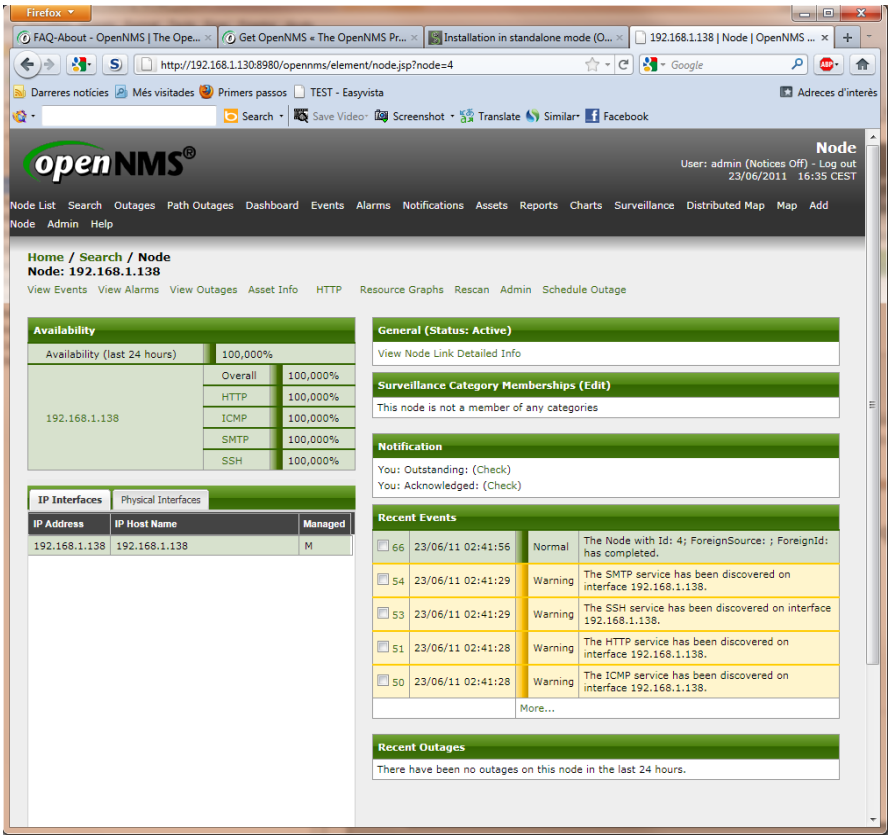


Configuro perquè descobreixi la VLAN domèstica:



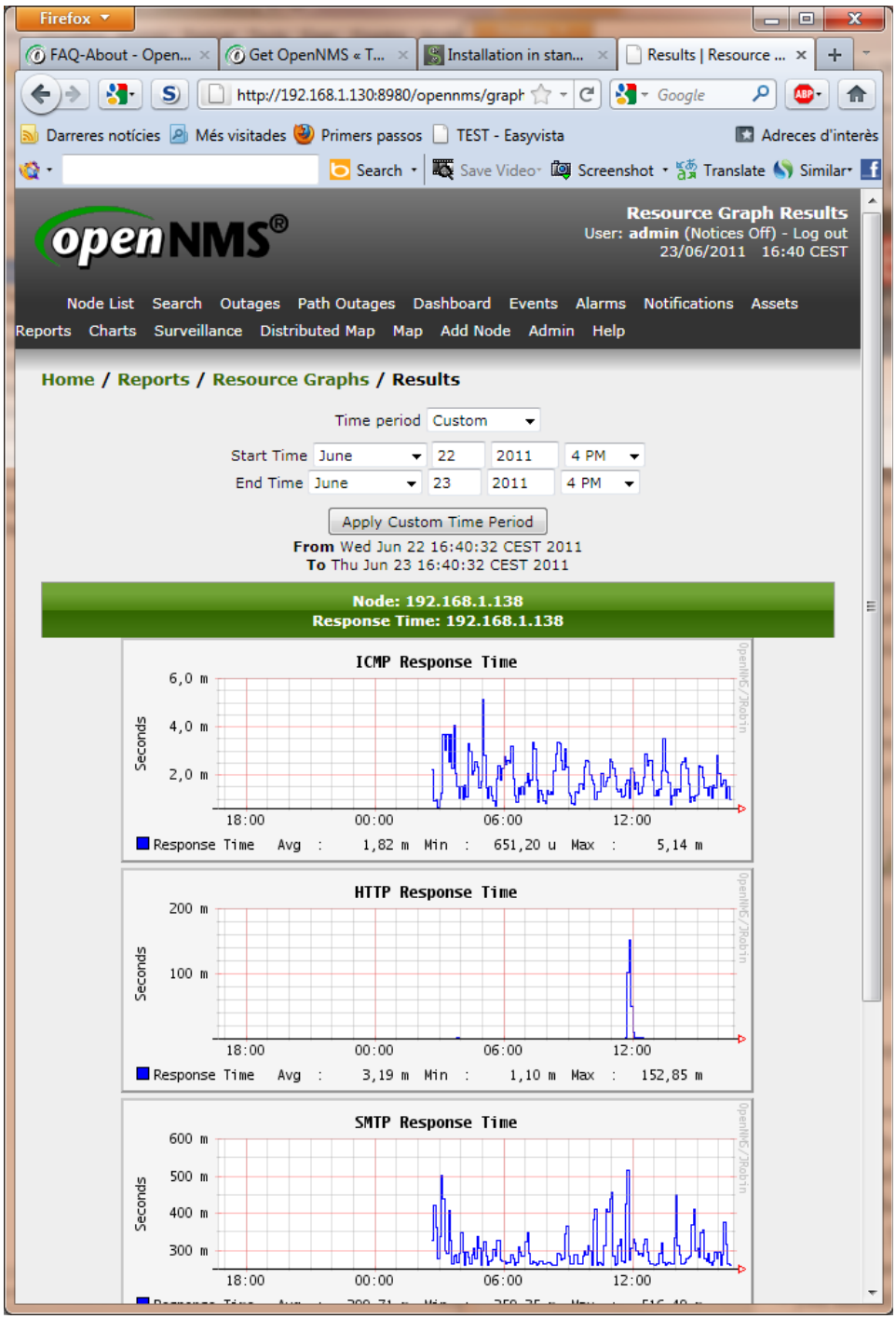


i cliquem a un d'ells i ens en llistarà el que ha descobert i per tant el que monitoritzarà a partir d'ara:



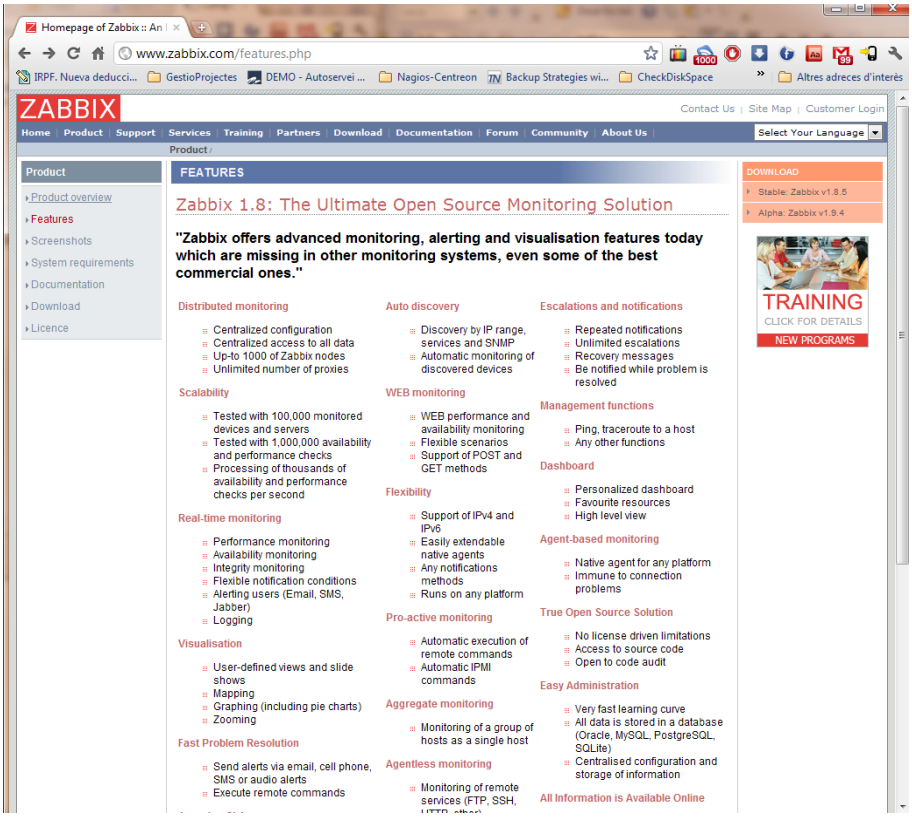
Déu n'hi do el que ha trobat, això sense fer cap configuració de res. A destacar que per exemple, no ha descobert el servei de *Samba* obert, però no es pot demanar més “sense fer res”.

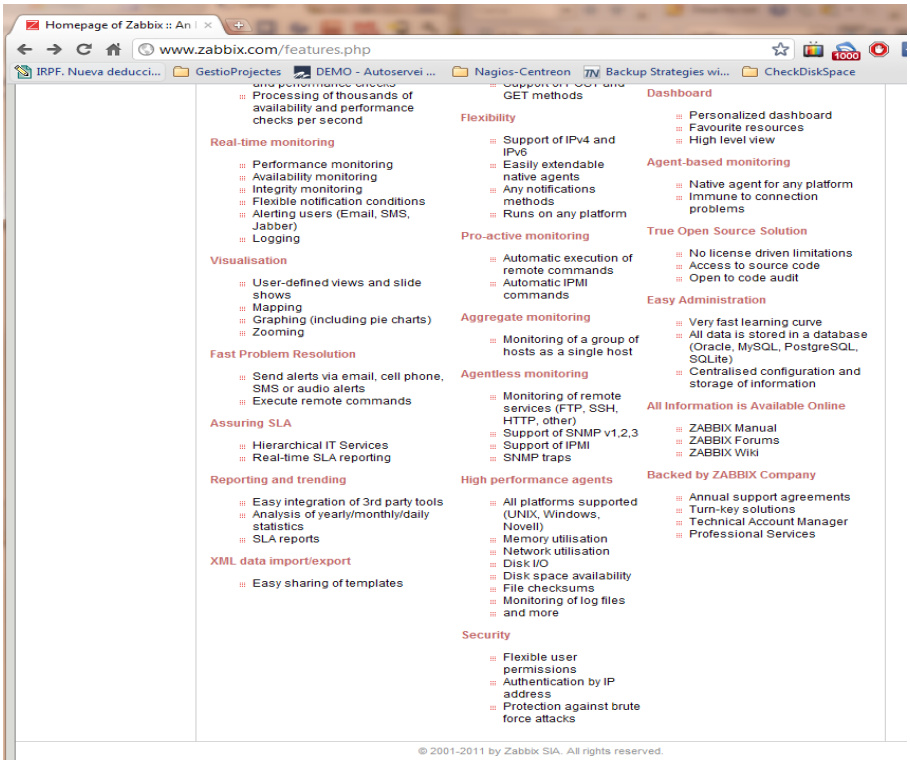
Les capacitats gràfiques són suficients i permeten generar gràfics de disponibilitat configurables en el temps:



Així doncs, aquesta alternativa sembla molt bona i fàcil, d'entrada. Perquè no es va adoptar?
- La documentació i la comunitat d'ús sembla molt més restringida que per a Nagios.
- L'extensibilitat és més acotada.
- És una eina més *tancada* pel que fa al model de desenvolupament, sense desmereixer-la en absolut però abusant del llenguatge i la comparació seria com comparar els models de desenvolupament de FreeBSD (openNMS) amb Linux (Nagios).

8.2 Zabbix (www.zabbix.com)





Les capacitats d'aquest producte són igualment interessants i impressionants, no es van avaluar amb detall en el moment de la decisió, i els motius en van ser similars que per a OpenNMS.